

Camera-based simultaneous localization and mapping: methods, camera types, and deep learning trends

Anak Agung Ngurah Bagus Dwimantara, Oskar Natan, Novelio Putra Indarto, Andi Dharmawan

Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia

Article Info

Article history:

Received Dec 5, 2024

Revised Feb 20, 2025

Accepted Mar 5, 2025

Keywords:

Camera

Deep learning

Map reconstruction

Simultaneous localization and mapping

Visual Odometry

ABSTRACT

The development of simultaneous localization and mapping (SLAM) technology is crucial for advancing autonomous systems in robotics and navigation. However, camera-based SLAM systems face significant challenges in accuracy, robustness, and computational efficiency, particularly under conditions of environmental variability, dynamic scenes, and hardware limitations. This paper provides a comprehensive review of camera-based SLAM methodologies, focusing on their different approaches for pose estimation, map reconstruction, and camera type. The application of deep learning also will be discussed on how it is expected to improve performance. The objective of this paper is to advance the understanding of camera-based SLAM systems and to provide a foundation for future innovations in robust, efficient, and adaptable SLAM solutions. Additionally, it offers pertinent references and insights for the design and implementation of next-generation SLAM systems across various applications.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Oskar Natan

Department of Computer Science and Electronics, Universitas Gadjah Mada

Sekip Utara Bulaksumur, Yogyakarta 55281, Indonesia

Email: oskarnatan@ugm.ac.id

1. INTRODUCTION

Simultaneous localization and mapping (SLAM) is a fundamental technology widely used in robotics, autonomous vehicles, and other applications where machines must interpret and navigate their surroundings. The SLAM process involves simultaneously generating a map of an unfamiliar environment and determining the device's position within it. Among various sensor options for SLAM, cameras are particularly notable due to their low cost, compact design, and ability to capture detailed visual data [1]. Camera-based SLAM has attracted substantial interest because it utilizes visual information to estimate motion and construct maps, offering a cost-effective alternative to sensors like LiDAR and excelling in tasks that demand high spatial resolution [2].

Despite its promise, visual SLAM faces several challenges, particularly in real-world applications. Dynamic environments, where objects are constantly moving, can disrupt feature tracking and reduce mapping accuracy [3]. Textureless surfaces, such as plain walls or floors, lack distinguishable features, making it difficult to extract and match key points [4]. Poor lighting conditions, such as dim environments or overexposed scenes, can degrade image quality and hinder the system's ability to detect and track features reliably. To overcome these obstacles, feature-based (indirect) methods [5] and direct methods [6] have provided a solid foundation for building accurate and efficient maps. Indirect methods follow a two-step process.

Camera-based SLAM systems can be classified into three main categories based on the type of camera used: monocular, stereo, and RGB-D systems [7]–[9]. The pipeline of camera-based SLAM generally

consists of three core stages: pose estimation, loop closure detection, and mapping. Pose estimation involves determining the camera's position and orientation within the environment. Loop closure detection identifies instances where the camera revisits previously explored areas, enabling the system to correct accumulated errors and enhance the global consistency of the estimated trajectory. Mapping, the final stage, focuses on creating a structured representation of the environment, such as a 3D map or other spatial models. The performance of SLAM systems is typically evaluated using widely recognized public datasets such as KITTI [10], New College [11], Technical University of Munich (TUM) [12], [13], EuRoc micro aerial vehicle (MAV) [14], which serve as benchmarks, offering real-world data collected from a variety of scenarios.

The rapid advancement of computer vision algorithms has significantly improved camera-based SLAM in recent years. This review aims to provide a comprehensive overview of camera-based SLAM, focusing on its key components, state-of-the-art techniques, and applications. We categorize and analyze existing methods, discuss their strengths and limitations, and highlight recent trends, including the incorporation of deep learning. Additionally, we address challenges and open problems in the field, emphasizing the importance of robust and scalable solutions for real-world applications.

2. DIFFERENT APPROACHES

Indirect methods and direct methods represent two primary approaches in camera-based SLAM, each with its own strengths and limitations. Indirect methods are particularly effective in environments rich in texture. Direct methods, in contrast, bypass the need for explicit feature extraction and instead operate on raw pixel intensities. There are also several methods for reconstructing maps, including sparse, semi-dense, and dense methods. While dense methods try to use and rebuild every pixel in the 2D picture domain, sparse methods simply use and reconstruct a chosen selection of independent points, usually corners. Direct and indirect are not interchangeable with the terms dense and sparse. All four pairings are actually feasible: Both direct and sparse, as well as direct and dense, indirect and dense, indirect and sparse [15].

2.1. Direct methods

The number of reconstructed points varies among the three types of direct methods in SLAM: dense, semi-dense, and sparse. These modifications strike a compromise between trade-offs between map detail, computational efficiency, and environmental robustness. Dense methods use all of the pixel intensity values in the image to reconstruct the surroundings and estimate camera motion. A notable example of this type is ElasticFusion by Whelan *et al.* [16], who used joint optimization, photometric pose estimation and geometric pose estimation. They utilize the randomized fern encoding [17] for appearance-based place recognition and five cost functions to optimize the deformation graph. The accuracy of the generated map is then maintained by optimizing this deformation graph, which is made up of a collection of nodes and edges dispersed throughout the model to be deformed.

Semi-dense methods do not rebuild the entire surface. One well-known example that shows semi-dense mapping capabilities in large-scale environments is large-scale direct SLAM (LSD-SLAM) [6]. The technique combines filtering-based estimate of semi-dense depth maps with direct image alignment. New camera images are continuously tracked by the tracking component. Filtering over several per-pixel, small-baseline stereo comparisons in conjunction with interleaved spatial regularization, as in [18], refines depth. They identify previously visited areas using fast appearance-based mapping (FABMAP) [19] and utilize pose graph optimization to minimize the error. Figure 1 shows the 3D map reconstruction of LSD-SLAM.

