VISUAL SIMULTANEOUS LOCALISATION AND MAPPING METHODOLOGIES

Zoulikha BOUHAMATOU*, Foudil ABDESSEMED*

*Faculty of Technology, Department of Electronics, University de Batna 2 - Mostefa Ben Boulaïd
53, Route de Constantine. Fésdis, Batna 05078, Algeria

z.bouhamatou@univ-batna2.dz, f.abdessemed@univ-batna2.dz

received 26 July 2023, revised 20 November 2023, accepted 29 November 2023

Abstract: Simultaneous localisation and mapping (SLAM) is a process by which robots build maps of their environment and simultaneously determine their location and orientation in the environment. In recent years, SLAM research has advanced quickly. Researchers are currently working on developing reliable and accurate visual SLAM algorithms dealing with dynamic environments. The steps involved in developing a SLAM system are described in this article. We explore the most-recent methods used in SLAM systems, including probabilistic methods, visual methods, and deep learning (DL) methods. We also discuss the fundamental techniques utilised in SLAM fields.

Key words*:* simultaneous localisation and mapping, SLAM, visual SLAM, deep-learning SLAM

1. INTRODUCTION

Simultaneous localisation and mapping (SLAM) has been a focus of active study in contemporary robotics for the past few years. In this challenge, a mobile robot locates itself in an unfamiliar location and continuously generates a map of that environment. SLAM can be applied to both indoor and outdoor settings. The technique can be applied to a wide variety of fields, including underwater or aerial planning, and is not just limited to land-based mobile robots. Navigating, locating and mapping are the core technologies meant for use by intelligent, autonomous mobile robots. The goal is to first create a map of an unknown environment; then, information pertaining to the robot's motion and that of the unknown environment are determined so as to track the position of the robot in the environment.

The goal of the probabilistic SLAM problem is to find the position of the robot x_t at time k, which moves in an unknown environment, from the set of all observed landmarks m, the set of observations $z_{0:t}$, the set of commands given to the robot $u_{0:t}$ and the initial state x_0. The robot moves in an erratic manner, making it harder and harder to pinpoint where it is right now in terms of global coordinates. The robot's noise sensor allows it to detect its surroundings while it is moving. After creating the map, the goal is to be able to gauge the vehicle's position.

According to pose graph optimisation in robotics, the state of the vehicle can be indicated. There are two techniques relevant to SLAM. If the present and previous postures of the robot are taken into consideration, the full SLAM technique can determine the entire trajectory of the robot. Based on the total sensor data, it estimates the entire set of poses. The online SLAM technique is carried out if we take into account the current pose and disregard environmental features (mapping) by observing the environmental features with a sensor and applying the command vector to the robot, often based on the most recent sensor data. The rule of Bayes can present the incremental nature of the problem.

Proprioceptive, exteroceptive or a combination of both sensors is used to determine the location and mapping of a robot. In SLAM systems, well-known sensors, including GPS, SONAR, LIDAR, IR, inertial measurement units (IMU), and cameras, have been used. When a camera serves as the single external sensor, the SLAM system is known as visual SLAM or V-SLAM. Visual SLAM is primarily divided into the monocular, RGB-D and stereo SLAM techniques based on the type of camera used. (1) Monocular SLAM is focused on using just one camera. (2) The RGB-D SLAM sensor, or RGB-D camera, is made up of the monocular camera and the infrared sensor combinations. When used in RGB-D cameras, they can produce colourful images with depth and real-time 3D data. It is based on structured light. These cameras capture real-time 3D data. (3) SLAM stereo vision refers to the employment of multiple cameras, two or more lenses and a separate image sensor. The visual sensors have visual odometry on their own. It is precise, robust, and easy to implement.

The camera is the most-popular sensor for acquiring visual information. However, it has several drawbacks, such as its poor optical resolution and sensitivity, which are particularly apparent in dim and complicated lighting conditions. Several imaging technologies, including LIDAR, have been created to overcome these drawbacks. However, because cameras are so closely modelled after the human visual acquisition system (eye), they provide significant benefits in terms of their ability to record colour and texture information as well as their ease of interpretation and understanding by a human observer.

The type of map needed and the environment will influence the sensor selection. To accurately estimate the robot's pose and model the scene spatially, one can put a variety of sensors on the robot's body and combine the collected data.

The task of visual localisation depends on three principal concepts: VO (1)(2), structure from motion (SFM) (3), [4] and SLAM, where VO depends on locating the ego-motion or 3D motion of a robot by relying on the input from the camera's 'image.' It is primarily focused on reconstructing the camera's path. The SFM is based on the recovery of the relative poses of a camera and the three-dimensional (3D) structure from a set of (2D) images of a camera or video. SLAM consists of using these two pieces of information at once, estimating the trajectory of the camera while simultaneously reconstructing the environment.

Visual simultaneous localisation and mapping (VSLAM) is used to enhance surgical performance in the medical field since a large number of individuals face surgical difficulties each year. Fifty percent of these issues can be avoided with proper surgical training and evaluation. Current research combines many deep learning (DL) approaches. Automating surgical reviews, keeping an eye on surgical procedures and assisting surgeons in making decisions during operations all depend on the recognition of surgical tools and workflows to improve surgical performance. Various neural networks (NNs) have been developed in this sector to conduct tool and workflow recognition as well as to extract visual information from surgical videos (5)(6).

This research has also found applications in the agriculture domain. It is necessary to create innovative approaches that can boost production while reducing the demand for human labour. Automated and intelligent agricultural systems are crucial to addressing issues including the lack of manpower, improving worker safety and cutting production costs by preserving energy, money and time. Precision agriculture may be characterised as a strategy that enables the producer to make better decisions per unit area of land and per unit of time. Nowadays, a greater number of fruits and vegetables are cultivated in greenhouses, and it is just as crucial to monitor indoor cultivars as it is for crops produced outdoors (7). Image sensors are becoming more and more common, and they are being utilised in greenhouses to gather data for purposes like plant-monitoring techniques. A technique for identifying and categorising bacterial spot infections in tomato crops using camera pictures was developed by Borges et al. (8). In the study by Liu et al. (9), the authors take pictures of cucumbers within a greenhouse using a handheld camera, and then they apply DL to recognise the objects. This is yet another example using camera image analysis. Methods for calculating the animal condition score have also been used after digital picture processing (10)(11). Real-time site monitoring is a current difficulty in indoor precision farming and animal management. The most-common and time-consuming tasks in on-farm operations were found to be gathering data for tracking crop growth or animal conditions (12). Thus, novel remote-sensing techniques based on self-governing robots may prove to be a valuable resource for indoor agriculture and dairy farm administration in the future.

The objective of this paper is to present the evolution of SLAM since its inception, the technique used at each stage and their contributions to tackling various difficult applied research problems. The second section presents an overview of visual SLAM, its architecture and its different parts. The third section presents the probabilistic methods of SLAM. The fourth section is devoted to the SLAM based on vision. The fifth section introduces deep-learning-based approaches. The sixth section raises problems and challenges. A conclusion is drawn in section seven.

2. VISUAL SLAM OVERVIEW

VSLAM is an emerging embedded vision technology and is found very effective. The architecture of visual SLAM consists of three principal tasks: initialisation, localisation and mapping. The first phase of initialisation is to create a 3D initial map, made possible by the extraction of feature points and then determining their 3D world locations from the depth image. The phases of tracking and mapping are applied simultaneously; tracking estimates the path of the camera by matching features and refining them by tracking the local map. Localisation computes the novel 3D map points. Fig. 1 presents the architecture of visual SLAM. To improve its performance, two modules have been added: relocalisation and global map optimisation. Sometimes, the tracking process fails due to several constraints, including rapid camera motion, disturbances, scenes without texture or a dynamic environment. To solve these problems, the task of relocalisation is necessary to compute the camera pose. While the camera is moving, a previously recognised image is captured, from which the loop-closing steps are designed.

The latter compares the current landmarks with the previous keyframes. The cumulative estimation error is generated at the map level. To get rid of this error, global map optimisation is usually done. This process is done to refine the map, taking into account the consistency of the entire map information.

Visual SLAM requires feature points from the environment, but it also requires static landmarks to provide an accurate approximation. In addition, a classic SLAM and a current SLAM make up the V-SLAM domain. The classic V-SLAM technique supposedly depends on the surroundings. With a moving camera, the surroundings are actually thought of as static. During the VO procedure, a number of dynamic feature points are considered in the real world. To find dynamic feature points and discard them from the V-SLAM estimate process, modern visual SLAM integrates the architecture of V-SLAM with object detection. This method of pose estimation and mapping lowers the amount of pose estimation and mapping error by accounting for the overall movement of dynamic objects in the scene. Without using any previous object models, the environment is examined to gather all relevant data, including dynamic, geometric and contextual information.

In recent works on the current V-SLAM, moving objects in a dynamic environment are estimated and then represented in a spatiotemporal map. The estimation of the cumulative error of localisation and mapping is decreased by the improvement in feature point selection. Differentiating between static and dynamic objects is one of the most crucial aspects of the V-SLAM method. As a result, scientists have created cutting-edge algorithms based on DL, computer vision and artificial intelligence. The three stages of the authorised discrimination process are as follows: detection of prospective dynamic objects based on categories of dynamic objects that have been established. The second stage, segmentation, is optional. The third stage, optical flow estimation-based motion detection of possibly dynamic moving objects, comes next.

The ability to obtain information about the location and shape of objects is a benefit of localisation and object detection. There are overlapping concepts in this work that need to be clarified.

Classification/recognition—Finding the identity of the object in an image is necessary for this activity. In another way, assign it to a category from a list of pre-established categories. The localisation process seeks to pinpoint the object's position and create a bounding box around it.

All objects in the image are identified and classified during the object-detection process. Each object has a bounding box around it and is assigned a category.

By constructing a pixel mask for each object in the image, the segmentation approach enables a deeper understanding of the scene. Semantic segmentation and instance segmentation are the two basic categories into which it is divided. Semantic segmentation-based classification represents all pixels that fall under a certain classification. Semantic segmentation groups related items and assigns them to a single class, rather than classifying pixels. Each pixel in the image is classified into a class using instance segmentation, and each class is then assigned to a different instance of the item. Fig. 2 presents different types of segmentation.

As a result of the relative movement between the observer and the scene, optical flow expresses the shape of the apparent movement of objects, texture and edges in a visual scene. The distribution of apparent velocities in the brightness level of the image created by moving objects is another way to describe it.

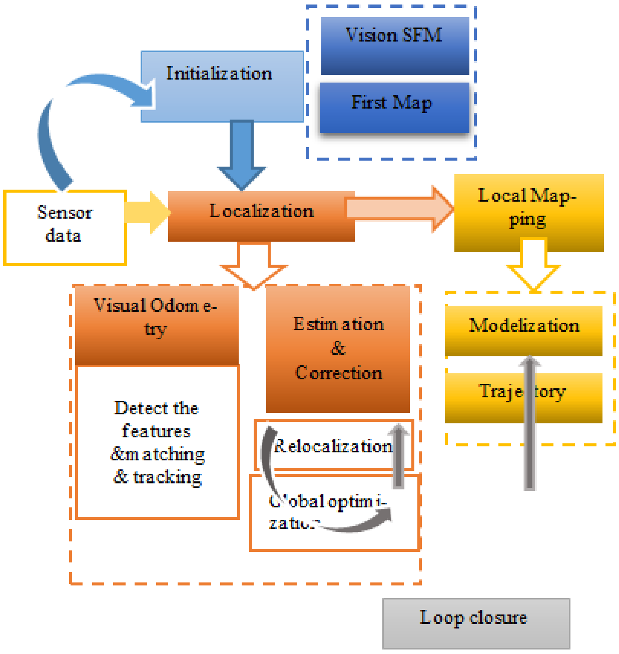

Fig. 1. SLAM Architecture, SFM, structure from motion; SLAM, simultaneous localization and mapping

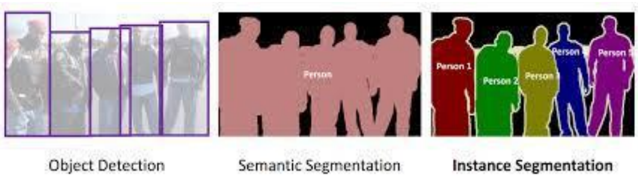

Fig. 2. Different types of segmentation

SLAM algorithms have evolved over time in response to the advancement of sensors, objectives and the pursuit of answers to specific issues in many research areas. Three phases can be identified in the evolutionary process. The initial phase was based on 1980 probability methodologies, which included filter-based techniques and optimisation-based strategies. The filtering techniques fit into iterative workflows that are appropriate for online SLAM. The full SLAM problem is addressed by the optimisation techniques, which group batch processing approaches. Classical sensors like Lidar, GPS and other sensors are the main focus of this phase. The second phase, based on computer vision and camera vision, was introduced in 2003. This technique is called vision-based SLAM. Its research advanced quickly and was able to resolve the SLAM issue and reconstruct 3D maps. The most recent phase, perception, began in 2014. The goal is to use learning to determine the system's correctness and robustness. It is based on DL, an algorithm that uses a convolutional neural network (CNN) to recognise objects in an image. Fig. 3 presents the phases of evolution.

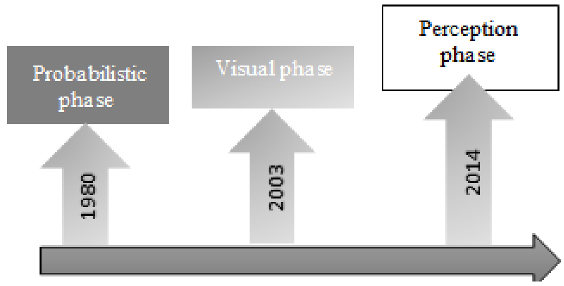

Fig. 3. SLAM evolution

3. SLAM PROBABILISTIC METHODS

3.1. Techniques based on filters

The methods based on filters are derived from Bayesian filtering. It works as an iterative process with two phases: prediction and correction. To forecast vehicle states and maps, the prediction phase first uses the evolution model and control inputs u_k. The second stage aims to correct the state that had been previously projected by comparing the map's current observation with the sensor data's current observation. The observation model combines mapping and observation. To estimate the location of the vehicle and the map, these two phases iterate and then gradually integrate sensor data.

Four main paradigms, on which various models have been created, form the foundation of SLAM. 'EKF' stands for extended Kalman filter. It is the oldest robotic system in terms of history. However, due to its restricted mathematical capabilities, it has lost some of its appeal. The second is the unscented Kalman filter that is known as 'UKF'. It was created to solve EKF issues with extremely non-linear systems. Information filtering ('IF') is the third strategy. It is the Kalman filter's (KF) inverse form. The fourth approach makes use of particle filters, which are non-parametric statistical filtering methods. They are widely used as online SLAM techniques and can address the issue of data association.

3.1.1. EKF

The first branch derivative of the KF created by Kalman (13) is the EKF. To handle linear systems, the KF was created. SLAM also employs it extremely rarely, despite its high convergence. On the other hand, The EKF can linearize non-linear systems using the first-order Taylor expansion(14).

EKF SLAM's first publications appeared in Refs (16)(17). Their method is based on estimating the movement of robot locations and a set of environmental characteristics using a single-state vector. A covariance matrix generates their estimation uncertainty as well as the correlations between the robot condition and the estimation of attributes. Using the EKF, the system's state vector and covariance matrix are updated (13)(18). When new features are noticed, more examples are added to the vector of states, and the system's covariance matrix grows significantly in size.

The EKF-SLAM methods have been extended by many authors to accommodate the issue. The calculation of the Jacobian and the linear approximation of non-linear models are two significant issues with EKF-SLAM. It can result in a filter inconsistency issue. A SLAM technique based on the central difference Kalman filter (CDKF) has been suggested by Zhang et al. (19) as a solution to this issue. For the purpose of approximating the non-linear models, the authors created the Sterling's polynomial interpolation method. It is based on trying to solve the SLAM issue in the probabilistic state space. The adaptive KF employing the AKF (20) has the benefits to include real-time processing. AKF has the ability to modify KF's parameters and improve the filtering. Additionally, the method can enhance the mapping and localisation accuracy by overcoming the challenge of information mismatch.

The adaptive EKF is a method that was proposed by Tian et al. (21) to enhance the conventional EKF. It is based on both maximisation of expectation creation (EM) and the maximum likelihood estimation (MLE). Its goal is to enable repeated approximations of the statistical noise and its covariance matrices by the standard EKF. As a result, EKF offers the capability to modify and improve the values created by MLE and EM production. Although it uses unbiased estimation to estimate the noise recursive statistics and produces high-quality results, one potential issue is that non-positive matrices of statistical covariance of process and measurement noise are defined. To lessen this issue, innovation covariance estimation (ICE) is added.

An autonomous wheeled mobile robot's SLAM problem is resolved using the suggested method (AEKF-SLAM). The disadvantage of this method is that it has a larger computational cost than EKF.