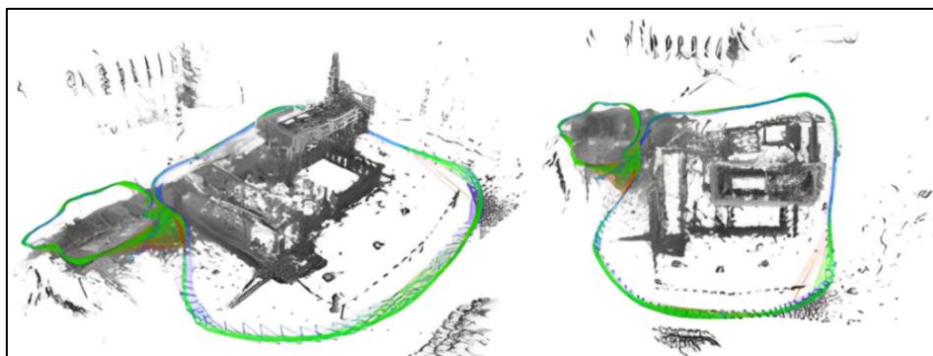


Figure 1. LSD-SLAM 3D reconstruction [6]

Dense (or semi-dense) methods, which usually favor smoothness, create a geometric prior by taking use of the connectedness of the employed picture region. However, in the sparse formulation, geometry parameters (key point positions) are conditionally independent given the camera poses and intrinsics, and the concept of neighborhood is absent [15]. These techniques are more reliant on the presence of observable key points in the surroundings and may result in less detailed maps. For the visual odometry (VO) application known as direct sparse odometry (DSO), Engel *et al.* [15] effectively integrated the advantages of direct methods with the adaptability of sparse approaches. They accomplish the photometric counterpart of windowed sparse bundle adjustment by jointly optimizing for all involved parameters (inverse depth values, camera intrinsics, and camera extrinsic). Additionally, they maintain the geometry representation used by previous direct techniques, which is representations of 3D points as inverse depth in a reference frame. The example result of the direct method is shown in Figure 2. Figure 2(a) shows the 3D map reconstruction of DSO.

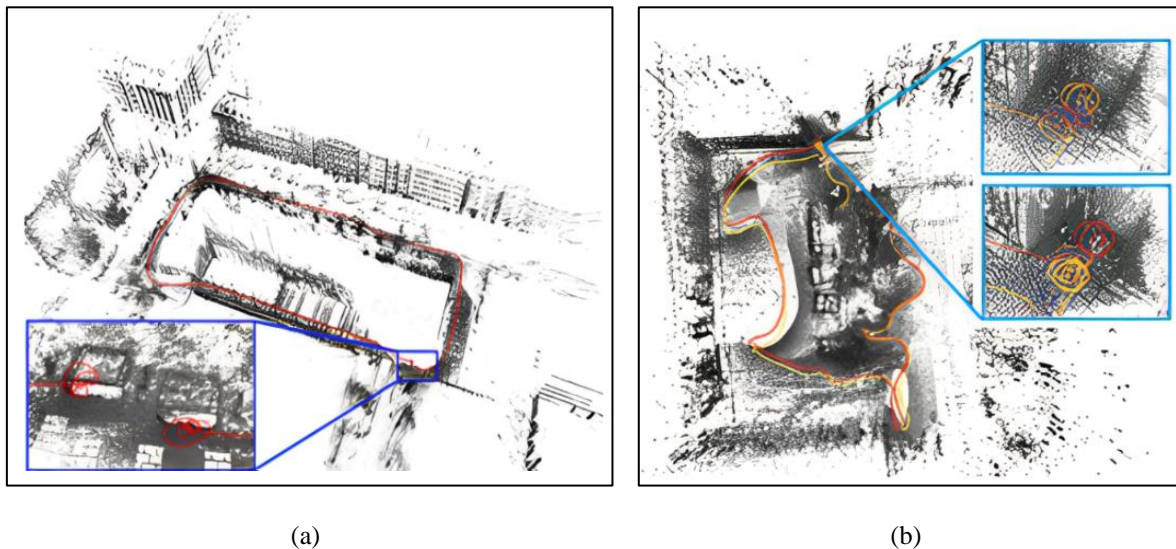


Figure 2. Example results of direct methods (a) DSO 3D reconstruction [15] and (b) the estimated trajectory using LDSO [20] (before (red) and after (yellow) loop closure)

However, VO suffers from the cumulative drift in unobservable degrees of freedom in the absence of a loop closing. This restricts the application to short-term motion estimation because it results in an erroneous long-term camera trajectory and map. A loop closures module was added to the DSO algorithm by Gao *et al.* [20]. While maintaining DSO's resilience in feature-poor contexts, they modify its point selection approach to prioritize recurring corner features. Then, using traditional BoW, the chosen corner characteristics are employed for loop closure detection [21]. The drift error is then decreased by using pose graph optimization. The effects of a loop closure module are depicted in Figure 2(b). Table 1 shows the comparison of the direct methods.

Table 1. The comparison of direct methods

SLAM algorithm	Map reconstruction	Pose estimation	Loop closures and global map refinement
ElasticFusion	Dense	Minimizes the geometric and photometric errors between the global surface model and the current RGB-D frame	Utilize the randomized fern encoding [17] for appearance-based place recognition, utilize five cost functions to optimize the deformation graph
LSD-SLAM	Semi-dense	Created an initial depth map using frame-to-frame motion estimate, reduces the photometric error and aligns the reference frame with the current frame	Identifies previously visited areas using FABMAP [19], utilize pose graph optimization to minimize the error
LDSO	Sparse	Maintaining the geometry model used by other direct approaches, jointly optimize for all related parameters	Utilize corner features for loop closure detection with BoW, utilize pose graph optimization to minimize the error

2.2. Indirect methods

Indirect methods rely on detecting, describing, and matching visual features between consecutive frames, for instance [5], [22]–[25]. Feature detection and description are central to feature-based methods. Detectors, such as features from accelerated segment test (FAST), speeded-up robust features (SURF) [26], and oriented FAST and rotated BRIEF (ORB) [27], identify feature points in the image. Figure 3 illustrates examples of extracted features used in feature-based methods. The example of identified ORB features in the outdoor dataset [28] are displayed in Figure 3(a). These features are then described using descriptors that are often utilized to help detect a loop as in [5], [23], [29]. Motion estimation and mapping are achieved by analyzing the correspondence between detected features. Visual odometry computes the relative motion between frames, often using robust techniques like random sample consensus (RANSAC) [30] to eliminate outliers, as seen in [23], [29], [31].

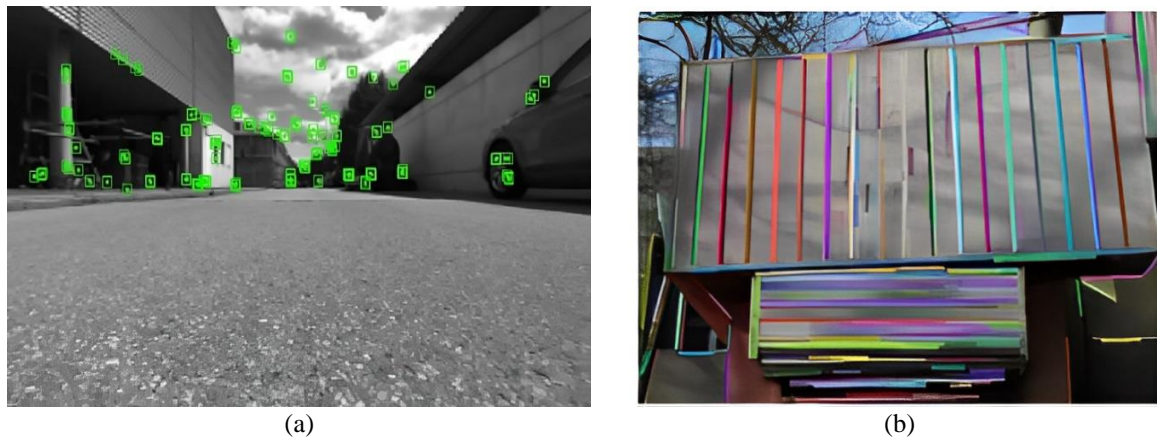


Figure 3. Example of extracted features (a) ORB features in outdoor dataset [28] and (b) of SLD extraction algorithm [25]

The indirect methods can also make use of line features. In order to extract more dependable features in a low-textured environment, Li *et al.* [25] merged point and line features. Compared to point features, line features are more common in outdoor settings and are less impacted by variations in illumination. They reflect organized environments more effectively than point features and provide more important information about the geometric content of an image. Point and line features are extracted in parallel by RPL-SLAM [25] with the ORB method for point features and the straight line segment detector (SLD) algorithm for line segment extraction as shown in Figure 3(b).

Indirect methods usually utilize a sparse approach, where the less detailed map is reconstructed. ORB-SLAM [5] is an example of indirect methods that utilize a sparse approach, where they extract feature points using the ORB algorithm which are oriented multiscale FAST corners with a 256-bit descriptor associated. These extracted ORB descriptors are utilized to create the vocabulary for the place recognition module based on a bag of words (DBoW2) [21], to perform loop detection and re-localization. A covisibility graph is constructed along the process, which is based on the covisibility information between keyframes. This graph is utilized to build an essential graph, i.e., a sparser subgraph of the covisibility to reduce the amount of utilized keyframes. An optimization is performed over the essential graph using the Levenberg-Marquardt algorithm implemented in g2o [32] to maintain global consistency and loop closing. ORB-SLAM has successfully tested on real-world datasets [10], [11], where it can handle loop closure and re-localization effectively as shown in Figure 4.

Regarding the performance of RPL-SLAM, the authors claim that their proposed method outperforms the RGB-D version of ORB-SLAM2 [33] in the majority of sequences across the TUM RGB-D [13] and Imperial College London and National University of Ireland Maynooth (ICL-NUIM) [34] datasets. However, in certain datasets, the positioning accuracy of RPL-SLAM decreases. This issue arises because, in images with rich texture information, false positives in straight-line extraction can occur. These false detections introduce additional system noise, leading to increased errors during computation and a reduction in positioning accuracy. To address this limitation, future research will focus on exploring optimization strategies.