3.1.2. UKF

Julier and Uhlmann (22) introduced the unbiased KF, sometimes known as the UKF in the literature. The gradient calculation of the system equations is not explicitly used by the UKF algorithm, in contrast to EKF. The method relies on sampling particles, or a collection of points dubbed 'sigma', which are weighted around the expected value using a probability function and then passed to the non-linear function to recalculate the estimation. They enable accurate evaluation of the state vector distribution's mean and standard deviation up to the second-order approximation. Hence, to obtain the equivalent set of modified points, the sigma points are replaced. The UKF-SLAM was developed to address issues with the EKF, such as inconsistent performance. Traditional UKF-SLAM models the system as being precisely known and the perturbations as Gaussian noises with well-known statistics. Asymmetry in the actual applications could result from all these presumptions. These defects must be avoided to boost estimate precision; therefore, the Robust SLAM (RSLAM) was developed by Havangi (23). The H∞ square root UKF, which is applicable to non-linear systems with non-Gaussian disturbances, forms the basis of RSLAM. This technique has the benefit of not requiring knowledge of noise distributions or that they must be Gaussian, which makes it more adaptable and less constrained in practical applications. Additionally, because the resulting covariance matrices will continue to be semi-positive definite, RSLAM has steadily increased the numerical stability when compared to the UKF-SLAM technique. An adaptive neuro-fuzzy inference system is used to tweak the RSLAM parameters, producing better performances. RSLAM thus outperforms other UKF-SLAMs, according to the Monte Carlo simulation.

An unscented adaptive Kalman filter (AUKF), employed on the SLAM problem both in the simulation data and in the actual application, was presented by Bahraini et al. [27, 28]. It is suggested that the scale parameter, which is based on the maximum likelihood function at each time step, be adjusted. The algorithm's results show that it reduces error estimation and increases navigational accuracy.

The covariance positive defined the positive loss prevents UKF from operating, and its strong correction amount reduces the SLAM algorithm's efficiency to improve the performance of UKF. Tang et al. (26) proposed an improved Schmidt Orthogonal Unscented Kalman Filter (ISOUKF). The approach is based on a two-step modification of the UKF algorithm. The Schmidt Orthogonal transform (SOT) sampling method is used in the first step to select sample points, and the SOT sampling approach is employed to lower the computational amount of UKF to some level. The notion of a strong tracking algorithm (27) then employs an adaptive fading factor, and the prediction covariance matrix uses the fading factor effect to boot system state tracking capacity. The ISOUKF algorithm was enhanced, and the SLAM technique was made more effective in the second stage using the square root. This technique lowers processing costs while presenting a high degree of precision in tracking robots for SLAM.

3.1.3. Information filter

IF is a KF variant that is the inverse of the KF, as shown by Maybeck (28). This filter adds the vector and the informational matrices directly, presenting an inverse information matrix of the covariance matrix, which contains the state error. The extended information filter (EIF), a non-linear version of the IF, is computationally comparable to the EKF with one key distinction: the EIF has an inverse covariance matrix.

The SLAM issue was also addressed by using the candidate's EIF techniques (29). The sparse extended information filter (SEIF) technique, which was introduced by Thrun et al. (30), has been suggested as another extension of IF. The IF can be upgraded to exactly sparse extended information (ESEIF) (31), which is more consistent locally than SEIF, by leveraging parsed data for less complexity. He et al. (32) proposed the iterative sparse extended information filter (ISEIF). By solving the measurement update equations iteratively and adaptively, this approach seeks to minimise linearisation errors. The consistency and accuracy of SEIF have increased because the scaling advantage is still present. However, IF has various uses as given in Refs. [36(34). It is not popular in SLAM.

3.1.4. Particle filter

This was proposed by Del Moral (35), under the name 'Particle Filters' also called 'bootstrap filter' (36), and 'Sequential Monte-Carlo (SMC)'' (37). It is a filter that allows for finding solutions to a problem of localisation. In the observation, it does not need the limitations of the Gaussian noise, and it can adapt to any distribution. It is based on a set of generated points called 'particles'. Each of these particles represents the probable state of the system. The weight coefficients (weights) on each particle are a measure of the degree of confidence one may have in the latter to effectively represent the state. Their principle is as follows: samples of the state are taken with a set of particles according to their probability density. The particles are evaluated according to the equation of the state of the system; this is the prediction step, and then the weights are adjusted according to the observations; this is the correction step. The most probable particles are kept, the rest are removed and new particles are generated. Many versions of particulate filters have been proposed in the literature, such as sampling importance resampling (SIR) (38) and regularised particle filter (RPF) (39).

The first to make particulate filters adaptable to the SLAM problem was Blackwell (40), which is known as Rao–Blackwellisation. Doucet and Murphy (41) and Kevin Murphy (42) observed that the probability between the landmark sites is conditionally independent when the robot's route is known. Rao–Blackwellised (RB) decomposition was therefore introduced and carried out in a manner that added to the broad framework of PF for solving the SLAM issue. Based on this concept, Montemerlo et al. (43)(44) suggested the FastSLAM once more, this time utilising a few low-dimensional EKF to estimate the landmark locations and the Rao–Blackwellised particle filter (RBPF) to estimate the robot path. Stated differently, FastSLAM employs a hybrid technique that combines the PF with EKF, allowing the robot to attain more precision. The procedure in this method is predicated on the robot's prior posture prediction. Additionally, the method presumes that the landmarks are not conditionally dependent on one another while the robot's position is known. Furthermore, the robot localisation problem and the challenge of gathering estimated landmarks, both of which depend on estimating the robot's pose are separated from the SLAM by the method.

The computing complexity of FastSLAM, denoted as (M log N), is contingent upon the quantity of landmarks (M) and particles (N), both of which may have fixed values. Since every particle prescribes landmarks in a distinct way, FastSLAM performs several data associations, making the data association incredibly error-resistant. FastSLAM is easy to use and has a significant advantage in data association over EKF-based SLAM techniques, but, in some situations, the chosen samples are frequently ineffective. It is not necessary to linearise the robot's motion and measurement models. Its use in non-linear and non-Gaussian systems is more effective and convenient. The primary benefit of the FastSLAM approach is that particles carry out their own data associations, whereas the KF-based SLAM technique bases its system design on a single data association assumption for the whole filter. Furthermore, compared to KF-based approaches, the use of particle filters for robot trajectory sampling results in lower memory use and processing costs. However, because FastSLAM must perform an independent data association, it is vulnerable to divergence, and its computing cost increases significantly in noisy situations due to sparse maps. However, the limited feature position dependencies in FastSLAM instantiations lead to sluggish convergence. Moreover, the method's poor universal consistency renders it unsuitable for long-term navigation in expansive situations.

A better version of this technique, known as FastSLAM 2.0, was later published by Michael et al. (45). According to them, the proposed distribution of this approach depends on the actual measurement of the mobile robot as well as the previously estimated pose. Along with the improvements of FastSLAM 1.0, FastSLAM 2.0 also has an improved proposal distribution that results in a more consistent computing cost. The derivation of the Jacobian matrices and the linear approximations of the non-linear functions are two significant potential shortcomings of FastSLAM. It takes work to calculate the Jacobian matrix, and the estimate accuracy degenerates when the posterior covariance is not accurately approximated. To solve these problems, a novel method named Unscented FastSLAM (UFastSLAM) (46) was proposed to address linearisation-related issues in the FastSLAM framework. It is based on the use of scale unscented transformation. The linearisation procedure involving Jacobian computations is eliminated without the buildup of linearisation mistakes. This approach offers resilience in the mapping and localisation processes. However, the UFastSLAM often reduces particle diversity throughout the particle resampling process, and importance sampling is prohibited owing to covariance positive definite loss, resulting in accuracy degradation.

Variations of FP have appeared, such as distributed particle DP-SLAM approaches (47) and DP-SLAM 2.0 (48). These approaches proposed a data-storage structure based on the use of a minimal ancestry data tree. It makes quick updates by leading the PF while the number of iterations of the latter is reduced. In 2015, a new improved version of FastSLAM2.0 called six degrees of freedom (6-Dof) low dimensionality SLAM (L-SLAM) was developed by Zikos and Petridis (49). L-SLAM is based on using a particle filter of lower dimensionality than FastSLAM, for a small number of particles. L-SLAM achieved better accuracy than FastSLAM1.0 and FastSLAM2.0, and its speed surpasses FastSLAM2.0 by a factor of 3. L-SLAM is suitable for solving problems with high dimensions that have high computational complexity. To update the particles of the L-SLAM approach using a linear KF, in contrast we use an EKF to update the FastSLAM algorithm.To build a map by RBPF and ensure overall consistency, Nei et al. (50) presented an efficient system of RBPF that is an improved Lidar SLAM system by adding loop detection and correlation called LCPF-SLAM. The suggested LCPF SLAM enhances the consistency of the RBPF SLAM to be usable in comparatively wider scenarios. It also has enhanced loop identification and a new metric known as the usable ratio for determining the relevant information gained from laser readings. Still, the approach performs slowly since additional criteria are used to determine if a loop is reliable.

Resampling fixes the major flaw of the particle filter, the degeneracy of the weights, but after several iterations, particle diversity in particle concentration is completely absent; it is the problem of particle depletion. Hua and Cheng (51) proposed an adaptive fading unscented KF method (UFastSLAM) to solve the problem of particle degradation using the resampling method. It uses the UT transformation to eliminate the Jacobian matrix from the FastSLAM approach and improves the assessment of the position estimation. They replaced the KF with an unscented KF, which is suitable for non-linear systems. It also has other advantages in avoiding the accumulation of errors during linearity and has a better effect on pose estimation. In the UFastSLAM algorithm, the particles of PF are produced from the distribution of system state variables that do not depend on the posterior probability. It builds a proposed distribution function to edit and adjust the parameters adaptively and make the function of distribution closer to the system's posterior probability distribution. It is effective in improving the problem of particle degradation. An improved transformed unscented FastSLAM (ITUFastSLAM) with the adaptive genetic resampling ITUFastSLAM was introduced by Lin et al. (52). An improved importance sampling using the UKF was transformed to improve the performance of FastSLAM. A new fuzzy noise estimator is used for the improvement, which allows adjusting the state and observation noises online according to the residual and associated covariance and results from it, attenuating the flaws resulting from the imprecision of the model. They replaced the step of conventional resampling with adaptive genetic resampling. Tang and Chen (53) developed an improved adaptive unscented FastSLAM (IAUFastSLAM) with genetic resampling, to ameliorate the low tracking accuracy. This algorithm uses QR and SVD decomposition to deal with the positive definite loss of covariance in the UKF and to give the system the ability to track. They used an adaptive factor consisting of the orthogonal principle of residual vectors to predict the covariance matrix, and the function of Huber cost is generated by the modified covariance matrix to effectively eliminate the error of the measurement model. To increase the particle diversity, they used an improved genetic approach (GA) and used the suppressed sample impoverishment effectively to complete the resampling for UFastSLAM.

3.2. Techniques based on optimisation

SLAM's improvement-oriented approaches branch out into two disciplines. The first subsection is based on finding a match between the novel observations and the map derived from the sensor data. The second subsection seeks to obtain a coherent whole by refining the car's position (and subtracting the past) and the map by looking at the constraints. When it comes to optimisation, we can classify these algorithms into two main branches: the SLAM graph and bundle adjustment (BA).

3.2.1. The graph SLAM

This is an algorithm that solves the SLAM problem owing to non-linear parsimonious optimisation. Lu and Milios (54) proposed this algorithm as the first work in robotics to solve the problem of SLAM, based on the graphical representation of the Bayesian SLAM shown in Fig. 4 (55). The graphic has been translated into a matrix that describes and combines the relationships between features and robot positions. It can easily be constructed for use to optimise the framework. The graph SLAM is based on two types of nodes: motion nodes and measurement nodes. Motion nodes connect two consecutive robot locations x_{t-1} and x_t. The measurement nodes connect the poses x_t, to the landmarks m_i. The graph edges present a non-linear constraint that represents the negative logarithmic likelihood of both the measurement and movement patterns. One of the greatest disadvantages of this method is the problem of the non-linear least squares produced by the sum of all the constraints. Many implementations are used to develop Graph-SLAM TORO (56), TreeMap (57), HOGMan (58), ISAM2 (59), g2o (60), GTSAM (61), DCS (62), SacViSLAM (63) and SSA (64).

Zhao et al. (66) proposed a method named LinearSLAM to solve the problem of large scale in SLAM based on a submap joining approach. The local sub-map is constructed using the local information to find the solution to a small-scale SLAM. The advantages of combining sub-maps include solving linear least squares and establishing non-linear coordinate transformations. This approach does not require initial values and iterations since there are closed form solutions to linear least squares problems. The algorithm can be used in pose-graph SLAM, D-SLAM, feature-based SLAM and in both 2D and 3D scenarios. Holder et al. (67) presented an algorithm that builds a map from radar detections by applying the iterative closest point (ICP) algorithm with the goal of matching successive scans given from a single radar sensor. Youyang et al. (68) proposed a G-pose graph optimisation algorithm that is an algorithm without having to handle the complex Bayes factor graph. In their proposed method, they transform the absolute pose estimation problem into a relative pose estimation problem. The main advantage of the G-pose graph optimisation method is its robustness to outliers. In fact, they added a loop closure metric to handle outliers. Fan et al. (69) presented the CPL-SLAM algorithm, which is efficient and certifiably correct. It uses complex numbers to solve SLAM based on a planar graph. Sun et al. (70) proposed an active integrated method by using the method of a Cartographer to build and do efficient frontier detection. Pierzchała et al. (71) used the Graph-SLAM algorithm to generate localised forest maps. With the aim of mapping, they collected the 3D data using a specially designed mobile platform composed of several sensors.

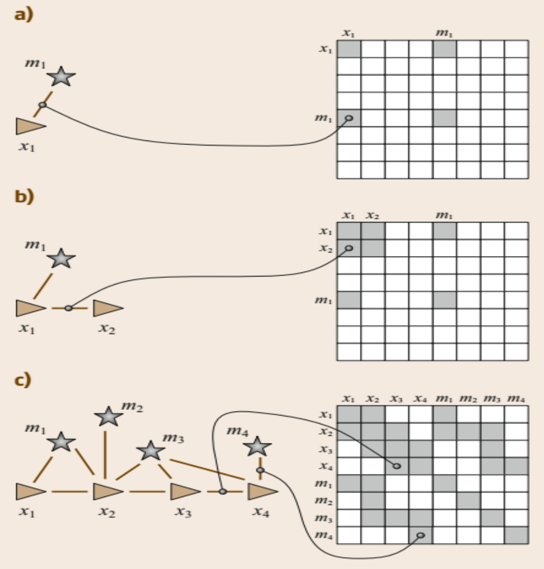

Fig. 4. (a–c) Schematic diagram of building a graph. The (a) diagram graph shows the observation 1s landmark m1, the (b) shows the constraints in the matrix form and presents robot motion from ×1 to ×2. The (c) shows several steps later (65)

3.2.2. Bundle adjustment

It is a vision technology that aims to refine a visual reconstruction of the three-dimensional structure and parameters of the camera (pose and calibrations). The symbol 'bundles' refers to rays of light leaving each 3D feature and converging on each camera centre. Then they are optimally 'adjusted' concerning both the feature and the positions of the camera. The main idea is optimisation, usually based on the objective function (ML) Levenberg-Marquardt algorithm (72). The optimisation of the best parameters (camera and landmark positions) is achieved by reducing some cost functions that determine the fitting error and finding the optimal solution concerning both structure and camera variations. To perform optimisations, many approaches have been proposed (73)(74).

In many works, the BA method is used in visual state-estimation problems SLAM and Visual-Inertial Odometry. The objective of the BA approach is to estimate the 6-DOF camera path and 3D map (3D point cloud) according to the tracks of the input feature. One of the disadvantages of this algorithm is that it is computationally heavy and cumbersome because, in the process of optimisation, it takes all the variables at once. Second, 3D structure estimation needed a baseline sufficiently inherent in BA; the algorithm of SLAM will struggle in slow–motion periods or pure rotational motion. Melbouci et al. (75) proposed a method based on combining depth measurements and monocular visual information in a cost function fully presented in pixels. The idea is to consider sparse depth information as an extra constraint in BA. Frost et al. (76) presented a technique for integrating scale data from object classes into monocular visual SLAM based on BA. In Schops et al. (77), one finds a description of a methodology named fast-direct BA formulation that can be applied in a real-time dense RGB-D SLAM approach. This results in the use of rich information in global optimisation, which obtains paths with great precision. Zhao et al. (78) proposed a novel, rigorous and efficient method called good graph. They used a BA-based V-SLAM back-end to improve their cost efficiency. Their objective is to define graphs with small sizes to be improved in the phase of local BA using preservation conditions. Wang et al. (79) introduced the SLAM framework based on saliency and the backbone given by ORB-SLAM3 (80). They developed the salient BA based on the saliency map value that can make the salient feature fully play its value. A new SLAM system is proposed by Gonzalez et al. (81). It uses the semantic segmentation of objects and structures in the scene. The authors modified the classical BA formulation using geometrical priors to constrain each cluster, which allows for improving both camera localisation and reconstruction and enables a better understanding of the scene. Tanaka et al. (82) proposed a learning-based BA based on a graph network. It replaces the standard Levenberg–Marquardt approach of BA with an algorithm based on learning. The advantage here is that it runs very fast and can be applied instead of conventional optimisation-based BA. Tab. 1 summarises the probabilistic methods by specifying the type of algorithm adopted for each method and the year they appeared.