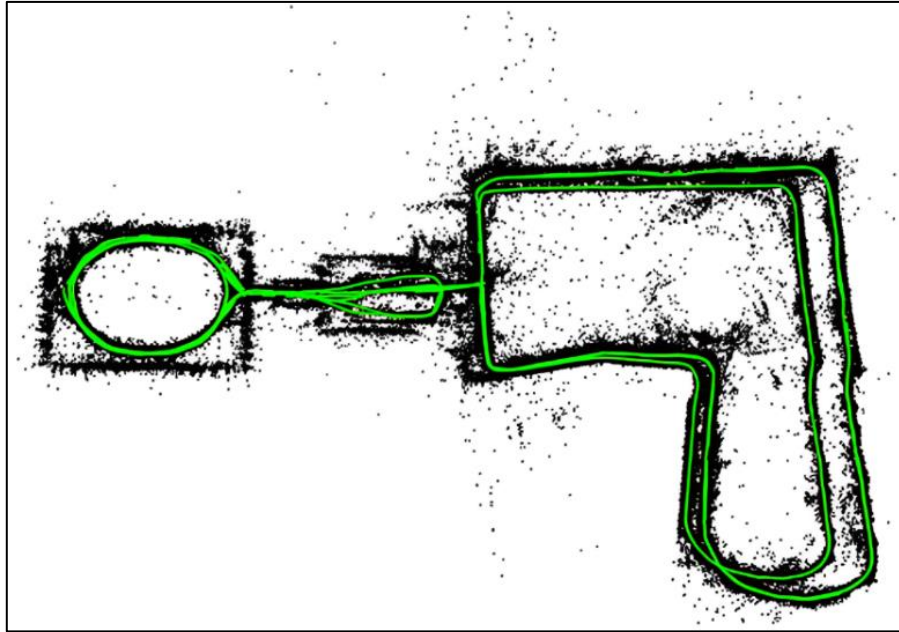


Figure 4. Result of ORB-SLAM in new college dataset [5]

2.3. Semi-direct methods

It is also feasible to combine direct and indirect approaches, as demonstrated in [35]–[37]. A semi-direct visual odometry (SVO) was presented by Foster *et al.* [33] that combines the accuracy and speed of direct methods with the success criteria of feature-based approaches (keyframe selection, parallel tracking and mapping, and tracking numerous features). For motion estimation, their semi-direct method does away with the requirement for expensive feature extraction and reliable matching methods. At high frame speeds, their system achieves subpixel precision by working directly with pixel intensities. 3D points are estimated using a probabilistic mapping approach that explicitly models outlier observations, resulting in fewer outliers and more dependable points. In scenes with minimal, repetitive, and high-frequency texture, robustness is enhanced by precise and high frame-rate motion estimates. A modified parallel tracking and mapping (PTAM) algorithm [38] that can operate in vast areas was used to compare the performance of SVO. PTAM [39] is one of the indirect methods used for micro aerial vehicles (MAVs). According to the study, SVO is a more effective option for visual odometry in MAV applications since it delivers higher accuracy than PTAM.

3. DIFFERENT CAMERA TYPES

Camera-based SLAM algorithms employ various types of camera systems, each offering unique advantages and limitations depending on the application and environment. Monocular cameras are among the most commonly used due to their simplicity, affordability, and compact form factor, and can be utilized for direct [6], [18] and indirect methods [5], [7], [24], [38], [39]. For indirect methods, motion between two consecutive views is determined by solving the epipolar geometry equation. This requires assumptions about the intrinsic camera parameters. Since monocular cameras lack depth information, pose estimation only provides relative motion, not absolute scale. To address this, additional constraints or assumptions, like known object dimensions or scene regularities, are introduced.

Stereo cameras provide two images from left and right perspectives separated by a fixed baseline. These images can be utilized to capture depth information by triangulating corresponding points in the two images. This makes stereo systems inherently more robust in estimating scene geometry and scale compared to monocular systems. Stereo cameras can be utilized for direct, indirect, or semi-direct methods as demonstrated in [8], [36], [40], [41]. Engel *et al.* [8] and Wang *et al.* [40] utilized stereo camera for direct approaches. Stereo LSD-SLAM [8] utilize both static, fixed-baseline stereo and temporal, variable-baseline stereo cues. Their technique uses photometric and geometric residuals at a semi-dense subset of pixels to directly align pictures. When there is enough information available for either static or temporal stereo estimation, these pixels are selected.

The benefits of using a stereo camera are also highlighted in Stereo DSO [40], which combines static stereo with multi-view stereo. Rather than relying on random depth for initialization [6], [15], [18], the

system leverages depth information from static stereo matching, enabling the direct calculation of absolute scale and providing initial depth estimates for multi-view stereo. Qualitative and quantitative evaluations were conducted on the KITTI [10] and Cityscapes [42] datasets, comparing the results with other stereo SLAM methods, such as ORB-SLAM2 [33] and Stereo LSD-SLAM [8]. According to the assessments, Stereo DSO outperforms all other compared approaches in terms of accuracy. Specifically, an analysis on the KITTI dataset shows that Stereo DSO outperforms Stereo ORB-SLAM2 with loop closing and global bundle adjustment, even in the absence of closing big loops. The Stereo DSO result in the KITTI dataset is displayed in Figure 5.



Figure 5. Result of Stereo DSO on sequence 00 of the KITTI dataset [40]

RGB-D cameras, such as Microsoft Kinect or Intel RealSense, provide both color (RGB) and depth (D) information directly. Instead of just employing photometric error, it may also incorporate geometric error to improve performance for direct methods. Both direct and indirect methods may be used with RGB-D cameras, as shown in [16], [25]. The benefits of RGB-D cameras are demonstrated in [25], [33]. It is possible to immediately recover the 3D information of point and line characteristics from the RGB-D pictures that are taken by a depth camera. The precision of camera location is therefore increased as the matching process uses 3D-3D correspondences instead of the 2D-2D correspondences found in conventional RGB cameras.

4. DEEP LEARNING APPLICATIONS

Deep learning techniques have been more and more effective as artificial intelligence has developed, especially in domains like object identification where they provide noticeably greater accuracy [43]. The front end of conventional camera-based SLAM techniques is built on manually designed feature extraction and matching algorithms. These techniques, which each have advantages and disadvantages, usually employ descriptor or Kanade-Lucas-Tomasi (KLT)-based feature tracking. Although KLT tracking is often quicker, it is less resilient to occlusions, age contrast (such as challenging visibility circumstances), and significant perspective shifts (which may be caused by fast camera movements). Longer-term feature monitoring is possible using descriptor-based tracking, but the computational cost is higher. To solve this problem, some researchers use deep learning in the feature extraction and tracking portion of the camera-based SLAM.

Han *et al.* [3] introduced a visual odometry system that leverages convolutional neural networks (CNN) for feature extraction, specifically using the SuperPoint network [44]. This approach replaces traditional hand-engineered feature extraction methods with a CNN-based method, where the comparison is shown in Figure 6. However, the researchers stated that the system failed to perform in a dynamic environment. However, Hamesse *et al.* [45] provide a hybrid visual-inertial odometry (VIO) system that combines a conventional visual-inertial optimization back end with a deep feature matching front end. Based on SuperPoint and LightGlue [46] neural networks, the authors created a feature tracker that can be directly

linked to the VINS-Mono [29] estimation back-end. The system outperforms the standard VINS-Mono, according to extensive testing on Vicon room and EuRoC [14] machine hall datasets.

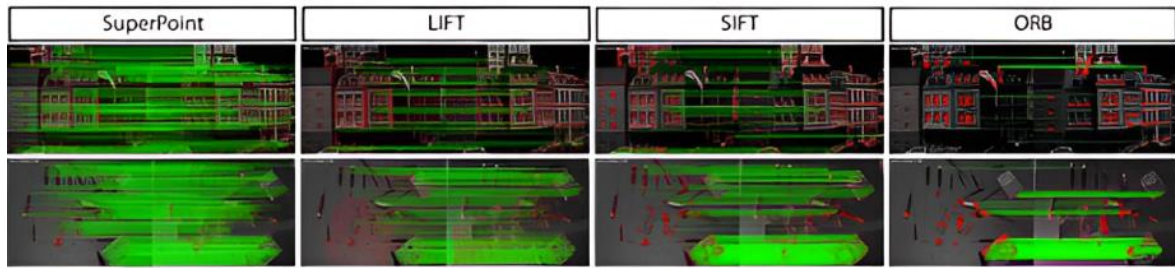


Figure 6. Comparison SuperPoint matching ability to other traditional algorithms [44]

Traditional SLAM systems typically rely on low-level geometric features, which can result in failures in recognizing loop closures in environments with repetitive or unclear visual information. For loop closure detection, several methods also have been proposed to improve SLAM performance. Chen *et al.* [47] proposed a method to improve the performance of traditional ORB-SLAM2 [33] by incorporating semantic information through the Mask R-CNN model. The Mask R-CNN model detects objects in the image, provides semantic labels, and gives a high-quality segmentation result to the object. On the other hand, Dai *et al.* [48] utilized Resnet34 to detect loop closures.

For handling the problem of performing in a dynamic environment, Xinguang *et al.* [49] introduced an enhanced visual SLAM system designed for dynamic environments, based on an improved Mask R-CNN neural network. The proposed SLAM algorithm leverages the semantic segmentation capabilities of the modified Mask R-CNN to differentiate between static and dynamic parts of the scene as shown in Figure 7. Subsequently, the dynamic feature points are disregarded by the algorithm that detects motion consistency and estimates the camera's pose by static feature points in the static region. Fu *et al.* [50] also proposed a method for dealing with dynamic environments by integrating Mask R CNN with an attention mechanism. The researchers integrated the convolutional block attention module (CBAM) into the Mask R-CNN network to enhance dynamic object segmentation. These dynamic object removal methods are then combined with ORB-SLAM2 [33]. However, both proposed methods remain slow, even with GPU acceleration, making them unreliable for real-time applications.



Figure 7. Example of segmentation scenarios in SLAM systems [49]

Deep learning methods have also been applied directly as the method in VO as in [51], [52]. Wang *et al.* [52] proposed a novel DL-based monocular VO algorithm by leveraging deep recurrent convolutional neural networks (RCNNs) [53], which is the first end-to-end approach on the monocular VO through deep neural networks (DNNs). By leveraging the geometric feature representation that CNN has learned, they suggested an RCNN architecture that allows the DL-based VO technique to be applied to whole new contexts. The KITTI dataset is used to assess this VO's performance, and it yields a trajectory that is quite precise and compatible with the ground reality as shown in Figure 8.

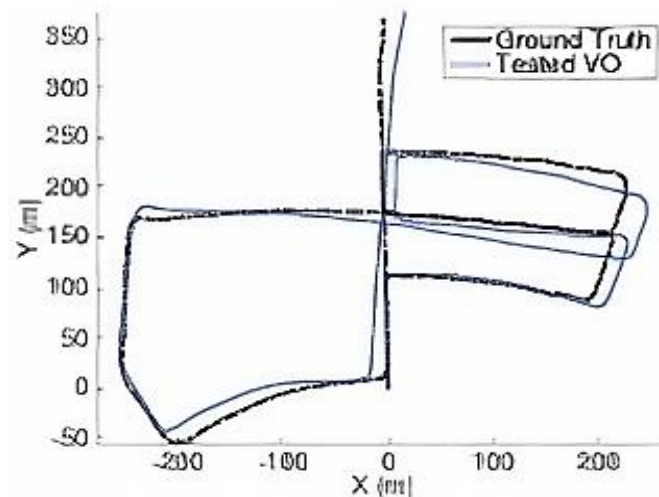


Figure 8. Estimated VO on sequence 05 of KITTI dataset [52]