Tab. 1. Probabilistic methods strengths and problems

SLAM- probabilistic methods				
Methods	**Type**	**Algorithm**	**Year**	**Comment**
Filters- Techniques	Extended Kalman	KF (13)	1986	Strength: -Efficient convergence -Adapt to uncertainty -The mean must be known. Problem: -Gaussian restrictions -The issue with high-dimensional maps. -Association of data in large environments -First order Tylor expansion. -Vulnerable to linearisation errors.
		EKF (15)	1990	
		CDKF (19)	2009	Strength: -Approximate the non-linear model-solve the SLAM issue in the probabilistic state space -Reduction in the ambiguity for data association. Problem: -Calculate the mean and covariance.
		AKF (20)	2016	Strength: -Gain adjustment in real time-accurate-robust mapping. -Strong estimation for AKF-Unbiased estimation for AEKF. Problem: -High computational cost-Association data problem -Gaussian noise.
		AEKF (21)	2020	
	Unscented Kalman	UKF (22)	2000	Strength: -Dealing with non-linearities-Coping with uncertainty -Expansion of the second order Taylor. Problem: -It is necessary to know the mean and covariance -Assumes that the system is exactly understood and that disturbances are stationary Gaussian noises with known statistics. The covariance positive defines loss and its calculation amount is large.
		AUKF [27(25)	2019	Strength: -Find the appropriate value for the scaling parameter and improve the estimate accuracy-Accurate.
		RSLAM (23)	2016	Strength: -Applied to non-linear systems with non-Gaussian noise -It is more flexible and adaptative -Has fewer limitations in real application.
		ISOUKF (26)	2022	Strength: -High precision-reduces computational cost -Accuracy and efficiency.
	Information	IF (28)	1979	Strength: -Straightforward and easy to execute-Handle maps with high dimension. Problem: -Challenges when integrating maps-Issue with connection of data.
		EIF (29)		
		SEIF (30)	2004	Strength: -Representation of graphical grids. -Sparsification-constant computational cost. Problem: -Inadequate representation-Iterative and slow.
		ESEIF (31)	2007	Strength: -More consistent with leveraging parsed data.
		ISEIF (32)	2015	Strength: -Measurement update equations iteratively and adaptively.
	Particle filter	PF (35)	1996	Strength: -Handles non-linearities -Handles with non-Gaussian noises. Problem: -Big complexity-Data Association.
		Rao-Blackwell PF (40)	2000	Strength: -Cost of calculation in logarithms-linearisation is not necessary-Accuracy. Problem: -Data association has a high cost-The landmarks' information is limited. -Higer dimensional map.
		FastSLAM (43)	2002 2007	Strength: -Higher accuracy-Path and landmark estimation 2007. -Does not necessary to linearise the robot's motion and measurement models. -Its use in non-linear and non-Gaussian systems -FastSLAM2.0 more consistent computing cost-FastSLAM2.0 linearises the non-linear model. Problem: -It must perform an independent data association -It is vulnerable to divergence. -It is computing cost increases significantly in noisy situations due to sparse maps -Universal consistency renders it unsuitable for long-term navigation in expansive situations. The derivation of the Jacobian matrices and the linear approximations of the non-linear functions.
		FastSLAM 2.0 (45)	2003	

DOI 10.2478/ama-2024-0049

		DP-SLAM (47)(48)	2003	Strength: -Data storage. -It makes quick updates.
		L-SLAM (49)	2015	Strength: -Uses small number of particles.-Better accuracy than FastSLAM. -Speed-Solve problems with high dimensions that have high complexity computational.
		LCPF (50)	2020	Strength: -Improved loop detection .-Detects the useful information obtained from laser readings. - Improve the consistency. Problem: -The approach performs slowly since additional criteria are used to determine if a loop is reliable.
		ITUFastSLAM (52)	2019	Strength: -Adjusts the state and observation noises online.
		UFastSLAM (51)	2020	Strength: -Solves the problem of particle degradation. -Uses the UT transformation to eliminate the Jacobian matrix.-Improves the assessment of the position estimation.-Avoids the accumulation of errors. Problem: -Reduces particle diversity throughout the particle resampling process. -The importance sampling is prohibited owing to covariance positive definite loss.
		IAUFastSLAM (53)	2021	Strength: -Ameliorates the low tracking accuracy. -Deals with the positive definite loss of covariance in UKF. -Predicts the covariance matrix. - Eliminates the error of the measurement model effectively.
Optimization technique	The Graph SLAM	TreeMap (57)	2006	Strength: -incremental optimisation approach-Update O.log N/ time. Problem: -Only provides a mean estimate
		TORO (56)	2008	Strength: -Optimisation strategy based on SGD. - Robust under the poor first predictions. -Assumes that constraints have covariance matrices that are generally spherical. Problem: -ecovers fast from big mistakes but has delayed minimum convergence. -Only provides a mean estimate.
		HOGman (58)	2010	Strength:- Incremental optimization approach via hierarchical pose graphs and lazy optimization. Problem: -Requires pose-graphs with full rank constraints.
		g^2o (60)	2011	Strength: -Flexible and readily adaptable SLAM optimization framework -It includes many optimisation methods and error routines. -External plugins are supported.
		iSAM2 (63)	2012	Strength: -General incremental non-linear optimisation with variable elimination. -Sparsity is preserved by variable re-ordering. -Relinearization of specified variables on demand.
		GTSAM (61)	2012	Strength: -Flexible optimisation framework for SLAM and SFM structure derived from motion. -Direct and iterative optimization approaches are used. -SAM, iSAM, and iSAM2 are all supported. -BA for Visual SLAM and SFM is implemented.
		SSA (64)	2012	Strength: -Optimises both robot positions and proximity sensor data. -Estimates the smoothness of a surface. -Assumes the presence of a range sensor (e.g., laser scanner, Kinect, or similar).
		DCS (62)	2013	Strength: -Outliers are dealt with by optimising with a strong cost function included into g2o.
		SacViSLAM (63)	2011	Strength: -For on-the-fly processing, it combines local bundle correction with sparse global optimization.
		LinearSLAM (66)	2018	Strength: -Solves the problem of large scale.
		G-pose graph (68)	2020	Strength: -handling the complex Bayes factor graph. -It is robustness to outliers. -Adds a loop closure metric to handle outliers.
		CLP-SLAM (69)	2020	Strength: -It uses complex numbers to solve SLAM based on a planar graph
	BA		1999- – 2023	-The primary concept is optimization. -Based on the Levenberg –Marquardt algorithm's objective function (ML). -The optimal parameters (camera and landmark locations) are optimised by lowering several cost functions that affect the fitting error. -Finds the best option in terms of structure and camera variations. -It is important to note that many SLAM methods developed after 2014 do not exclusively fall into the category of DL-based SLAM. Pose graph optimisation with BA remains a mainstream back-end algorithm.

AKF, adaptive Kalman filter, AEKF, Extended Adaptive Kalman filter, UKF, unscented Kalman filter, AUKF,(24)(23)(23) adaptive unscented Kalman filter; BA, bundle adjustment; (25)(24)(24)CDKF(19)(18)(18), central difference Kalman filter; DCS, *(62)(61)(61)*dynamic covariance scaling; DL, deep learning; (47)(46)(46)EIF, extended information filter; EKF, extended Kalman filter; ESEIF(31)(30)(30), exactly sparse extended information; IAUFastSLAM, improved adaptive unscented FastSLAM; IF(28)(27)(27), information filter; iSAM, incremental Smoothing And Mapping; ISEIF, iterative sparse extended information filter; ISOUKF, improved Schmidt Orthogonal Unscented Kalman Filter; ITUFastSLAM, improved transformed unscented FastSLAM; (52)(51)(51)KF, Kalman filter; L-SLAM, low dimensionality SLAM; (23)(22)(22)RSLAM, robust SLAM; *(63)(62)(62)*SAM, smoothing and mapping; SEIF, sparse extended information filter; SFM, structure from motion; SGD, stochastic gradient descent(13)(13)(13); SLAM, simultaneous localisation and mapping; SSA, sparse surface adjustment; UFastSLAM, Unscented FastSLAM, distributed particle DP-SLAM. iterative closest point (ICP), G-pose graph optimisation, CPL-SALM Correct Planar Graph-Based SLAM. LCPF: A Particle Filter Lidar SLAM, HOGman, Hierarchical optimization on manifolds, GTSAM, Georgia Tech Smoothing and Mapping, DCS, Dynamic Covariance Scaling.

4. VISUAL SLAM AND RGB-D-SLAM

4.1. Classical methods

In theory, the method of visual localisation uses the theory of geometry mainly to estimate motion; it is based on the extraction of geometric constraints from images. It is based on elegant well-established principles and is extensively studied. The VO algorithms can be classified according to the type of image used: stereoscopic or monocular VO. Their processing techniques are based on feature direct and indirect methods, which are 'appearance-based' and 'feature-based', respectively.

4.1.1. Feature-based methods

The first approach 'feature-based' or the indirect method is based on two steps: detecting and tracking a set of salient features of the image, such as corners and lines, and following them in the following images. The calculation of the Euclidean distances of each element, the points between frames and the displacement and the velocity vectors by using detectors such as: Feature From Accelerated Segment Test (FAST) (83), Speeded Up Robust Features (SURF) (84), Binary Robust Independent Elementary Features (BRIEF) (85), Oriented Fast and Rotated BRIEF (ORB) (86), Harris and Stephens (87) detected the corners. The features are used to estimate the camera's state and reconstruct the environment. This technique is able to deal with large motions from frame to frame due to the distinctiveness of the features and is ideal to optimise the motion of the camera and the geometric structure; BA is suitable for its use. Camera tracking depends on the geometric feature error by reducing the Euclidean distances between the two corresponding sets of geometric primitives in 2D or 3D. The geometric errors are classified into three types: 2D point-to-point error, 3D point-to-point error and 3D point-to-plan error (88). Several techniques have been developed for this approach:

MonoSLAM The first monocular V-SLAM was developed in 2007 (89)(90). They were based on estimating simultaneously the movement of the camera in 6-DoF and the 3D positions of the characteristic points of an unknown environment by applying an EKF and representing them as a vector of state in EKF. The disadvantage of this approach is that the computational cost increases proportionally with the size of the environment. The algorithm of parallel tracking and mapping (PTAM) has been proposed to solve this problem (91)(92). The PTAM algorithm divided both tracking and mapping into different threads on the CPU, which are run in parallel, and therefore the computational cost is not affected. PTAM is the first algorithm that integrates a BA optimisation process into real-time V-SLAM algorithms with freed-up computing capacity.

RGB-D-SLAM was proposed by Endres et al. (93). The approach created for SLAM is based on RGB cameras. This system enables it to handle challenging data in common indoor scenarios and is fast to work online.

ORB-SLAM was designed by Mur-Artal et al. (94). It is an extension of the main ideas of PTAM algorithms: location recognition (95), scale-sensitive loop closure (96) and use of co-visibility information for large-scale operations (63), with some improvements and novelties. Indeed, ORB-SLAM is a feature-based single SLAM system that operates in real–time; the third parallel phase is added to detect the loop closure [105(98). All these additions make the system efficient and reliable.

ORB-SLAM2 was proposed by Mur-Artal and Tardos (99). It is an extension of the ORB-SLAM algorithm. It is suitable for monocular, RGB-D and stereo cameras and allows the reuse of maps, relocalisations and loop closing. In RGB-D results, the use of BA presents more precision than the methods of ICP or photometric and depth error minimisation. In stereo SLAM, they used near and far stereo points and monocular observations; the results depicted a high accuracy compared to the direct method. It allows reusing the map with mapping disabled by using the lightweight localisation mode.

OpenVSLAM was proposed by Sumikura et al. (100). It is a visual SLAM framework; it corresponds to a monocular, stereo and RGBD visual SLAM system, which contains a basic SLAM algorithm. These modules allow creating local and global maps and store and load them.

UcoSLAM was developed by Muñoz-Salinas and Medina-Carnicer (101). It is a monocular V-SLAM system fusing natural and artificial features to have strong long-term tracking. This gives the system an advantage; it can initialise both markers and key points. It makes the real scale of the maps accessible as long as a marker is available. It can solve problems caused by repetitive environments, false relocalisations and loop-closures by using the markers. It is distinguished from ORB-SLAM2 in that it can load and store the generated maps. The main idea to combine the plane and edge features was proposed by Sun et al. (102), named plane-edge-Slam. This methodology estimates robust motion, which depends on constraint analysis and an adaptive weighting algorithm.

4.1.2. Appearance-based methods

The second technique, 'appearance-based' or the direct method, estimates camera movements directly using pixel-intensity changes, usually photometric errors. The pixel selection can be all pixels (dense) or a sparse selection (sparse). The direct method eliminates feature extraction time at a cost that is much greater for optimisation problems than the feature-based method.

DTAM: The first direct method is called 'Dense Tracking and Mapping' and was published by Newcombe et al. (103). It is a method for tracking and reconstructing images from live cameras. To monitor the dense camera, it records the full image with the intention of creating a dense 3D surface model and using it right away. This approach offers keyframe tracking based on the reduction of photometric errors but does not include the closure-detection procedure or global optimisation.

LSD-SLAM: Engel et al. (104) created the large-scale semi-dense (LSD) SLAM. It uses the monocular camera VO technique. To estimate a semi-dense inverted depth map of the current frame, the primary idea is to use dense image alignment to track camera movement. The semi-dense VO was extended to the LSD-SLAM by Engel et al. (105). The recent advancements in this technique are based on a scale-aware image alignment algorithm to increase the similarity transform $\xi \in \mathrm{sim}(3)$ between two keyframes. It is a monocular SLAM system that seeks to preserve and track the global map of the environment. The authors propose a new direct tracking method that allows for detecting and explaining scale drift. They developed a probabilistic method for the fusion of noisy depth estimation with tracking. In 2015, Engel et al. (106),(107) used LSD-SLAM with stereo cameras and omnidirectional cameras.

SVO 'semi-direct visual odometry’: It is a reliable semi-direct monocular VO algorithm, proposed by Forster et al. (108). They used a probability mapping method that explicitly models external observations to estimate the 3D points, which results in fewer outliers and more accurate points.

DSO ‘direct sparse odometry’ was created by Engel et al. (109). It aims to combine a model that minimises optical error (full direct probabilistic) and optimisation for all model parameters represented by the intrinsic camera, extrinsic camera and inverse depth value. It is based on continuous photometric error optimisation over a window of recent frames, accounting for the model of a photometrically calibrated image. Gao et al. (110) proposed the LDSO, which is a development of the DSO that adds closing loop detection and pose-graph optimisation. They used a conventional feature called bag-of-words (BoWs) to inject the feature points into the loop closure (95). Another extension of DSO, called dynamic-DSO is proposed by Sheng et al. (111). It is a semantic direct VO of monocular vision using DL in the process of semantic-image segmentation. They applied CNNs to the original RGB image to extract the pixel-level semantic information of dynamic objects.

KinectFusion was introduced as a real-time mapping system in complex conditions and changing lighting by using a moving depth-camera called ‘hand-held Kinect’ and commodity graphics hardware (112). The obtained current sensor position tracks the live depth frame relative to the global model by applying an iterative nearest point (ICP) algorithm.

RGB-DTAM developed by Concha and Civera (113) introduced a direct RGB-D SLAM system with the ability to close the loop and reuse the map. With advanced technology, the approach allows accuracy and durability at a low cost. The inclusion of multiple RGB visibility limitations in thread tracking and mapping is the technique’s key innovation. Extending the RGB-D sensor range, using high-parallel setups, and adding distant locations to the map all improve estimation accuracy.

ID-RGBDO proposed by Fontán et al. (114) aims to achieve great accuracy in calculating the direct speed of RGB-D with minimal losses. Therefore, they introduced new, efficient information to determine the most informative measurements in BA and position-tracking optimisations.

4.1.3. Semi-direct

Another highly popular approach is called semi-direct; it combines the benefits of the two methods mentioned above as well as the success aspects of the feature-based process, such as tracking numerous features, parallel tracking and mapping, with the accuracy and speed of direct methods.