5. FUTURE RESEARCH DISCUSSION

Environmental elements including illumination, motion blur, and scene texture can have an impact on camera-based SLAM systems. Due to the effect of dynamic objects, it is also challenging to perform well in dynamic situations. Deep learning methods have been utilized to handle dynamic objects, but real-time performance is hard to achieve. Additionally, to increase processing speed without compromising accuracy, optimization strategies like GPU acceleration and sparse representations are used. Despite these developments, there is still a challenge in striking a balance between robust handling of dynamic aspects and real-time speed, which motivates continued study into algorithm efficiency and adaptation to a variety of real-world situations

6. CONCLUSION

Past research on simultaneous localization and mapping (SLAM) has achieved significant advancements through traditional techniques like indirect and direct methods, which leverage robust geometric and photometric data for accurate localization and mapping. Innovations in loop closure detection and the use of various camera systems, such as monocular, stereo, and RGB-D, have addressed challenges like scale ambiguity, depth estimation, and scene complexity, with trade-offs between simplicity and spatial information richness. The integration of deep learning has further revolutionized SLAM by enhancing feature extraction, environment understanding, and dynamic object segmentation, enabling improved robustness and adaptability in complex scenarios. Despite these advancements, achieving real-time performance and scalability in diverse, complex environments remains a major challenge.

FUNDING INFORMATION

This work is supported by Universitas Gadjah Mada under the funding of the Final Project Recognition Program (RTA, contract number: 5286/UN1.P1/PT.01.03/2024) for the year 2024.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Anak Agung Ngurah Bagus Dwimantara	✓				✓		✓		✓		✓			
Oskar Natan	✓	✓		✓	✓	✓		✓		✓		✓	✓	
Novelio Putra Indarto	✓				✓			✓	✓					
Andi Dharmawan		✓			✓	✓				✓		✓		✓

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

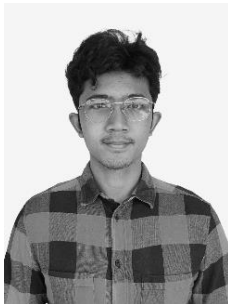
REFERENCES

- [1] B. Gao, H. Lang, and J. Ren, "Stereo visual SLAM for autonomous vehicles: A review," in *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, Oct. 2020, vol. 2020-Octob, pp. 1316–1322. doi: 10.1109/SMC42975.2020.9283161.
- [2] C. Pang, L. Zhou, and X. Huang, "A low-cost 3D SLAM system integration of autonomous exploration based on Fast-ICP enhanced LiDAR-inertial odometry," *Remote Sensing*, vol. 16, no. 11, p. 1979, May 2024, doi: 10.3390/rs16111979.
- [3] X. Han, Y. Tao, Z. Li, R. Cen, and F. Xue, "SuperPointVO: A lightweight visual odometry based on CNN feature extraction," in *Proceedings - 5th International Conference on Automation, Control and Robotics Engineering, CACRE 2020*, Sep. 2020, pp. 685–691. doi: 10.1109/CACRE50138.2020.9230348.
- [4] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600. doi: 10.1109/cvpr.1994.323794.
- [5] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015, doi: 10.1109/TRO.2015.2463671.
- [6] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8690 LNCS, no. PART 2, Springer International Publishing, 2014, pp. 834–849. doi: 10.1007/978-3-319-10605-2_54.
- [7] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007, doi: 10.1109/TPAMI.2007.1049.
- [8] J. Engel, J. Stückler, and D. Cremers, "Large-scale direct SLAM with stereo cameras," in *IEEE International Conference on Intelligent Robots and Systems*, Sep. 2015, vol. 2015-December, pp. 1935–1942. doi: 10.1109/IROS.2015.7353631.
- [9] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for RGB-D cameras," in *Proceedings - IEEE International Conference on Robotics and Automation*, May 2013, pp. 3748–3754. doi: 10.1109/ICRA.2013.6631104.
- [10] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2012, pp. 3354–3361. doi: 10.1109/CVPR.2012.6248074.
- [11] M. Ramezani, Y. Wang, M. Camurri, D. Wisth, M. Mattamala, and M. Fallon, "The newer college dataset: Handheld LiDAR, inertial and vision with ground truth," in *IEEE International Conference on Intelligent Robots and Systems*, Oct. 2020, pp. 4353–4360. doi: 10.1109/IROS45743.2020.9340849.
- [12] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," *arXiv:1607.02555*, Jul. 2016.
- [13] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE International Conference on Intelligent Robots and Systems*, Oct. 2012, pp. 573–580. doi: 10.1109/IROS.2012.6385773.
- [14] M. Burri et al., "The EuRoC micro aerial vehicle datasets," *International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, Jan. 2016, doi: 10.1177/0278364915620033.
- [15] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 3, pp. 611–625, Mar. 2018, doi: 10.1109/TPAMI.2017.2658577.
- [16] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "ElasticFusion: Real-time dense SLAM and light source estimation," *International Journal of Robotics Research*, vol. 35, no. 14, pp. 1697–1716, Sep. 2016, doi: 10.1177/0278364916669237.
- [17] B. Glocker, J. Shotton, A. Criminisi, and S. Izadi, "Real-time RGB-D camera relocation via randomized ferns for keyframe encoding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 21, no. 5, pp. 571–583, May 2015, doi: 10.1109/TVCG.2015.2463671.