CPA-SLAM was developed by Ma et al. (115). This technique combines frame-to-keyframe and frame-to-plane data. It is co-optimised with alignment constraints between keyframes for global consistency. This technique creates a global model that enables position estimation. A world map is made by segmenting the RGB-D picture planes using the ‘agglomerative hierarchical clustering’ method and an information association rule. The CPA-SLAM technique provides a photometric residual, a point-to-point residual and a plane-to-plane residual, which use the EM frame to minimise jointly to estimate the camera position.

BundleFusion proposed by Haque et al. (116) is a global pose-optimisation framework, the parallelisable sparse-then-dense. It is a method that accomplishes robust tracking while performing online real-time 3D reconstruction. Additionally, by improving the path globally for each frame retrieved, the loop-closure problem is solved.

KDP-SLAM 'keyframe-based dense planar SLAM’ was proposed by Hsiao et al. (117). To estimate odometry, they used a fast dense approach. The depth values from small baseline images are combined in a local map to build dense 3D structures and extract planes. Then they used the method of incremental smoothing and mapping (iSAM) to optimise the positions of keyframes and landmark planes.

FSD-SLAM' fast semi-direct SLAM‘ was created by Dong et al. (118). This method’s goal is to combine the feature point approach with a direct way to estimate and enhance the system’s accuracy in a setting with few visual elements and little texture. Based on the sub-graph, a reliable feature point-extraction technique was selected. They suggested a reliable technique based on apparent shape-weighted fusion to determine the camera’s position. The incremental dynamic covariance scaling (DCS) approach reduces the inaccuracy in calculating the camera location. They suggested a face element model based on the improved camera position to obtain a flawless 3D point cloud map as well as estimate and integrate the point cloud pose. Tab. 2 summarises all these techniques in the order present in the text.

4.2. Visual-inertial odometry (VIO) methods

The combination of an IMU and a VO system is the foundation of the VIO technique. The fundamental concept is to combine visual data with inertial measures to produce a more accurate and effective measurement. IMU is characterised by strength in certain situations, such as speed motion, textureless and lighting changes. Therefore, IMUs are used because they provide reliable information that we can use instead of visual information, or they add information in typical cases.

VIO systems may be classified into two primary streams: loosely coupled and tightly coupled techniques, based on directly or indirectly fused readings from sensors. In loosely coupled techniques, pictures and IMU measurements are processed by two estimators that estimate relative motion independently. The final result is obtained by fusing the estimates from the two estimators. Tightly coupled techniques combine raw data from the camera and IMU directly into one estimator to find optimum estimates. Tightly connected techniques are often more accurate and resilient than weakly coupled approaches.

ROVIO is presented by Bloesch et al. (119) as a monocular VIO method. It uses the errors of pixel intensity from image patches, which gives accurate and robust tracking. After detection, the multi-level correction feature tracking is based on a basic EKF by directly using the errors of intensity.

MSCKF-VIO stands for multi-state constraint KF used in stereo VIO without GPU. It was proposed by Sun et al. (120) as an approach that proved its accuracy, efficiency and durability compared to other algorithms. This method uses the multi-state KF, which was developed by Mourikis and Roumeliotis (121). It is used in stereo VIO without a GPU.

OKVIS ‘Open keyframe-based visual inertial SLAM’ is provided by Leutenegger et al. (122). It is a tightly coupled framework presented as a combination of both inertial measurements and image key points. The goal is to form keyframes in the problem of non-linear optimisation that uses linearity and marginalisation.

Maplab was developed by Schneider et al. (123). It is a platform written in the C++ language for visual-inertial mapping. It is a system ready for planning and visual localisation and offers researchers a set of tools for multi-session mapping that allow map merging, loop closure and inertial batch optimisation.

Tab. 2. Comparison of visual SLAM methods

V-SLAM methods	Name	Year	Camera Model	Back-End	Mapping	Relocalisation	Loop-closure
Feature-Based	Mono-SLAM (89), (90)	2007	Monocular	Filter-based	Sparse	No	No
	PTAM (91)	2007	Monocular	Optimisation	Sparse	No	No
	RGB-D-SLAM (93)	2012	RGB	Optimisation	Dense	No	Yes
	ORB-SLAM (94)	2015	All types	Optimisation	Sparse	Yes	Yes
	ORB-SLAM2 (99)	2017	All types	Optimisation	Sparse	Yes	Yes
	OpenVSLAM (100)	2019	All types	Optimisation	Sparse	Yes	Yes
	UcoSLAM (101)	2019	All types	Optimisation	Sparse	Yes	Yes
Appearance-Based	DTAM (103)	2011	Monocular	-	Dense	No	No
	LSD-SLAM (104)	2014	Monocular	Optimisation	Semi-Dense	Yes	Yes
	SVO (108)	2014	Monocular	-	Sparse	No	No
	DSO (109)	2017	Monocular	-	Semi-Dense	No	No
	LDSO (110)	2018	Monocular	Optimisation	Semi-Dense	Yes	Yes
	Dynamic-DSO(111)	2020	Monocular	Optimisation	Semi-Dense	No	No
	KinectFusion (112)	2011	RGB-D	-	Dense	No	No
	RGB-DTAM (113)	2017	RGB-D	-	Semi-Dense	No	No
	ID-RGBDO (114)	2020	RGB-D	-	-	No	No
Semi-Direct	CPA-SLAM (115)	2016	RGB-D	Optimisation	Dense	No	Yes
	KDP-SLAM (117)	2017	RGB-D	Optimisation	Dense	No	Yes
	BundleFusion (116)	2022	RGB-D	Optimisation	Dense	Yes	Yes
	FSD-SLAM (118)	2022	All type	Optimisation	-	No	Yes

DSO, direct sparse odometry; FSD-SLAM, Fast Semi-Direct SLAM; LSD, large-scale semi-dense; ORB, oriented fast and rotated BRIEF; PTAM, parallel tracking and mapping; SLAM, simultaneous localization and mapping, Mono-SLAM, monocular Visual SLAM, RGB-D-SLAM, Red-Green-Blue-Depth-SLAM, DTAM, Dense Tracking and Mapping, LSD-SLAM, large-scale semi-dense (LSD) SLAM,SVO semi-direct visual odometry, DSO, direct sparse odometry, LDSO: Direct Sparse Odometry with Loop Closure ,KDP-SLAM,'keyframe-based dense planar SLAM, ID-RGBDO, Information-Driven Direct RGB-D Odometry, FSD-SLAM, fast semi-direct SLAM, CPA-SLAM, Consistent Plane-Model Alignment, KDP, Keyframe-based dense planar SLAM.

ICE-BA stands for incremental, consistent and efficient bundle adjustment developed by Liu et al. (124). It gives a solution accurately and efficiently compared to traditional solutions. It used a larger number of measurements to achieve higher robustness and accuracy. It is based on solving the global consistency problem to ensure the minimisation of the reprojection function and inertial constraint function during loop closure.

SVOGTSAM was proposed by Forster et al. (125) as a program that aims to develop a new theory for the pre-integration stage. It deals with the multiple structures of the rotation group. It operates on the generative scaling model in addition to the nature of the rotation noise and determines the expression for the maximum post-state estimator. It integrates the IMU model into an inertial pipeline under the unified factor graphics framework. Therefore, it is allowed to use the method of incremental-smoothing and the use of a structureless model for visual measurement, which increases computation speed by avoiding optimization via 3D points.

VI-DSO 'direct sparse visual-inertial odometry' is an extension of DSO that uses inertial information developed by Von Stumberg et al. (126). The objective of this algorithm is to find the position of the camera and sparse scene geometry by reducing an energy function that combines the photometric and IMU measurement errors. They used the 'dynamic marginalization' approach in order to achieve marginalisation adaptively.

VINS-Mono 'A monocular visual inertial system' is presented by Qin et al. (127). It is a robust and versatile approach based on a low-cost IMU and a single camera to determine the 6 degrees of freedom state of the system. The main contributions of this approach presented are a high-precision VIO measurement obtained by integrating IMU measurements and feature observations using a tightly correlated non-linear optimisation-based method. Integrating the module of loop detection with a tightly coupled formula that allows relocalisations with minimal computational cost to achieve global consistency, they optimised the pose graph for four degrees of freedom. This algorithm has been further developed in many research papers (128)(129).

PL-VIO, which is an acronym for point-line-visual inertial odometry, was proposed by He et al. (130). It is a strongly connected point-and-line-based monocular VIO system. Compared to dot features, lines provide more information about the environment's geometric structure. To determine the representative pressure of a 3D spatial line and the ease of calculation, Plucker coordinates and an orthogonal representation of the line are both employed. States are optimised by lowering a cost function, owing to the tightly and effectively integrated information between IMU and optical sensors.

Trifo-VIO (Trifo visual inertial odometry) proposed by Zheng et al. (131) utilised points and lines in a stereo VIO system with tightly coupled filtering. They create a novel technique for closing loops based on light filtering developed as EKF updates, which correctly repositions the sliding window now in use and keeps the filter active to detect loops. They make use of IMU data from the Trifo ironsides sensor, stereo camera data and the Trifo Ironsides dataset.

Co-Planar parametrisation for stereo-SLAM and VIO pipeline was proposed by Li et al. (132). By creating efficient and reliable parameters for co-planar points and lines that make use of particular geometrical restrictions, this method intends to increase the camera positioning's efficiency and accuracy. The pipeline comprises extracting 2D points and lines, forecasting planar areas with random-sample consensus (RANSAC) and outlier filtering. Two steps are used in the detection of RGB images: robust outlier filtering and the deployment of a NN for planar segmentation. They employ the smaller and more sparsely distributed Hessian matrix, which optimises BA, to determine new parameters for points and coplanar lines to unify the parameters.

Mesh-VIO (133). They devised a method for building a 3D mesh progressively by limiting its extent to the time horizon of VIO optimisation in order to get a representation of the topology of the environment. The 3D mesh offers a richer and lighter model that seeks to identify and enforce structural regularities in the optimisation problem.

ORB-SLAM3 (80). It is a comprehensive system that uses monocular, RGB-D and stereo cameras to perform visual SLAM, visual-inertial SLAM and multi-map SLAM. It is a visual-inertial SLAM system that utilises the maximum-a-posteriori (MAP) estimation method. In both large and small environments, indoors and outdoors, it has a reliable real-time result. Additionally, it has an accuracy that is 2 to 10 times better than earlier techniques. It is a multi-map system built on a cutting-edge method for position identification with better recall. The outcomes demonstrate the precision and reliability of the ORB-SLAM3 system.

HybVIO (134) is a hybrid approach that combines optimisation-based SLAM and filter-based inertial optical measurement (VIO) to estimate ego-motion. The contributions of this strategy include the development of a probabilistic inertial visual odometry (PIVO) methodology that can be used for monocular or stereo applications; the modelisation of the IMU bias in PIVO using the Ornstein–Uhlenbeck random walk approach and using improved and derived mechanisms for aberration detection, stability detection and feature path selection that take advantage of the special characteristics of the probabilistic framework. In real-time, this technology offers exceptional performance. Tab. 3 summarises the VIO methods in the order present in the text.

5. SLAM DL METHODS

A branch of machine learning called DL is based on artificial NNs. It has more than two layers built on algorithms that can be trained to process nonlinear data (Fig. 5). The learning field is characterised by supervised methods and unsupervised ways of learning. CNNs, recurrent neural networks (RNNs) and other designs are used in DL for a variety of tasks. The training process in the supervised learning method requires supervision and labelled data. Its objective is to train the model so that, given fresh data, it can forecast the outcome. An unsupervised learning method uses unlabelled data without the need for supervision during training. Their main objective is to find hidden patterns and useful insights from the unknown dataset.

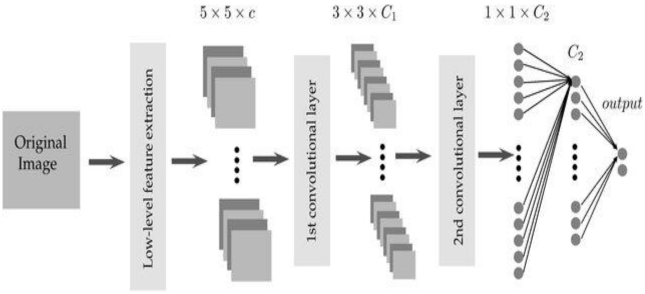

Fig. 5. DL architecture

Numerous papers filed by researchers in the SLAM field have combined DL with visual SLAM to address issues and create algorithms. This section outlines the applications of DL across several SLAM components.

Tab. 3. Comparison of VIO methods

Name	Year	Back-End-Approach	Camera Type	Fusion Type	Mapping	Loop closing	Relocalization
OKVIS (122)	2014	Optimisation-base	Monocular	Tightly coupled	Sparse	No	No
ROVIO (119)	2015	Filtering based	Monocular	Tightly coupled	Sparse	No	No
MSCKF-VIO (120)	2018	Filtering based	Monocular/stereo	Tightly coupled	Sparse	No	No
SVOGTSAM (125)	2017	Optimisation-base	Monocular	Tightly coupled	-	No	No
Maplab (123)	2018	Filtering based	Monocular	Tightly coupled	Dense	Yes	No
ICE-BA (124)	2018	Optimisation-base	-	-	-	Yes	No
VI-DSO (126)	2018	Optimisation-base	Monocular	Tightly coupled	Sparse	No	No
VINS-Mono (127)	2018	Optimisation-base	Monocular	Tightly coupled	Sparse	Yes	Yes
PL-VIO (130)	2018	Optimisation-base	Monocular	Tightly coupled	-	No	No
Trif-VIO (131)	2018	Filtering based	Stereo	Tightly coupled	-	Yes	No
Co-Planar (132)	2020	Optimisation-base	Stereo	Tightly coupled	Dense	No	No
Mesh-VIO (133)	2019	Optimisation-base	Stereo	Tightly coupled	Dense	No	No
ORB-SLAM3 (80)	2021	Optimisation-base	All	-	Sparse	Yes	Yes
HybVIO (134)	2022	Optimisation-base	Monocular/stereo	Loosely coupled	Sparse	Yes	No

ROVIO, Robust visual inertial odometry; ICE-BA, innovation covariance estimation-bundle adjustment; Mesh-VIO, MSCKF-VIO, multi-state constraint Kalman filter-visual-inertial odometry; OKVIS, open Keyframe-based visual inertial SLAM; ORB, oriented fast and rotated BRIEF; SLAM, simultaneous localisation and mapping; Trifo-VIO, Trifo visual inertial odometry; VIO, visual–inertial odometry; VI-DSO, direct sparse visual-inertial odometry; PL-VIO, point-line-visual inertial odometry; VINS-Mono, A Robust and Versatile Monocular Visual-Inertial State Estimator; HybVIO, hybrid visual–inertial odometry.

5.1. Initialisation

One of the most crucial deficiencies is the inability to estimate scale and determine depth during the initialisation phase of monocular visual SLAM. The issue of depth has been addressed in a number of works, some of which include optical flow, ego-motion, scale ambiguity and drift based on DL. We mention some of the most important works : ADAADepth (135), CNN-SLAM (136), Code-SLAM (137), DeepVO (138), UnDeepVO (139), D3VO (140), GeoNet (141), L-VO (142) and Un-L of depth (143).

The algorithm for creating the depth and disparity map in stereovision consists of four steps: feature extraction, feature matching across pictures, computation of disparity and disparity refining and post-processing. Researchers concentrated on estimating these stages using DL techniques. The primary goal of DL in stereo matching is to substitute learned features for manually specified characteristics: (144)(145). Mayer et al. (146) used DL to estimate the disparity, scene flow and optical flow. The works that have been done in Refs (147)(148) used DL to estimate the depth for disparity maps. Song et al. (149) suggested a technique end-to-end that allows to predict the edge map and disparity.

5.2. The front-end enhanced by DL

The two key components of this step are feature point extraction from the successive photographs and VO. The feature points enable the estimation of the camera movement alone without taking into account the entire map. The three-dimensional pose translation and three-dimensional pose rotation make up the estimation of the 6-DoF motion state. This motion estimation is applied to feature points throughout the tracking task using a RANSAC-based matching procedure. To estimate the homogeneous transformation between frames, as well as to ascertain the camera's present location and the environmental characteristics, matching is used.

Shao et al. (150) created a faster region-based convolutional neural network (Faster-R-CNN)-based semantic filter to address the issue of outliers in RANSAC-based F-matrix calculations. The semantic filter's training phase relies on semantic patches created by inliers, which enables various picture regions to define various semantic labels. The approach improves and increases the precision of F matrix calculations.

Zhang et al. (151) focused on the use of visual semantic information in the problem of camera localisation. They suggested a coarse-to-fine strategy in the visual localisation method and created a visual semantic database based on a deep-learning algorithm.

The approach based on integrating DL and machine learning with 2D-SLAM grid maps was proposed by Lin et al. (152) to estimate 2D object segmentation, feature extraction and pattern identification. DL is used by Wang et al. (153) to complete monocular VO in a comprehensive manner. The stances are calculated based on the actual scene. Deep neural networks (DNNs) understand the intricate dynamic motion of image sequences to do sequence-to-sequence posture estimation. By combining the RGB-D SLAM with optical flow-based feature tracking, Li et al. (154) improved the SLAM algorithm. To achieve the function of object detection, they combined 101 layers of deep residual networks (ResNet) with region-based fully convolutional networks (R-FCN).

V-SLAM-CNN (155): Current systems combine DL to automate surgical instrument and workflow identification in order to decrease surgical problems and ensure correct performance. In this study, visual SLAM and Mask R-CNN are combined. They employ V-SLAM for object detection, drawing on geometry data for area recommendations and CNN for object recognition, classifying images using semantic data, and combining these techniques into a single end-to-end training assignment. They are based on visual characteristics and spatiotemporal data gathered from video. By substituting a region proposal module (RPM) for the region proposal network (RPN) in mask R-CNN, bounding boxes are placed precisely, and the need for annotations is decreased. DVS-SLAM: A visual semantic map in a dynamic situation is called a dynamic visual semantic SLAM (156). They employed SSD-MobileNetV2 lightweight DL to obtain the 2D data.

Some studies use supervised or unsupervised learning techniques to estimate the absolute 6-DoF posture or the relative transformation matrix when implementing an end-to-end VO system. The deep convolutional generative adversarial networks (GANs)-based unsupervised learning framework GANVO (157) predicts 6-DoF posture camera motion and a monocular depth map of the scene from unlabelled RGB image sequences. A supervised monocular VO system is called DL_Hybrid (158). It is based on recovering camera trajectory and estimating 6-Dof posture frame-by-frame. First, they concentrate on the DL_Hybrid VO system overview. The dense optical flow map between picture frame pairs is then estimated using a DL NN called DenseFlowNetwork, and the dense depth map per-frame is extracted using a different DL NN called DenseDepthNetwork. Finally, the true monocular scale-estimation methodology is applied frame-by-frame as we describe the hybrid 2d–2d and 3d–2d posture-estimation approach paired with optical flow map and depth map.

Liang et al.'s (159) successful direct sparse VO approach is called SalientDSO. It blends DSO with semantic data in the form of visual saliency. SalientDSO is based on a deep-learning visual saliency and scene-analysis method that selects a feature for accurate and reliable VO. Their contributions help to present a framework of indoor VO in which the features are selected based on a visual saliency map. The authors suggested a method for filtering the saliency map based on scene parsing. A DL technique called GCNv2, which is an extension of the 'Geometric Correspondence Network', depends on a network created by Tang et al. (160) to identify the salient features and descriptors. A binary descriptor vector serves as the ORB feature in GCNv2. In addition to having more computational efficiency than GCN, GCNv2 also maintains accuracy levels comparable to GCN, which results in observable advancements in movement estimation. They used feature vector binaries in the training phase, which significantly accelerated matching.

SuperPoint (161), a self-supervised system for training to detect and describe interest points for the issues of a large number of multiple-view geometries, is used to identify and describe points of interest. A complete CNN is the SuperPoint. It operates on full-size images, producing the interest point detection with fixed-length descriptions in a single forward pass Kwang. et al.'s 'Learned Invariant Feature Transform' (LIFT) was proposed (162). The detector, orientation estimator and descriptor are the three CNNs-based components that make up this system.

SIVO(semantically informed visual odometry and mapping) (163) is founded on a system that chooses which feature to use for V-SLAM. It incorporates NN uncertainty and semantic segmen-

DOI 10.2478/ama-2024-0049

tation into the feature-selection procedure. Since the new feature is added with feature entropy classification from the Bayesian NN, the approach finds the spots offering the biggest Shannon entropy drop between the entropy of the current state and the entropy of the shared state.

To extract the binary visual feature descriptors with triplet loss, even distribution loss, correlation loss and quantisation loss, Gu et al. (164) created the DBLD-SLAM 'deep binary local descriptor'. They create a CNN model with four fundamental loss functions to extract binary visual feature descriptors from picture patches: adaptive scale loss, even distribution loss, quantisation loss and correlation loss.

Based on this learned deep binary feature descriptor, which has the same structure as the ORB descriptor, a monocular SLAM system called DBLD-SLAM is built, with the ORB descriptor replaced by conventional ORB-SLAM. They also train Bag of Words to recognise loop closures visually.

The foundation of ORBDeepOdometry (165) is a method for integrating DL with pipeline methodology to tackle the monocular VO problem. It models sequential data by stacking multiple deep LSTMs, feature extraction ORB and dimensionality reduction based on CNN.

It was suggested to use the 'Criss-Cross Network' (CCNet) (166) to obtain contextual information for the entire image. It is developed by utilising the criss-cross recurrent attention module to get the best results in benchmarks dependent on segmentation, such as Cityscapes, ADE20K and COCO. A general framework called MonoGRNet (167) is used to learn how to detect monocular 3D objects based on geometric reasoning, the observable 2D projection and the depth dimension that is not being seen. This method splits the job into four smaller tasks – 2D object identification, instance-level depth estimation, projection 3D centre estimation and local corner regression – and uses the network to perform each of them simultaneously.

Tab. 4 summarises the front-end methods enhanced by DL in the order they appeared in the text.

5.3. Back-end enhanced by deep learning

This step aims to enhance this estimation through tasks involving localisation, optimisation and loop closure.

5.3.1. Optimisation

The global optimisation process aims to maintain the geometric consistency of the full map. It has been made for localisation and mapping tasks.

To estimate the motion, the sequence-to-sequence learning algorithm VINet (168) was developed. It is supported by optical and inertial sensors. For the VIO, it is an end-to-end system that is completely trainable. The authors suggested a method for training the architecture's parameters as well as a design for recurrent networks.

A brand-new frame-to-frame estimation technique called Deep_VO (169) makes use of CNN to forecast camera motion. The best visual feature and the best estimator for visual ego-motion estimation are both learned using the CNN architecture.

Tab. 4. Comparison of front-end methods

Name	Year	Architecture/method	Main contribution
LIFT (162)	2016	-CNNs	-Learning invariant features
Faster-R-CNN (150)	2020	-CNN -Semantic filter	-It solves the outlier problem in F-Matrix computations based RANSAC.
A 3D Semantic Visual SLAM (156)	2021	- Mask R-CNN/MySQL - It creates a semantic database based on the information contained in the object.	-It is beneficial for localisation. -The accuracy and efficiency of the localisation.
V-SLAM-CNN (155)	2022	- Mask R-CNN. - It combines the greatest features of both worlds, such as (1) object detection using vSLAM and (2) CNN for identifying objects. -Spatio-temporal information.	-Concentrating on geometric data for suggested regions. -Concentrating on semantic data for picture classification and merging them into a single, collaborative, end-to-end training procedure.
DVS-SLAM (156)	2021	-Multi-view geometry and region growing algorithm. -SSD-MobileNetV2 lightweight DL. -Colour bumpy supervoxel clustering algorithm.	-Creating a visual semantic map. Removing dynamic feature points will improve localisation accuracy. -Get 2D data. -Achieve the extraction of 3D target information.
GANVO (157)	2019	-unsupervised learning framework. -GANs	-Predicts 6-DoF pose camera motion and camera depth.
DL_Hybrid (158)	2021	-Hybrid 2D–2D and 3D–2D localisation theory. -DNN 'DenseDepthNetwork' -DFN 'DenseFlowNetwor'	-One-frame-at-a-time estimation of a six-degree-of-freedom pose and camera trajectory recovery are possible. -Accurate key points extracted from each frame even in harsh scene conditions, and the system performs well even in situations where motion is restricted to the camera, such as when it is rotating or stationary. -Large-scale displacement motion of the camera is also a possibility.
SalientDSO (159)	2019	-High semantic information -CNNs+VO	-Drive feature selection for visual saliency. -Offers a technique for saliency map filtering depending on scene parsing.

GCNv2 (160)	2019	-Geometric Correspondence Network -Incorporates feature vector binarisation into training.	-Offering remarkable gains in motion estimation compared to similar DL-based feature extraction algorithms, while dramatically lowering inference time. -The matching is substantially accelerated.
Superpoint (161)	2018	-FCN -Homographic adaptation -Multi-scale -Multi-homograph approach.	-Self-supervised interest features. -A deep SLAM frontend. -Designed for real time.
SIVO (163)	2019	-BNN	-Allow for long-term localisation
DBLD-SLAM (170)	2021	-CNN	-Using four important loss functions, extract binary visual feature descriptors from picture patches. -Train Bag of Words to recognise loop closures visually.
ORBDeepOdometry (165)	2019	-CNN	-For modelling the sequential data, the authors propose using an ORB-based feature extractor, CNN-based dimensionality reduction, and stacking several deep LSTMs.
CCNet (166)	2019	-Mask R-CNN+ResNet-101	-Acquiring such contextual information in a more efficient and effective manner.
MonoGRNet (167)	2021	-End-to-End network +Joint geometric loss	-Detecting 3D objects in monocular pictures.

Faster-R-CNN, faster region-based convolutional neural network; DNNs, Deep neural networks ; R-FCN, region-based fully convolutional networks ; CCNet, criss-cross network; CNN, convolutional neural network; DL, deep learning; DNN, deep neural network; DVS, dynamic visual semantic SLAM ; Faster-R-CNN, faster region-based convolutional neural network; RPM, region proposal module; RPN, region proposal network; FCN, fully convolutional network; GANs, generative adversarial networks; LIFT, learned invariant feature transform; LSTM, long-short term memory; ORB, oriented fast and rotated BRIEF; RANSAC, random-sample consensus; SIVO, semantically informed visual odometry; SLAM, simultaneous localisation and mapping; GANVO, generative adversarial networks visual odometry; DL_Hybrid, deep learning Hybrid; GCNv2 ,Geometric Correspondence Network; DBLD-SLAM, deep binary local descriptor; CCNet ,Criss-Cross Network; MonoGRNet, monocular geometric reasoning netwoek, SalientDSO, Salient Direct Sparse Odometry

A NN that is cognizant of geometry is SFM-Net (171), a DL technique that is self-supervised and works with videos to gauge motion. Scenes, object depth, camera motion and 3D object translations and rotations are the categories used to categorise frame-to-frame pixel motion. The program makes predictions about object motion, depth and masks.

The Konda approach (172) was used to predict the direction and velocity changes from visual input using an end-to-end DL architecture. Based on learning rules and a single computational model, the extraction of depth from visual motion and information from odometry are both possible.

DeepVO (173) is a monocular VO that makes use of a cutting-edge end-to-end architecture and a deep recurrent convolutional neural network. The primary goal is to directly predict portions from raw RGB images. To restore the absolute scale, no prerequisite information or criteria are required. The RCNN architecture enables the DL-based VO technique to be generalised to entirely new contexts by using the representation of the geometric features learned via the CNN. Deep recurrent neural networks (DRNNs) are used to automatically learn the complex motion dynamics of image sequences. Tab. 5 summarises optimisation methods by DL in the order they appear in the text.

5.3.2. Relocalisation

When tracking is unsuccessful, the task of relocalisation seeks to increase the accuracy of the camera posture. In this part, we outline some DL-based research projects that try to solve this issue.

VidLoc (174): It is a recurrent model that tries to reduce pose estimate error and accomplish 6-Dof localisation of video. The

authors created a spatio-temporal model for global localisation and utilised CNN to predict the scene coordinates. A technique for calculating the instantaneous covariances of position estimations of the input RGB-D picture was implemented into their network.

YOLO (175): It is a method that enhances relocalisation through the use of semantic data. It presents the object 'YOLO' as an array and classifies it using a DL NN using high-level features. This array makes it possible to reject weak candidates and shorten the computation time for the relocalisation tasks.

The research conducted on indoor relocalisation entitled Dual-Stream-CNN (176). It seeks to offer a dual stream CNN-based indoor relocalisation system that takes both colour and depth images as inputs. The suggested technique effectively illustrated the system's robustness in difficult circumstances like large-scale, dynamic, fast-moving and nighttime settings.

Outlier-aware neural tree (177): It is a brand-new outlier-aware neural tree that links decision trees and DL techniques. It uses only stable and secure regions of the surroundings to establish point correspondences for an accurate estimation of camera position. The approach also has decision trees' broad framework characteristics. It is built around three main sections: a hierarchical space section over the indoor scene to create a decision tree; a deep classification network used to better comprehend the 3D scene and an outlier rejection module used to filter dynamic points during the hierarchical routing process.

SIR-NET (178): The CNN is used by the authors to build a framework for relocalisation. It can be trained end-to-end and is unaffected by the environment. Using the backpropagation of relocalisation faults to both processes enhances retrieval and matching to have the best accuracy in relocation. By selecting pixels based on uncertainty, they can accelerate the unit-matching inference without compromising the accuracy of relocation.

LSTMFCN (long-short term memory fully convolutional network) (179): The research was designed to compare two DL-based algorithms to address the issue of single-picture relocalisation. The first uses a DNN end-to-end to directly understand the relationship between an image's position and its mapping. The LSTMFCN algorithm is the second. The LSTMFCN method is distinguished by a much larger receiving range, which avoids the problem of aperture and makes it robust to partial blockages and moving objects. It is composed of a fully convolutional network (FCN) that performs feature extraction and a long-short term memory (LSTM) that is a pooling layer to group information across the image.

xyzNet (180) is a light CNN. It is a hybrid technique; to relocalise the camera pose from a single RGB image, the researchers merged the geometric method with the machine learning method. The most precise camera position calculation is provided by the geometric information about 2D–3D correspondences, which also eliminates uncertain predictions. Tab. 6 summarises relocalisation methods by DL in the order appearing in the text.

Tab. 5. The optimisation methods of DL SLAM

Name	Year	Architecture/method	Main contribution
Konda Approach (172)	2015	-End-to-End + DL	-Using VO, predict velocity and direction.
Deep_VO (169)	2016	-CNN	-Estimate scale and motion robustly
SFM-Net [194]	2017	-Self-supervised GNN	-A DNN that predicts pixel-wise depth from a single frame as well as camera motion, object motion and object masks from a pair of frames.
VINet (168)	2017	-Sequence-to-sequence + RCNNs	-Offer a unique recurrent network design and training approach to optimise model parameter training
DeepVO (173)	2017	End-to-End + RCNNs	-Presents an RCNN architecture that allows the DL-based VO technique to be generalised to whole new settings by using the CNN's geometric feature representation.

VINet, Visual-Inertial Odometry; DBLD-SLAM , Binary Local Descriptor SLAM; CNN, convolutional neural network; DL, deep learning; DNN, deep neural network; SLAM, simultaneous localisation and mapping; Deep_VO, Deep- visual odometry; SfM-Net: Learning of Structure and Motion.

Tab. 6. The relocalisation methods of DL SLAM

Name	Year	Architecture/method	Main contribution
VidLoc (174)	2017	-CNN	-Attempts to decrease pose estimation error and achieve 6-D of video localisation.
Dual-Stream-CNN (176)	2018	-CNN	-Improves the relocalisation accuracy. -Investigates depth image encoding techniques and proposes a fresh approach termed minimised normal.
LSTMFCN (179)	2018	-FCN	-Avoids the problem of aperture and makes it robust to partial blockages and moving objects. -Suggest refining as a way for improving training model performance.
xyzNet [197]	2018	-Light CNN (xyzNet)	-Geometric information concerning 2D-3D correspondences enables the elimination of unclear predictions and the creation of more precise camera poses. -The accuracy and the performance of our solution on diverse datasets as well as the power to solve difficulties involving dynamic scenario.
SIR-NET (178)	2019	-CNN	-This system simultaneously optimises retrieval and matching tasks to maximise relocalisation accuracy.
Outlier-aware Neural tree (177)	2021	-DL + decision tree approaches.	-Relocalisation in dynamic indoor environments. It achieves robust neural routing through space partitions.
YOLO (175)	2022	-YOLO - Semantic data.	-Rejects unqualified candidates. -Shortens the computation time for the relocalisation tasks.

LSTMFCN, long-short term memory fully convolutional network; SIR-Net : Scene-Independent End-to-End Trainable Visual Relocalize; CNN, convolutional neural network; DL, deep learning; FCN, fully convolutional network; LSTMFCN, long-short term memory fully convolutional network; SLAM, simultaneous localisation and mapping.

5.4. Loop-closure detection

An essential function of the SLAM system is the loop-closure process, which lowers the drift that has collected over time. There are a number of stable, efficient and light-weight DL loop-closure techniques. Traditional feature-point extraction algorithms are used in loop-closing detection methods. The majority of algorithms made use of hand-crafted features and bags of visual words (BoVW).

Wu et al. (181) presented the loop-closure detection for visual SLAM derived from the SuperPoint Network. The SuperPoint NN, which is intended to concurrently recognise points of interest and their associated descriptors, was utilised by the authors to learn inner structures from raw data. The similarity of the image is determined using cosine similarity. Merrill and Huang (182) suggested that for the visual close-loop, an unsupervised automatic encoder network architecture is used. The Histogram of Oriented Gradients (HOG) technique provides geometric data and illumination invariance, which forces the encoder to reconstruct the HOG descriptor rather than the original image.