- 10.1109/TVCG.2014.2360403.
- [18] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Proceedings of the IEEE International Conference on Computer Vision*, Dec. 2013, pp. 1449–1456. doi: 10.1109/ICCV.2013.183.
 - [19] A. Glover, W. Maddern, M. Warren, S. Reid, M. Milford, and G. Wyeth, "OpenFABMAP: An open source toolbox for appearance-based loop closure detection," in *Proceedings - IEEE International Conference on Robotics and Automation*, May 2012, pp. 4730–4735. doi: 10.1109/ICRA.2012.6224843.
 - [20] X. Gao, R. Wang, N. Demmel, and D. Cremers, "LDSO: Direct sparse odometry with loop closure," in *IEEE International Conference on Intelligent Robots and Systems*, Oct. 2018, pp. 2198–2204. doi: 10.1109/IROS.2018.8593376.
 - [21] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, Oct. 2012, doi: 10.1109/TRO.2012.2197158.
 - [22] H. Cho, E. K. Kim, and S. Kim, "Indoor SLAM application using geometric and ICP matching methods based on line features," *Robotics and Autonomous Systems*, vol. 100, pp. 206–224, Feb. 2018, doi: 10.1016/j.robot.2017.11.011.
 - [23] A. Kasyanov, F. Engelmann, J. Stuckler, and B. Leibe, "Keyframe-based visual-inertial online SLAM with relocalization," in *IEEE International Conference on Intelligent Robots and Systems*, Sep. 2017, vol. 2017-September, pp. 6662–6669. doi: 10.1109/IROS.2017.8206581.
 - [24] H. Lim, J. Lim, and H. J. Kim, "Real-time 6-DOF monocular visual SLAM in a large-scale environment," in *Proceedings - IEEE International Conference on Robotics and Automation*, May 2014, pp. 1532–1539. doi: 10.1109/ICRA.2014.6907055.
 - [25] D. Li *et al.*, "A SLAM system based on RGBD image and point-line feature," *IEEE Access*, vol. 9, pp. 9012–9025, 2021, doi: 10.1109/ACCESS.2021.3049467.
 - [26] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, Jun. 2008, doi: 10.1016/j.cviu.2007.09.014.
 - [27] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proceedings of the IEEE International Conference on Computer Vision*, 2011, pp. 2564–2571. doi: 10.1109/ICCV.2011.6126544.
 - [28] F. Nobis, O. Papanikolaou, J. Betz, and M. Lienkamp, "Persistent map saving for visual localization for autonomous vehicles: An ORB-SLAM 2 extension," in *2020 15th International Conference on Ecological Vehicles and Renewable Energies, EVER 2020*, Sep. 2020, pp. 1–9. doi: 10.1109/EVER48776.2020.9243094.
 - [29] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018, doi: 10.1109/TRO.2018.2853729.
 - [30] J. M. Martínez-Otzeta, I. Rodríguez-Moreno, I. Mendiola, and B. Sierra, "RANSAC for robotic applications: A survey," *Sensors*, vol. 23, no. 1, p. 327, Dec. 2023, doi: 10.3390/s23010327.
 - [31] J. Stalbaum and J. B. Song, "Keyframe and inlier selection for visual SLAM," in *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence, URAI 2013*, Oct. 2013, pp. 391–396. doi: 10.1109/URAI.2013.6677295.
 - [32] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization," in *Proceedings - IEEE International Conference on Robotics and Automation*, May 2011, pp. 3607–3613. doi: 10.1109/ICRA.2011.5979949.
 - [33] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017, doi: 10.1109/TRO.2017.2705103.
 - [34] A. Handa, T. Whelan, J. McDonald, and A. J. Davison, "A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM," in *Proceedings - IEEE International Conference on Robotics and Automation*, May 2014, pp. 1524–1531. doi: 10.1109/ICRA.2014.6907054.
 - [35] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proceedings - IEEE International Conference on Robotics and Automation*, May 2014, pp. 15–22. doi: 10.1109/ICRA.2014.6906584.
 - [36] X. Zhao, L. Liu, R. Zheng, W. Ye, and Y. Liu, "A robust stereo feature-aided semi-direct SLAM system," *Robotics and Autonomous Systems*, vol. 132, p. 103597, Oct. 2020, doi: 10.1016/j.robot.2020.103597.
 - [37] X. Dong, L. Cheng, H. Peng, and T. Li, "FSD-SLAM: a fast semi-direct SLAM algorithm," *Complex and Intelligent Systems*, vol. 8, no. 3, pp. 1823–1834, Mar. 2022, doi: 10.1007/s40747-021-00323-y.
 - [38] S. Weiss *et al.*, "Monocular vision for long-term micro aerial vehicle state estimation: A compendium," *Journal of Field Robotics*, vol. 30, no. 5, pp. 803–831, Aug. 2013, doi: 10.1002/rob.21466.
 - [39] G. Klein and L. Murray, "Parallel tracking and mapping for small AR workspaces," in *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR*, Nov. 2007, pp. 225–234. doi: 10.1109/ISMAR.2007.4538852.
 - [40] R. Wang, M. Schorwer, and D. Cremers, "Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras," in *Proceedings of the IEEE International Conference on Computer Vision*, Oct. 2017, vol. 2017-October, pp. 3923–3931. doi: 10.1109/ICCV.2017.421.
 - [41] N. Krombach, D. Droschel, and S. Behnke, "Combining feature-based and direct methods for semi-dense real-time stereo visual odometry," in *Advances in Intelligent Systems and Computing*, vol. 531, Springer International Publishing, 2017, pp. 855–868. doi: 10.1007/978-3-319-48036-7_62.
 - [42] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2016, vol. 2016-December, pp. 3213–3223. doi: 10.1109/CVPR.2016.350.
 - [43] B. R. Solunke and S. R. Gengaje, "A review on traditional and deep learning based object detection methods," Mar. 2023. doi: 10.1109/ESCI56872.2023.10099639.
 - [44] D. Detone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: Self-supervised interest point detection and description," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2018, vol. 2018-June, pp. 337–349. doi: 10.1109/CVPRW.2018.00060.
 - [45] C. Hamesse, M. Vlaminck, H. Luong, and R. Haelterman, "Practical deep feature-based visual-inertial odometry," in *International Conference on Pattern Recognition Applications and Methods*, 2024, vol. 1, pp. 240–247. doi: 10.5220/0012320200003654.
 - [46] P. Lindenberger, P. E. Sarlin, and M. Pollefeys, "LightGlue: Local feature matching at light speed," in *Proceedings of the IEEE International Conference on Computer Vision*, Oct. 2023, pp. 17581–17592. doi: 10.1109/ICCV51070.2023.01616.
 - [47] X. Chen, Z. Zhou, W. Liang, and M. Wang, "A method of performing loop closing using Mask R-CNN model in SLAM system," in *2019 IEEE International Conference on Mechatronics and Automation (ICMA)*, Aug. 2019, pp. 1035–1040. doi: 10.1109/ICMA.2019.8816547.
 - [48] K. Dai, L. Cheng, R. Yang, and G. Yan, "Loop closure detection using KPCA and CNN for visual SLAM," in *2021 40th Chinese Control Conference (CCC)*, Jul. 2021, pp. 8088–8093. doi: 10.23919/CCC52363.2021.9550432.
 - [49] X. Zhang, X. Wang, and R. Zhang, "Dynamic semantics SLAM based on improved Mask R-CNN," *IEEE Access*, vol. 10, pp.