The resulting models do not require labelled training data or environment-specific training; instead, they extract strong to extreme changes in appearance directly from the raw photos.

Utilising the feature obtained through unsupervised DL can increase the loop-closure detection method's accuracy. PCANet, a deep cascade network, was utilised by Yifan Xia, et al. (183) to extract features as image descriptions. Principal component analysis (PCA), binary hashing and block-wise histograms are the three components that make up the PCANet, a straightforward DL network. The PCA and deep CNN were used by Dai et al. (184) to execute a closed-loop detection process and to scale down the extracted feature dimensions. It is important to note the low detection accuracy of combination approaches. By using the pre-trained ResNet34 model to extract features, this issue is resolved. To reduce the dimension of the features, they then used Kernel PCA (KPCA) on the extraction features.

Seq-CAL, introduced by Xiong et al. (185), is a lightweight sequence-based unsupervised loop-closure-detection approach that integrates sequence information with PCA to achieve good detection accuracy and faster detection times. They reduced descriptor dimensions while retaining sufficient expressive power using PCA. An algorithm for lightweight loop-closure detection and product quantisation (PQ) was created by Huang et al. (186). By using the pre-trained CNN model, SSE-Net, they were able to extract the image's deep visual and semantic features and obtain a vector of feature descriptions. The loop is demonstrated by locating and matching the most comparable pair of candidate frames after PQ and encoding.

Zhu and Huang (187) developed fast and robust visual loop-closure detection using CNN. The authors improved the pre-training model using the Lite-shuffleNet network by extracting the semantic data and depth of the image to derive the feature descriptor, measuring the cosine similarity, choosing the best candidate frames and judging whether to loop.

To represent a picture, Jiayi Ma et al. (188) proposed the fast and robust loop-closure detection through the convolutional auto-encoder and motion consensus. To extract the features, they used a compact convolutional auto-encoder (CAE) network. They trained the network to offer the data of the visual loop closure-detection procedure using the deep perceptual similarity loss function. The principle of place sequence division is the foundation for the phase of loop-closure detection. The CAE network's mapped coding space is employed in the query job to determine which historical image is most similar to the current query image. They introduced an image-to-sequence section method based on place sequence division and distance-weighted voting for loop-closing selection.

In some works, the loop-closure process is addressed using a hybrid DL architecture (HDLA). To enhance spatial awareness and loop-closure detection using a hybrid CNN, Cai et al. (189) devised an effective way to produce high-level semantic image features. It is built using ResNet-18 and optimised with the split–transform–merge concept as well as the squeeze-and-excitation structure, allowing for the compensation of the network's ability to represent pictures without sacrificing performance. To save the time needed to measure the distance between deep semantic features, the authors provided a straightforward method of reducing dimensions during their fitting. Liu et al. (190) proposed a method for developing high-level semantic features that are resistant to changes in both viewpoint and lighting. The architecture of the network is a hybrid ConvNet network tuned to handle robust and real-time feature extraction. Although it shares AlexNet's fundamental structure, it functions best when the appearance is drastically altered. Shi and Li (191) employed a YOLOv4 model with an improved loss function to find the target in the camera-obtained images. The locality sensitive hash function is used to reduce the high-dimensional data dimension, and the cosine distance is used to detect loops.

Local3Ddeep descriptors (L3Ds) (192) is a method for loop detection that measures the overlap. It saves the loop candidate point cloud by their estimated relative positions and then determines the error metric between points that mutually correspond to the nearest neighbour descriptors. This technique enables precise loop recognition in the event of slight overlaps in 6-DoF estimation.

LoopNet (193) aims to discover important landmarks for the scene to focus on without being distracted by scene fluctuations. It is a plug-and-play algorithm. Additionally, it is a multi-scale attention-based Siamese convolutional model that learns feature embeddings that emphasise the distinguishable objects in the scene rather than comprehensive features.

MAQBOO (194) is a sophisticated algorithm. It increases the effectiveness of pre-trained models to boost visual recall and use them in real-time with multi-agent SLAM systems. In comparison to a high descriptor, the suggested approach achieves equivalent accuracy in a low descriptor dimension. Tab. 7 summarises loop-closing methods by DL in the order appearing in the text.

6. PROBLEMS AND CHALLENGES

We can infer from this study that the visual SLAM system changes over time. Every aspect of its architecture and computer vision tasks confronts issues. The researchers applied environmental perception research to address these problems, enhancing V-SLAM and enhancing resilience in real-world contexts caused by variations in lighting, dynamic objects and shifts in viewpoint. The use of low-level sensors has been found to be another significant SLAM issue. Numerous sources of ambiguity and problems must be solved in order to get a trustworthy SLAM. The three fundamental problems are temporal complexity, uncertainty and correspondence, sometimes known as data association. Classical difficulties and perception problems can be distinguished as issues in the development of visual SLAM. The classical problems result from algorithms whose tasks rely on computer vision-related issues. Among the most common problem:

- Estimation of intrinsic parameters is set before using visual SLAM systems because camera calibration is done before visual SLAM systems and is adjusted during the V-SLAM process.
- Pure rotation is a problem in the field of computer vision due to the inability to observe disparities in the monocular visual SLAM during purely rotational motion. To address this problem, several projection models were used (195)(196).
- The map initialisation presents the first estimation of the localisation and is the main task for the rest of the process of visual SLAM. Among the things that make a preliminary map accurate is to make the baseline wide.
- The scale ambiguity is a particular problem with monocular SLAM. It lies in their geometric inability to get the information of absolute scale about the trajectories and environment.
- Fusion of multi-sensors: The use of a single sensor in the SLAM process generates several limitations. The fusion of multiple sensors can provide rich data resulting in a more accurate and robust system. However, sensor fusion can cause problems on several levels.

Classical visual SLAM needs to address several issues, computation for large-scale environments, distortion of movement and achieving compatibility between accuracy and real-time process relationship.

- Perception problems give rise to algorithms that improve performance in all tasks and seek to implement a robust and precise system that confuses perception with optimization.

DL offers practical precision and robust object detection, and prediction, which can understand the scene this improves the processes of visual SLAM.

– Computation speed: DL offers many advantages in recognition. However, the low computational speed remains one of the most important problems. This makes dynamic SLAM not usable in embedded V-SLAM systems.
– Computation complexity: This problem is generated by incorporating the object-detection modules.
– Future research into SLAM perception will focus on solutions capable of handling real-world conditions and lighting changes and developing and improving tasks for performing visual SLAM in real-time scenes.

Tab. 7. The loop-closing methods of DL SLAM

Name		Year	Architecture/method	Main contribution
Light_unsupervised_D (182)		2018	-Unsupervised deep NN	-Efficient, and robust place recognition. -The visual loop closure that is both reliable and compact.
SuperPoint (181)		2019	-SuperPoint	-Simultaneously identifies interest spots and related descriptions. -By computing the cosine similarity of the respective vectors, it determines how similar the pictures are to one another.
HDLA	(189)	2018	-ResNet+ split-transform-merge strategy + squeeze-and-excitation structure.	-Produces high-level semantic picture features for better loop-closure detection and location recognition. -A simple while fitting dimension reduction algorithm, particularly useful for lowering the time required to estimate distance between deep semantic features.
	(190)	2019	-Hybrid CNN	-Provides high-level semantic picture characteristics specifically for loop closure detection. -By using locality-sensitive hashing (LSH) and employing the nearest neighbour of a single image to search for the key frame using the cosine similarity score, you may guarantee the real-time performance of loop closure detection.
YOLOv4 (191)		2020	-YOLOv4+ optimised loss function	-High-dimensional data can have its dimensions reduced by using the Locality Sensitive Hash function. -The cosine distance is used to determine the loop.
Local3DDeep descriptors (192)		2022	-L3Ds	-After registering the loop candidate point cloud by its estimated relative posture, computes the metric error between points that correspond to mutually-nearest-neighbour descriptors. -In the event of tiny overlaps, properly recognise loops and estimate 6-DoF postures.
LoopNet (193)		2022	-Plug-and-play model+LoopNet,	-Identifies similarities across scenes by identifying essential key landmarks to focus on while being unaffected by scene differences.
MAQBOO (194)		2022	-Multiple AcQuisitions of perceptiBle regiOns for priOr Learning	-Uses spatial information to improve the recall rate in image retrieval on pre- trained models

6-DoF, six degrees of freedom; CNN, convolutional neural network; DL, deep learning; HDLA, hybrid deep learning architecture; L3Ds, local 3D deep descriptors; NN, neural network; SLAM, simultaneous localisation and mapping; BoVW,bags of visual words; HOG,The Histogram of Oriented Gradients; PCA, Principal component analysis; CAE, convolutional auto-encoder; MAQBOOL, Multiple AcQuisitions of perceptiBle regiOns for priOrLearning.

7. CONCLUSION

The most-significant fundamental techniques and problems related to visual SLAM are highlighted in this paper's presentation of the emergence and development phases of the technology. The evolution of SLAM in this study was broken down into three phases: SLAM probabilities, vision SLAM and SLAM perception. Each phase tries to find solutions to the issues raised in the phase before it, while also fostering competency in visual SLAM. The study attempted to demonstrate the benefits, contributions and restrictions of each of the algorithms that were offered.

A lucrative field that also advances visual SLAM is created by the combination of DL methods with machine visions. DL has made numerous advances in recent years, notably for tasks like image analysis, processing and decision-making, which performs with great accuracy and speed. It is possible to use DL to enhance various SLAM tasks, including visual odometry, optimisation, relocalisation and loop closure. DL techniques are utilised in visual SALM to speed up computation and are crucial for fully comprehending the complex scene that is being viewed.

REFERENCES

1. Hans P Moravec. Obstacle Avoidance and Navigation by a Seeing Robot Rover in the Real World. SPittsburgh, Penna Carnegie-Mellon Univ Robot Institute. 1980.
2. D. Nister ON and JB. Visual odometry. Proc 2004 IEEE Comput Soc Conf Comput Vis Pattern Recognition,004 CVPR 2004. Washington DC. USA. 2004;1:I–I.
3. Longuet-Higgins H. A computer algorithm for reconstructing a scene from two projections. Nature. 1981;293:133–5.
4. CG Harris JMP. 3d positional integration from image sequences. Image Vis Comput Sci Direct. 1988;6(2):87–90.
5. Twinanda AP, Shehata S, Mutter D, Marescaux J, De Mathelin M and NP. Endonet: a deep architecture for recognition tasks on laparoscopic videos. IEEE Trans Med Imaging. 2016;36(1):86–97.
6. Bodenstedt S, Ohnemus A, Katic D, Wekerle AL, Wagner M, Kenngott H, Muller-Stich B, Dillmann R and SS. Real-time image-based instrument clas- sification for laparoscopic surgery. 2018. preprint arXiv:1808.00178.
7. Yang IC, Chen S. Precision cultivation system for greenhouse production. In Intelligent Environmental Sensing. Springer Berlin/Heidelberg. Ger Google Sch. 2015;191–211.

8. Borges DL, Guedes ST, Nascimento AR, Melo-Pinto P. Detecting and grading severity of bacterial spot caused by Xanthomonas spp. in tomato (Solanum lycopersicon) fields using visible spectrum images. Comput Electron Agric. 2016;149–159.
9. Liu X, Zhao D, Jia W, Ji W, Ruan C, Sun Y. Cucumber fruits detection in greenhouses based on instance segmentation. IEEE Access. 2019;139635–139642.
10. Asdemir S, Urkmez A, Inal S. Determination of body measurements on the Holstein cows using digital image analysis and estimation of live weight with regression analysis. Comput Electron Agric. 2011;76, 189–197.
11. Norton T, Chen C, Larsen MLV, Berckmans D. Precision livestock farming: Building 'digital representations' to bring the animals closer to the farmer. Anim 1. 2019;3:3009–3017.
12. Chou WC, Tsai WR, Chang HH, Lu SY, Lin KF, Lin P. Prioritization of pesticides in crops with a semi-quantitative risk ranking method for Taiwan postmarket monitoring program. J Food Drug Anal. 2019;27: 347–354.
13. Kalman RE. A new approach to linear filtering and prediction problems. Trans ASME. J Basic Eng. 1960;82(1):35–45.
14. Julier SJ, Uhlmann JK. A counter example to the theory of simultaneous localization and map building. Proc 2001 ICRA IEEE Int Conf Robot Autom (Cat No01CH37164). Seoul. Korea (South). 2001; 4: 4238-4243. doi:101109/ROBOT2001933280
15. Gamini Dissanayake MWM, Newman P, Clark S, Durrant-Whyte HF, Csorba M. A solution to the simultaneous localization and map building (SLAM) problem. IEEE Trans Robot Autom. 2001;17(3):229–41.
16. Smith R, Self M, Cheeseman P. A stochastic map for uncertain spatial relationships. Mach Intell Pattern Recognit [Internet]. 1988;5:435–61.
Available from: http://portal.acm.org/citation.cfm?id=57472
17. Moutarlier P, Chatila R. An Experimental System for Incremental Environment Modelling by an Autonomous Mobile Robot. LAAS-CNRS 7. Ave du Colonel Roche 31077 Toulouse.
18. Jazwinski AH. Stochastic Processes and Filtering Theory. 1970;64.
19. Zhu J, Zheng N, Yuan Z, Zhang QXZ and YH. A SLAM algorithm based on the central difference kaiman filter. IEEE Intell Veh Symp Xi'an. China. 2009;123–8.
20. Jiang X, Li T, Yu Y. A novel SLAM algorithm with Adaptive Kalman filter. ICARM 2016 Int Conf Adv Robot Mechatronics. 2016; 107–11.
21. Tian Y, Suwoyo H, Wang W, Mbemba D, Li L. An AEKF-SLAM Algorithm with Recursive Noise Statistic Based on MLE and EM. J Intell Robot Syst. 2020;97:339–55.
22. Julier SJ, Uhlmann JK. New extension of the Kalman filter to nonlinear systems. Proc Vol 3068, Signal Process Sens Fusion, Target Recognit VI. 1997;3068.
23. Havangi R. Robust SLAM: SLAM base on H ∞ square root unscented Kalman filter. Nonlinear Dyn. 2016;83(1):767–79.
24. Bahraini M, Bozorg M, Rad A. A new adaptive UKF algorithm to improve the accuracy of SLAM. Int J Robot. 2019;5(1):35–46.
25. Bahraini MS. On the Efficiency of SLAM Using Adaptive Unscented Kalman Filter. Iran J Sci Technol Trans Mech Eng [Internet]. 2020;44:727–35.
Available from: https://doi.org/10.1007/s40997-019-00294-z
26. Tang M, Chen Z, Yin F. SLAM with Improved Schmidt Orthogonal Unscented Kalman Filter. Int J Control Autom Syst. 2022;20(1598–6446):1327–35.
27. Liu D, Duan J and HS. A Strong Tracking Square Root Central Difference FastSLAM for Unmanned Intelligent Vehicle With Adaptive Partial Systematic Resampling. EEE Trans Intell Transp Syst. 2016;17(11):3110–20.
28. Maybeck PS. Stochastic Models, Estimation, and Control. Acad Press. 1979;1:282.
29. Garritsen T. Using the Extended Information Filter for Localization of Humanoid Robots on a Soccer Field. 2018;1–25.
30. Thrun S, Liu Y, Koller D, Ng AY, Ghahramani Z, Durrant-Whyte H. Simultaneous localization and mapping with sparse extended information filters. Int J Rob Res. 2004;23(7–8):693–716.
31. Walter MR, Eustice RM, Leonard JJ. Exactly sparse extended information filters for feature-based SLAM. Int J Rob Res. 2007;26(4):335–59.
32. He B, Liu Y, Dong D, Shen Y, Yan T, Nian R. Simultaneous localization and mapping with iterative sparse extended information filter for autonomous vehicles. Sensors (Switzerland). 2015;15(2): 19852–79.
33. Zhang H, Liu Y, Tan J, Xiong N. RGB-D SLAM Combining Visual Odometry and Extended Information Filter. Sensors [Internet]. 2015;15:18742–66. Available from: www.mdpi.com/journal/sensors
34. Ila V, Porta JM, Andrade-Cetto J. Information-based compact pose SLAM. IEEE Trans Robot. 2010;26(1):78–93.
35. Del Moral P. Nonlinear filtering: Interacting particle resolution. Comptes Rendus l'Académie des Sci - Ser I - Math. 1996;2(4):555–80.
36. Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE. 1993;140(2): 107–13.
37. Liu JS, Rong C. Sequential Monte Carlo methods for dynamic systems. J Am Stat Assoc. 1998;93(443):1032–1044.
38. Øivind Skare EB and LH. Improved Sampling-Importance Resampling and Reduced Bias Importance Sampling. Scand J Stat. 2003;30(4):719-737.
39. Bruno MGS. Regularized Particle Filters. Seq Monte Carlo Methods Nonlinear Discret Filtering Synth Lect Signal Process Springer. 2013.
40. Blackwell D. Conditional Expectation and Unbiased Sequential Estimation. Ann Math Stat. 1947;18(1):105–10.
41. Doucet A, Murphy K, Berkeley UC. Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. 1999.
42. Murphy K SR. Rao-Blackwellised Particle Filtring for Dynamic Bayesian Networks. Springer New York. 2001;43(2):499–515.
43. Montemerlo M, Thrun S, Koller D, Wegbreit B. FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem. Eighteenth Natl Conf Artif Intell Menlo Park. 2002;593–598.
44. Montemerlo M, Thrun S, Siciliano B. FastSLAM:A Scalable Method for the Simultaneous Localization and Mapping Problem in Robotics. Springer. 2007;27.
45. Michael M, Thrun S, Koller D, Wegbreit B. FastSLAM 2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges. IJCAI'03 Proc 18th Int Jt Conf Artif Intell. 2003;1151–6.
46. Kim C, Sakthivel R, Chung WK. Unscented FastSLAM : A Robust Algorithm for the Simultaneous Localization and Mapping Problem. 2008.
47. Eliazar A, Parr R. DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks. IJCAI Int Jt Conf Artif Intell. 2003;1135–42.
48. Eliazar AI, Parr R. DP-SLAM 2.0. Dep Comput Sci Duke Univ North Carolina 27708.
49. Zikos N, Petridis V. 6-DoF Low Dimensionality SLAM (L-SLAM). J Intell Robot Syst. 2015;79:55–72.
50. Nie F, Zhang W, Yao Z, Shi Y, Li F, Huang Q. LCPF: A Particle Filter Lidar SLAM System with Loop Detection and Correction. IEEE Access. 2020;8:20401–12.
51. Hua J, Cheng M. Improved UFastSLAM algorithm based on particle filter. IEEE 9th Jt Int Inf Technol Artif Intell Conf. 2020;(2693–2865):1050–5.
52. Lin M, Member S, Canjun Yang, Li D. An Improved Transformed Unscented FastSLAM with Genetic Resampling. IEEE Trans Ind Electron. 2019;66(5):3583–94.
53. Tang M, Chen Z, Yin F. An Improved Adaptive Unscented FastSLAM with Genetic Resampling. Int J Control Autom Syst. 2021;19(4):1677–90.
54. Lu F, Milios E. Globally Consistent Range Scan Alignment for Environment Mapping. Auton Robots. 1997;4(4):333–49.