- 126525–126535, 2022, doi: 10.1109/ACCESS.2022.3226212.
- [50] Y. Fu, B. Han, Z. Hu, X. Shen, and Y. Zhao, "CBAM-SLAM: A semantic SLAM based on attention module in dynamic environment," in *Proceedings of 2022 6th Asian Conference on Artificial Intelligence Technology, ACAIT 2022*, Dec. 2022, pp. 1–6, doi: 10.1109/ACAIT56212.2022.10137973.
- [51] G. Costante, M. Mancini, P. Valigi, and T. A. Ciarfuglia, "Exploring representation learning with CNNs for frame-to-frame ego-motion estimation," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 18–25, Jan. 2016, doi: 10.1109/LRA.2015.2505717.
- [52] S. Wang, R. Clark, H. Wen, and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," in *Proceedings - IEEE International Conference on Robotics and Automation*, May 2017, pp. 2043–2050. doi: 10.1109/ICRA.2017.7989236.
- [53] J. Donahue *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 2625–2634. doi: 10.1109/CVPR.2015.7298878.

BIOGRAPHIES OF AUTHORS






Anak Agung Ngurah Bagus Dwimantara    is currently pursuing a B.Sc. degree in electronics and instrumentation at Universitas Gadjah Mada. From November 2021 to November 2023, he was part of Gadjah Mada Robotics Team, where he contributed as a programmer. From September 2023 to January 2024, he was an exchange student in the International Program on Artificial Intelligence Department, I-Shou University, Taiwan. He can be contacted at a.a.ngurah.bagus.dwimantara@mail.ugm.ac.id.






Oskar Natan    received the B.A.Sc. degree in electronics engineering and the M.Eng. degree in electrical engineering from Politeknik Elektronika Negeri Surabaya, Indonesia, in 2017 and 2019, respectively, and the Ph.D. (Eng.) degree in computer science and engineering from Toyohashi University of Technology, Japan, in 2023. Since January 2020, he has been affiliated with the Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia, first as a lecturer and currently an assistant professor. His research interests include sensor fusion, hardware acceleration, and end-to-end systems. He is a member of the IEEE ITS Society, the IEEE-RA Society, and Indonesian Computer, Electronics, and Instrumentation Support Society (IndoCEISS). He has been serving as a reviewer for some reputable journals and conferences, including, IEEE Transactions on Intelligent Vehicles, IEEE Transactions on Intelligent Transportation Systems, IEEE ICRA, and IEEE/RSJ IROS. He can be contacted at oskarnatan@ugm.ac.id.



Novelio Putra Indarto    received his B.Sc. degree in electronics and instrumentation from Universitas Gadjah Mada, Indonesia, in 2023. He is currently pursuing an M.Sc. degree in Electronics and Instrumentation at Universitas Gadjah Mada. His research interests include robotics and machine learning, with a focus on advancing automation and intelligent systems. He can be contacted at novelio.p.i@mail.ugm.ac.id.



Andi Dharmawan    received the B.Sc. degree in electronics and instrumentation, M.Cs. degree in computer science, and Doctor in computer science from Universitas Gadjah Mada, in 2006, 2009, and 2017, respectively. His research interests include unmanned aerial vehicles (UAV), control system, and robotics. In addition to his role as a lecturer, Andi Dharmawan serves as the supervisor of the Gadjah Mada Flying Object Research Center and the Gadjah Mada Robotics Team, fostering innovation and development in aerospace and robotics research. He also holds the position of Secretary in the Department of Computer Science and Electronics at Universitas Gadjah Mada. He is a member of IEEE society and Indonesian Computer, Electronics, and Instrumentation Support Society (IndoCEISS). He can be contacted at andi_dharmawan@ugm.ac.id.