55. Thrun S. The GraphSLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures. Int J Rob Res. 1998;25: 403–29.
56. Grisetti G, Stachniss C, Grzonka S, Burgard W. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. Robot Sci Syst. 2008;3:65–72.
57. Frese U. Treemap: An O(log n) algorithm for indoor simultaneous localization and mapping. Auton Robots. 2006;103–22.
58. Grisetti G, Kümmerle R, Stachniss C, Frese U, Hertzberg C. Hierarchical optimization on manifolds for online 2D and 3D mapping. Proc - IEEE Int Conf Robot Autom. 2010;273–8.
59. Kaess M, Johannsson H, Roberts R, Ila V, Leonard JJ, Dellaert F. ISAM2: Incremental smoothing and mapping using the Bayes tree. Int J Rob Res. 2012;31(2):216–35.
60. Rainer K, Grisetti G, Hauke S, Kurt. K, Abstract—Many WB. g2o: A General Framework for Graph Optimization Rainer. IEEE Int Conf Robot Autom Shanghai Int Conf Cent. 2011;3607–13.
61. Dellaert F. Factor Graphs and GTSAM. A hands-on Introd Tech Rep (Georgia Tech, Atlanta 2012) [Internet]:1–27. Available from: http://tinyurl.com/gtsam.
62. Agarwal P, Tipaldi GD, Spinello L, Stachniss C, Burgard W. Robust map optimization using dynamic covariance scaling. Proc - IEEE Int Conf Robot Autom. 2013.
63. Strasdat H, Davison AJ, Montiel JMM, Konolige K. Double window optimisation for constant time visual SLAM. Int Conf Comput Vis. 2011.
64. M. Ruhnke R. Kümmerle G, Grisetti WB. Highly accurate 3D surface models by sparse surface adjustment. IEEE Int Conf Robot Autom. 2012;(10.1109/ICRA.2012.6225077).
65. Stachniss C, Leonard JJ, Thrun S. Simultaneous Localization and Mapping. In: Multimedia Contents 1153 springer Handbook Robotics Part E/46. 2016;1153–75.
66. Zhao L, Huang S, Dissanayake G. Linear SLAM: Linearising the SLAM problems using submap joining. Automatica. 2018;1–22.
67. Holder M, Hellwig S, Winner H. Real-time pose graph SLAM based on radar. IEEE Intell Veh Symp. 2019.
68. Youyang F, Qing W, Gaochao Y. Incremental 3-D pose graph optimization for SLAM algorithm without marginalization. Int J Adv Robot Syst. 2020;1–14.
69. Fan T, Wang H, Rubenstein M, Murphey T. Cpl-slam: Efficient and certifiably correct planar graph-based slam using the complex number representation. IEEE Trans Robot. 2020;36(6):1719–37.
70. Sun Z, Wu B, Xu CZ, Sarma SE, Yang J, Kong H. Frontier Detection and Reachability Analysis for Efficient 2D Graph-SLAM Based Active Exploration. IEEE/RSJ Int Conf Intell Robot Syst. 2020;2051–8.
71. Pierzchała M, Giguère P, Astrup R. Mapping forests using an unmanned ground vehicle with 3D LiDAR and graph-SLAM. Comput Electron Agric. 2018;145:217–25.
72. Press W, Keukolsky S WV and BF. Levenberg Marquardt Method. Numer Recipes C Art Sci Comput. 1992;542–54.
73. Shum HY, Ke Q and ZZ. Efficient Bundle Adjustment with Virtual Key Frames: A Hierarchical Approach to Multi-frame Structure from Motion. IEEE Comput Soc Conf Comput Vis Pattern Recognition. 1999.
74. Hartley R, Zisserman A. Multiple View Geometry in Computer Vision. Cambridge Univ Press. 2000;18.
75. Melbouci K, Collette SN, Gay-Bellile V, Ait-Aider O, Carrier M, Dhome M. Bundle adjustment revisited for SLAM with RGBD sensors. Proc 14th IAPR Int Conf Mach Vis Appl MVA. 2015;166–9.
76. Frost D, Prisacariu V, Murray D. Recovering Stable Scale in Monocular SLAM Using Object-Supplemented Bundle Adjustment. IEEE Trans Robot. 2018;34(3):1–11.
77. Schops T, Sattler T, Pollefeys M. Bad slam: Bundle adjusted direct RGB-D slam. IEEE/CVF Conf Comput Vis Pattern Recognit. 2019;134–44.
78. Zhao Y, Smith JS, Vela PA. Good Graph to Optimize: Cost-Effective, Budget-Aware Bundle Adjustment in Visual SLAM. Comput Vis Pattern Recognit [Internet]. 2020;1–20.
Available from: http://arxiv.org/abs/2008.10123
79. Wang K, Ma S, Ren F, Lu J. SBAS: Salient Bundle Adjustment for Visual SLAM. J LATEX Cl FILES.arxiv201211863v1[csRO]. 2015;14(8):1–11.
80. Campos C, Elvira R, Rodriguez JJG, Montiel JMM, Tardos JD. ORB-SLAM3: An Accurate Open-Source Library for Visual. Visual-Inertial and Multimap SLAM. IEEE Trans Robot. 2021;37(6): 1874–90.
81. Gonzalez M, Marchand E, Kacete A, Royan J. S3LAM: Structured Scene SLAM. Robotics [Internet]. 2022.
Available from: http://arxiv.org/abs/2109.07339
82. Tanaka T, Sasagawa Y, Okatani T. Learning to Bundle-adjust: A Graph Network Approach to Faster Optimization of Bundle Adjustment for Vehicular SLAM. Proc IEEE Int Conf Comput Vis. 2021;6230–9.
83. Rosten E, Drummond T. Machine Learning for High-Speed Corner Detection. Leonardis A, Bischof H, Pinz A Comput Vis – ECCV 2006ECCV 2006 Lect Notes Comput Sci Springer. Berlin. Heidelb. 2006;3951:430–43.
84. Bay H, Ess A, Tuytelaars T, Gool L Van. Speeded-Up Robust Features (SURF). Comput Vis Image Underst. 2008;110(3): 346–59.
85. Calonder M, Lepetit V, Strecha C, Fua P. BRIEF: Binary robust independent elementary features. ECCV 2010 Lect Notes Comput Sci Springer. Berlin. Heidelberg. 2010;6314:778–92.
86. E. Rublee, V. Rabaud KK and GB. ORB: an efficient alternative to SIFT or SURF. Int Conf Comput Vision. Barcelona. Spain. 2011;2564–71.
87. Harris C, Stephens M. A Combined Corner and Edge Detector. Proc 4th Alvey Vis Conf. 1988;147--151.
88. Civera J, Lee SH. RGB-D Odometry and SLAM. Rosin, P, Lai, YK, Shao, L, Liu, Y RGB-D Image Anal Process Adv Comput Vis Pattern Recognition Springer. Cham. 2019;117–144.
89. Davison AJ, Reid ID NDM, Stasse O. Monoslam: real-time single camera SLAM. Pattern Anal Mach Intell IEEE. 2007;29(6): 1052–67.
90. Davison AJ. Real-time simultaneous localisation and mapping with a single camera. Proc Ninth IEEE Int Conf Comput Vision. Nice. Fr. 2003;2:1403–10.
91. Klein G, Murray D. Parallel tracking and mapping for small AR workspaces. 2007 6th IEEE ACM Int Symp Mix Augment Reality. ISMAR. 2007;225–34.
92. Klein G, Murray D. Parallel tracking and mapping on a camera phone. th IEEE Int Symp Mix Augment Reality. Orlando FL. USA. 2009. 2009;83–6.
93. Endres F, Hess J, Engelhard N, Sturm J DC and WB. An evaluation of the RGB-D SLAM system. IEEE Int Conf Robot Autom Saint Paul. MN. USA. 2012;3(c):1691–6.
94. Mur-Artal R, Montiel JMM, Tardos JD. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. IEEE Trans Robot. 2015;31(5):1147–63.
95. Tardos DG-L and JD. Bags of Binary Words for Fast Place Recognition in Image Sequences. IEEE Trans Robot. 28(5):1188–97.
96. Strasdat H, Davison AJ, Montiel. JMM. Scale Drift-Aware Large Scale Monocular SLAM. Robot Sci Syst. 2010.
97. Mei C, Sibley G, Newman P. Closing loops without places. IEEE/RSJ 2010 Int Conf Intell Robot Syst IROS 2010 - Conf Proc. 2010;3738–44.
98. Mur-Artal R, Tardós JD. ORB-SLAM: Tracking and Mapping Recognizable Features. Conf Work Multi VIew Geom Robot - RSS 2014 [Internet]. 2014. Available from:
http://vindelman.technion.ac.il/events/mvigro/MurArtal14rss_ws.pdf
99. Mur-Artal R, Tardos JD. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. IEEE Trans Robot. 2017;33(5):1255–62.

100. Sumikura S, ShibuyaM KS. OpenVSLAM: A versatile visual SLAM framework. MM ’19 Proc 27th ACM Int Conf Multimedia. 2019;2292–5.
101. Muñoz-Salinas R, Medina-Carnicer R. UcoSLAM: Simultaneous localization and mapping by fusion of keypoints and squared planar markers. Comput Vis Pattern Recognit. 2019;
102. Sun Q, Yuan J, Zhang X, Duan F. Plane-Edge-SLAM: Seamless Fusion of Planes and Edges for SLAM in Indoor Environments. IEEE Trans Autom Sci Eng. 2021;18(4):2061–75.
103. Newcombe RA, Lovegrove SJ, Davison AJ. DTAM: Dense Tracking and Mapping in Real-Time. Int Conf Comput Vision, Barcelona, Spain. 2011;2320–7.
104. J Engel JS and DC. Semi-Dense Visual Odometry for a Monocular Camera. IEEE Int Conf Comput Vision. Sydney. NSW. Aust. 2013;1449–56.
105. Engel J, Sturm J, Cremers D. LSD-SLAM: Large-Scale Direct Monocular SLAM. Proc IEEE Int Conf Comput Vis. 2013;1449–56.
106. Engel J, Stuckler J DC. Large-scale direct SLAM with Stereo Cameras. IEEE/RSJ Int Conf Intell Robot Syst (IROS). Hamburg Ger. 2015;1935–42.
107. Engel J, Cremers, Daniel, Caruso D. Large-scale direct SLAM for omnidirectional cameras. IEEE/RSJ Int Conf Intell Robot Syst (IROS). Hamburg Ger. 2015;141–8.
108. Forster C, Pizzoli M, Scaramuzza D. SVO : Fast Semi-Direct Monocular Visual Odometry. IEEE Int Conf Robot Autom (ICRA)Hong Kong. China. 2014;15–22.
109. Engel J, Koltun V, Cremers D. Direct Sparse Odometry. IEEE Trans Pattern Anal Mach Intell. 2018;40(3):611–25.
110. Gao X, Wang R, Demmel N, Cremers D. LDSO: Direct Sparse Odometry with Loop Closure. IEEE Int Conf Intell Robot Syst Spain. 2018;2198–204.
111. Sheng C, Pan S, Gao W, Tan Y, Zhao T. Dynamic-DSO: Direct sparse odometry using objects semantic information for dynamic environments. Appl Sci. 2020;10(4):1–20.
112. Newcombe RA, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison AJ et al. KinectFusion: Real-time dense surface mapping and tracking. 210th IEEE Int Symp Mix Augment Reality. Basel. Switzerland. 2011;127–36.
113. Concha A, Civera J. RGBDTAM: A cost-effective and accurate RGB-D tracking and mapping system. IEEE Int Conf Intell Robot Syst Concha J Civera. RGBDTAM A cost-effective accurate RGB-D Track Mapp Syst 2017 IEEE/RSJ Int Conf Intell Robot Syst (IROS). Vancouver. 2017;6756–63.
114. Fontán A JC and RT. Information-Driven Direct RGB-D Odometry. IEEE/CVF Conf Comput Vis Pattern Recognit (CVPR). Seattle. WA. USA. 2020;4928–36.
115. Ma L, Kerl C, Stückler J, Cremers D. CPA-SLAM: Consistent Plane-Model Alignment for Direct RGB-D SLAM. IEEE Int Conf Robot Autom (ICRA). Stock Sweden [Internet]. 2016;1:1285–91. Available from: https://pdfs.semanticscholar.org/d41a/4ab403d6c7611047f83f575cf4c16bfd5282.pdf
116. Dai A, Nießner M, Zolloer M, Izadi S and C. BundleFusion: Real-time Globally Consistent 3D Reconstruction using On-the-fly Surface Re-integration. IEEE Int Conf Progr Compr. 2022;1(1):19.
117. Hsiao M, Westman E, Zhang G, Kaess M. Keyframe-based dense planar SLAM. IEEE Int Conf Robot Autom (ICRA). Singapore. 2017;5110–7.
118. Dong X, Cheng L, Peng H, Li T. FSD-SLAM: a fast semi-direct SLAM algorithm. Complex Intell Syst [Internet]. 2022;8:1823–34. Available from: https://doi.org/10.1007/s40747-021-00323-y
119. Bloesch M, Omari S, Hutter M, Siegwart R. Robust visual inertial odometry using a direct EKF-based approach. IEEE/RSJ Int Conf Intell Robot Syst (IROS). Hamburg Ger. 2015;298–304.
120. Sun K, Mohta K, Pfrommer B, Watterson M, Liu S, Mulgaonkar Y, et al. Robust Stereo Visual Inertial Odometry for Fast Autonomous Flight. IEEE Robot Autom Lett. 2018;3(2):965–72.
121. Mourikis AI, Roumeliotis SI. A multi-state constraint Kalman filter for vision-aided inertial navigation. Proc 2007 IEEE Int Conf Robot Autom Rome. Italy. 2007;3565–72.
122. Leutenegger S, Lynen S, Bosse M, Siegwart R, Furgale P. Keyframe-Based Visual-Inertial Odometry Using Nonlinear Optimization. Int J Rob Res. 2014;34(3):1–26.
123. Schneider T, Dymczyk M, Fehr M, Egger K, Lynen S, Gilitschenski I et al. Maplab: An Open Framework for Research in Visual-Inertial Mapping and Localization. IEEE Robot Autom Lett. 2018;3(3):1418–25.
124. Liu H, Chen M, Zhang G, Bao H, Bao Y. ICE-BA: Incremental, Consistent and Efficient Bundle Adjustment for Visual-Inertial SLAM. IEEE/CVF Conf Comput Vis Pattern Recognition. Salt Lake City. UT USA. 2018;1974–82.
125. Forster C, Carlone L, Dellaert F, Scaramuzza D. On-Manifold Preintegration for Real-Time Visual-Inertial Odometry. Iin IEEE Trans Robot. 2017;33(1):1–20.
126. Von Stumberg L, Usenko V, Cremers D. Direct Sparse Visual-Inertial Odometry Using Dynamic Marginalization. IEEE Int Conf Robot Autom (ICRA). Brisbane QLD. Aust. 2018;2510–7.
127. Qin T, Li P, Shen S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. IEEE Trans Robot. 2018;34(4):1004–20.
128. Yang Z, Shen S. Monocular visual-inertial state estimation with online initialization and camera-IMU extrinsic calibration. IEEE Trans Autom Sci Eng. 2017;14(1):39–51.
129. Mur-Artal R, Tardos JD. Visual-Inertial Monocular SLAM with Map Reuse. IEEE Robot Autom Lett. 2017;2(2):796–803.
130. He Y, Zhao J, Guo Y, He W, Yuan K. PL-VIO: Tightly-coupled monocular visual–inertial odometry using point and line features. Sensors (Switzerland). 2018;18(4):1–25.
131. Zheng F, Tsai G, Zhang Z, Liu S, Chu CC, Hu H. Trifo-VIO: Robust and Efficient Stereo Visual Inertial Odometry Using Points and Lines. IEEE/RSJ Int Conf Intell Robot Syst (IROS). Madrid Spain. 2018;3686–93.
132. Li X, Li Y, Ornek EP, Lin J, Tombari F. Co-Planar Parametrization for Stereo-SLAM and Visual-Inertial Odometry. IEEE Robot Autom Lett. 2020;5(4):6972–9.
133. Rosinol A, Sattler T, Pollefeys M, Carlone L. Incremental visual-inertial 3d mesh generation with structural regularities. Int Conf Robot Autom (ICRA). Montr QC. Canada. 2019;8220–6.
134. Seiskari O, Rantalankila P, Kannala J, Ylilammi J, Rahtu E, Solin A. HybVIO: Pushing the Limits of Real-time Visual-inertial Odometry. IEEE/CVF Winter Conf Appl Comput Vis (WACV). Waikoloa HI, USA. 2022;287-296.
135. Kaushik V, Jindgar K, Lall B. ADAADepth: Adapting data augmentation and attention for self-supervised monocular depth estimation. IEEE Robot Autom Lett. 2021;6(4):7791–8.
136. Tateno K, Tombari F, Laina I NN. CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction. Comput Vis Pattern Recognit. 2017;6243–52.
137. Bloesch M, Czarnowski J, Clark R, Leutenegger S AJD. CodeSLAM-Learning a Compact. Optimisable Representation for Dense Visual SLAM. 2018;2560–8. Available from: http://openaccess.thecvf.com/content_cvpr_2018/papers/Bloesch_CodeSLAM_--_Learning_CVPR_2018_paper.pdf
138. Mohanty V, Agrawal S, Datta S, Ghosh A, Vishnu Dutt Sharma DC. DeepVO: A Deep Learning approach for Monocular Visual Odometry. 2016. Available from: http://arxiv.org/abs/1611.06069
139. Li R, Wang S, Long Z, Gu D. UnDeepVO: Monocular Visual Odometry Through Unsupervised Deep Learning. IEEE Int Conf Robot Autom (ICRA). Brisbane QLD. Aust. 2018;7286–91.
140. Yang N, Von Stumberg L, Wang R, Cremers D. D3VO: Deep Depth, Deep Pose and Deep Uncertainty for Monocular Visual Odometry. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit. 2020;1278–89.

141. Yin Z, Shi J. GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose http: 2018;1983–92. Available from: geonet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose http://arxiv.org/abs/1803.02276v2
142. Zhao C, Sun L, Purkait P, Duckett T, Stolkin R. Learning Monocular Visual Odometry with Dense 3D Mapping from Dense 3D Flow. IEEE/RSJ Int Conf Intell Robot Syst (IROS). Madrid Spain. 2018;6864–71.
143. Zhou T, Brown M, Snavely N DGL. Unsupervised Learning of Depth and Ego-Motion from Video. CEEE Conf Comput Vis Pattern Recognit (CVPR), Honolulu, HI, USA [Internet]. 2017;6612–9. Available from: https: //github.com/tinghuiz/SfMLearner.%0A2
144. Zagoruyko S, Komodakis N. Learning to Compare Image Patches via Convolutional Neural Networks Sergey. IEEE Conf Comput Vis Pattern Recognit (CVPR). Boston MA. USA. 2015;4353–61.
145. G VKB, Carneiro G, Reid I. Learning Local Image Descriptors with Deep Siamese and Triplet Convolutional Networks by Minimizing Global Loss Functions. IEEE Conf Comput Vis Pattern Recognit (CVPR)IEEE Conf Comput Vis Pattern Recognit (CVPR). Las Vegas NV. USA [Internet]. 2016;5385–94. Available from: http://openaccess.thecvf.com/content_cvpr_2016/supplemental/G_Learning_Local_Image_2016_CVPR_supplemental.pdf
146. Mayer N, Ilg E, Hausser P, Fischer P. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. IEEE Conf Comput Vis Pattern Recognit (CVPR). Las Vegas NV. USA. 2016;4040–8.
147. Tankovich V, Häne C, Zhang Y, Kowdle A, Fanello S, Bouaziz S. HitNet: Hierarchical Iterative Tile Refinement Network for Real-time Stereo Matching. IEEE/CVF Conf Comput Vis Pattern Recognit (CVPR). Nashville TN. USA. 2021;14357–67.
148. Huang PH, Matzen K, Kopf J, Ahuja N, Huang J Bin. DeepMVS: Learning Multi-view Stereopsis. IEEE/CVF Conf Comput Vis Pattern Recognition. Salt Lake City UT. USA. 2018;2821–30.
149. Song X, Zhao X, Hu H, Fang L. EdgeStereo: A Context Integrated Residual Pyramid Network for Stereo Matching. Comput Vis – ACCV 2018 ACCV 2018 Lect Notes Comput Sci. 2018;arXiv:1803.05196.
150. 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. 2020;8:3212–23.
151. Zhang W, Liu G, Tian G. A Coarse to Fine Indoor Visual Localization Method Using Environmental Semantic Information. IEEE Access. 2019;7:21963–70.
152. Lin YF, Yang LJ, Yu CY, Peng CC, Huang DC. Object recognition and classification of 2D-SLAM using machine learning and deep learning techniques. Int Symp Comput Consum Control (IS3C). Taichung City. Taiwan. 2020;473–6.
153. Wang S, Clark R, Wen H, Trigoni N. End-to-End, Sequence-to-Sequence Probabilistic Visual Odometry through Deep Neural Networks. Int J Robot Res 37 [Internet]. 2018;37:513–42. Available from: doi.org/10.1177/0278364917734298
154. Li J, Li Z, Feng Y, Liu Y, Shi G. Development of a Human-Robot Hybrid Intelligent System Based on Brain Teleoperation and Deep Learning SLAM. IEEE Trans Autom Sci Eng. 2019;16(4):1664–74.
155. Lan E. A Novel Deep Learning Architecture By Integrating Visual Simultaneous Localization And Mapping (Vslam) Into Cnn For Real-Time Surgical Video Analysis. 19th Int Symp Biomed Imaging (ISBI). Kolkata. India. 2022;1–5.
156. Hu S, Li D, Tang G, Xu X. A 3D semantic visual SLAM in dynamic scenes. 6th IEEE Int Conf Adv Robot Mechatronics (ICARM). Chongqing. China. 2021;522–8.
157. Almalioglu Y, Saputra MRU, De Gusmao PPB, Markham A, Trigoni N. GANVO: Unsupervised deep monocular visual odometry and depth estimation with generative adversarial networks. Proc - IEEE Int Conf Robot Autom Conf Robot Autom (ICRA), Montr QC. Canada. 2019;5474–80.
158. Ban X, Wang H, Chen T, Wang Y, Xiao Y. Monocular Visual Odometry based on depth and optical flow Using deep learning. IEEE Trans Instrum Meas. 2021;70:1–19.
159. Liang HJ, Sanket NJ, Fermuller C, Aloimonos Y. SalientDSO: Bringing Attention to Direct Sparse Odometry. IEEE Trans Autom Sci Eng. 2019;16(4):1619–26.
160. Tang J, Ericson L, Folkesson J, Jensfelt P. GCNv2: Efficient Correspondence Prediction for Real-Time SLAM. IEEE Robot Autom Lett. 2019;4(4):3505–10.
161. Detone D, Malisiewicz T, Rabinovich A. SuperPoint: Self-supervised interest point detection and description. EEE/CVF Conf Comput Vis Pattern Recognit Work (CVPRW). Salt Lake City UT. USA. 2018;337–49.
162. Kwang Moo Yi, Eduard Trulls, Vincent Lepetit PF. LIFT: Learned Invariant Feature Transform Kwang. Springer Int Publ AG 2016.
163. Ganti P, Waslander S. Network uncertainty informed semantic feature selection for visual SLAM. 16th Conf Comput Robot Vis (CRV) Kingston QC. Canada. 2019;121–8.
164. Gu X, Wang Y, Ma T. DBLD-SLAM: A Deep-Learning Visual SLAM System Based on Deep Binary Local Descriptor. Int Conf Control Autom Inf Sci (ICCAIS). Xi'an China. 2021;325–30.
165. Krishnan KS, Sahin F. ORBDeepOdometry - A feature-based deep learning approach to monocular visual odometry. 14th Annu Conf Syst Syst Eng (SoSE). Anchorage AK. USA. 2019;296–301.
166. Huang Z, Wang X, Huang L, Huang C, Wei Y, Liu W. CCNet: Criss-cross attention for semantic segmentation. IEEE Trans Pattern Anal Mach Intell. 2019;603–12.
167. Qin Z, Wang J, Lu Y. MonoGRNet: A General Framework for Monocular 3D Object Detection. IEEE Trans Pattern Anal Mach Intell. 2021;44(9):5170–84.
168. Ronald Clark, Sen Wang, Hongkai Wen, Andrew Markham NT. VINet: Visual-Inertial Odometry as a Sequence-to-Sequence Learning Problem. Proc Thirty-First AAAI Conf Artif Intell. 2017;31(1):3995–4001.
169. G. Costante, M. Mancini PV and TAC. "Exploring Representation Learning With CNNs for Frame-to-Frame Ego-Motion Estimation,. IEEE Robot Autom Lett. 2016;1:18-25.
170. Gu X. DBLD-SLAM : A Deep-Learning Visual SLAM System Based on Deep Binary Local Descriptor. 2021;325–30.
171. Vijayanarasimhan S, Ricco S, Schmid C. SfM-Net: Learning of Structure and Motion from Video. 2017. arXiv preprint arXiv:1704.07804.
172. Konda K, Memisevic R. Learning visual odometry with a convolutional network. Proc ofthe 10th Int Conf Comput Vis Theory Appl. 2015;1:486–90.
173. Wang S, Clark R, Wen H, Trigoni N. DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks. IEEE Int Conf Robot Autom (ICRA). Singapore. 2017; 2043–50.
174. Clark R, Wang S, Markham A, Trigoni N, Wen H. VidLoc : A Deep Spatio-Temporal Model for 6-DoF Video-Clip Relocalization. IEEE Conf Comput Vis Pattern Recognit (CVPR). Honolulu HI. USA. 2017;2652–60.
175. Mahattansin N, Sukvichai K PB and TI. Improving Relocalization in Visual SLAM by using Object Detection. 9th Int Conf Electr Eng Comput Telecommun Inf Technol (ECTI-CON). Pr Khiri Khan. Thail. 2022;1–4.
176. Li R, Liu Q, Gui J DG and HH. Indoor Relocalization in Challenging Environments With Dual-Stream Convolutional Neural Networks. IEEE Trans Autom Sci Eng. 2018;15(2):651–62.
177. Dong S, Fan Q, Wang H, Shi J, Yi L, Funkhouser T, et al. Robust Neural Routing Through Space Partitions for Camera Relocalization in Dynamic Indoor Environments. IEEE/CVF Conf Comput Vis Pattern Recognit (CVPR), Nashville, TN, USA. 2021;8540–50.
178. Nakashima R, Seki A. SIR-Net : Scene-Independent End-to-End Trainable Visual Relocalizer Ryo Nakashima. Int Conf 3D Vis (3DV). Quebec City QC. Canada. 2019;472–81.

179. Zhou L. Visual Relocalization using Long-Short Term Memory Fully Convolutional Network. IEEE Int Symp Mix Augment Real Adjun (ISMAR-Adjunct), Munich, Ger. 2018;258–63.
180. Duong ND, Kacete A, Sodalie C, Oierre-Yves R JR. xyzNet: towards Machine learning camera relocalization by using a scene coordinate prediction network. IEEE Int Symp Mix Augment Real Adjun. 2018;2–7.
181. Wu X, Tian X, Zhou J, Xu P, Chen J. Loop Closure Detection for Visual SLAM Based on SuperPoint Network. 2019 Chinese Autom Congr (CAC). Hangzhou. China. 2019;3789–93.
182. Merrill N, Huang G. Lightweight Unsupervised Deep Loop Closure. Conf Robot Sci Syst . 2018;1–10.
183. Xia Y, Li J, Qi L, Fan H. Loop Closure Detection for Visual SLAM Using PCANet Features. Int Jt Conf Neural Networks (IJCNN), Vancouver BC. Canada,. 2016;2274–81.
184. Dai K, Cheng L, Yang R, Yan G. Loop Closure Detection Using KPCA and CNN for Visual SLAM. 40th Chinese Control Conf (CCC). Shanghai. China. 2021;8088–93.
185. Xiong F, Ding Y, Yu M, Zhao W NZ and PR. A Lightweight sequence-based Unsupervised Loop Closure Detection. Int Jt Conf Neural Networks (IJCNN). Shenzhen. China. 2021;1–8.
186. Huang L, Zhu M, Zhang M. Visual Loop Closure Detection Based on Lightweight Convolutional Neural Network and Product Quantization. IEEE 12th Int Conf Softw Eng Serv Sci (ICSESS). Beijing. China. 2021;122–6.
187. Zhu M, Huang L. Fast and Robust Visual Loop Closure Detection with Convolutional Neural Network. IEEE 3rd Int Conf Front Technol Inf Comput (ICFTIC). Greenville SC. USA. 2021;3681–91.
188. Ma J, Wang S, Zhang K, He Z, Huang J XM. Fast and Robust Loop-Closure Detection via Convolutional Auto-Encoder and Motion Consensus. IIEEE Trans Ind Informatics. 2022;18(6):3681–91.
189. Cai S, Zhou D, Guo R, Zhou H, Peng K. Implementation of Hybrid Deep Learning Architecture on Loop-Closure Detection. 2018; 521–6.
190. Liu Y, Xiang R, Zhang Q, Ren Z, Cheng J. Loop Closure Detection based on Improved Hybrid Deep Learning Architecture. IEEE Int Conf Ubiquitous Comput Commun Data Sci Comput Intell Smart Comput Netw Serv (SmartCNS). Shenyang. China. 2019;312–7.
191. Shi X, Li L. Loop Closure Detection for Visual SLAM Systems Based on Convolutional Netural Network. IEEE 24th Int Conf Comput Sci Eng (CSE). Shenyang. China. 2021;123–9.
192. Zhou Y, Wang Y, Poiesi F, Qin Q, Wan Y. Loop Closure Detection Using Local 3D Deep Descriptors. IEEE Robot Autom Lett. 2022;7(3):6335–42.
193. Osman H, Darwish N, Member S, Bayoumi A. LoopNet: Where to Focus? Detecting Loop Closures in Dynamic Scenes. IEEE Robot Autom Lett. 2022;7(2):2031–8.
194. Bhutta MUM, Sun Y, Lau D, Liu M, Member S. Why-So-Deep : Towards Boosting Previously Trained Models for Visual Place Recognition. 1824 IEEE Robot Autom Lett. 2022;7(2):1824–31.
195. Gauglitz S, Sweeney C, Ventura J MT and TH. Live Tracking and Mapping from Both General and Rotation-Only Camera Motion. IEEE Int Symp Mix Augment Real (ISMAR). Atlanta GA. USA. 2012;13–22.
196. Daniel HC, Kim K, Kannala J, Pulli K, Heikkilä J. DT-SLAM: Deferred triangulation for robust SLAM. 2nd Int Conf 3D Vision. Tokyo. Japan. 2014;609–16.

Zoulikha Bouhamatou: https://orcid.org/0000-0002-3985-0147

Foudil Abdessemed: https://orcid.org/0000-0003-0935-3147