applied sciences

MDPI

Article

PSMD-SLAM: Panoptic Segmentation-Aided Multi-Sensor Fusion Simultaneous Localization and Mapping in Dynamic Scenes

Chengqun Song [1,2,†], Bo Zeng [1,3,†], Jun Cheng [1,2,*], Fuxiang Wu [1,2] and Fusheng Hao [1,2]

[1] CAS Key Laboratory of Human-Machine Intelligence-Synergy Systems, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China; cq.song@siat.ac.cn (C.S.); bo.zeng@siat.ac.cn (B.Z.); fx.wu1@siat.ac.cn (F.W.); fs.hao@siat.ac.cn (F.H.)

[2] The Chinese University of Hong Kong, Hong Kong, China

[3] China Mobile Financial Technology Co., Ltd., Beijing 100045, China

[*] Correspondence: jun.cheng@siat.ac.cn

[†] These authors contributed equally to this work.

Abstract: Multi-sensor fusion is pivotal in augmenting the robustness and precision of simultaneous localization and mapping (SLAM) systems. The LiDAR–visual–inertial approach has been empirically shown to adeptly amalgamate the benefits of these sensors for SLAM across various scenarios. Furthermore, methods of panoptic segmentation have been introduced to deliver pixel-level semantic and instance segmentation data in a single instance. This paper delves deeper into these methodologies, introducing PSMD-SLAM, a novel panoptic segmentation assisted multi-sensor fusion SLAM approach tailored for dynamic environments. Our approach employs both probability propagation-based and PCA-based clustering techniques, supplemented by panoptic segmentation. This is utilized for dynamic object detection and the removal of visual and LiDAR data, respectively. Furthermore, we introduce a module designed for the robust real-time estimation of the 6D pose of dynamic objects. We test our approach on a publicly available dataset and show that PSMD-SLAM outperforms other SLAM algorithms in terms of accuracy and robustness, especially in dynamic environments.

Keywords: SLAM; dynamic scenarios; panoptic segmentation; multi-sensor fusion

check for updates

Citation: Song, C.; Zeng, B.; Cheng, J.; Wu, F.; Hao, F. PSMD-SLAM: Panoptic Segmentation-Aided Multi-Sensor Fusion Simultaneous Localization and Mapping in Dynamic Scenes. *Appl. Sci.* **2024**, *14*, 3843. https://doi.org/10.3390/app14093843

Academic Editor: João M. F. Rodrigues

Received: 27 March 2024
Revised: 22 April 2024
Accepted: 24 April 2024
Published: 30 April 2024

1. Introduction

Simultaneous localization and mapping (SLAM) [1] has long been a pivotal technology in the field of mobile robotics. This process entails the utilization of data derived from robot sensors to acquire state information essential for localization, as well as the construction of maps through sensor data analysis [2]. SLAM systems have been extensively applied in various domains, such as intelligent service robots [3], autonomous driving [4] and augmented reality [5]. SLAM can provide these systems with their location and posture information. SLAM technology can improve the positioning performance of robots, cars, and augmented reality systems, promote the development of these fields, and thus improve people's quality of life.

The advancement of SLAM technology has facilitated significant progress in addressing the SLAM problem within simple and static scenes. This is achieved through the utilization of sensors such as vision-based visual–inertial systems (VINS) [6] or light detection and ranging (LiDAR)-based LiDAR odometry and mapping (LOAM) systems [7]. These methods have demonstrated high robustness and localization accuracy. Consequently, there has been a growing emphasis on the development of SLAM techniques capable of functioning in dynamic and intricate environments. As illustrated in Figure 1, such environments present significant challenges due to the interference from dynamic objects and the degradation of sensors in harsh conditions. The challenges of dynamic and intricate environments in SLAM technology encompass interference from dynamic objects, complicating tracking and localization. Adverse conditions may degrade sensor performance, impacting data quality. The heightened complexity requires robust algorithms, elevating system complexity. Real-time processing is crucial, demanding timely feedback for successful SLAM task execution. Addressing these challenges necessitates advancements in algorithmic sophistication and sensor adaptability to enhance SLAM resilience in dynamic settings.

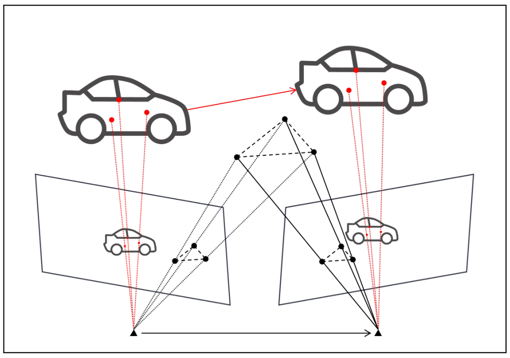


Figure 1. Schematic diagram of the dynamic scene. The triangle in the figure is the camera, the cars represent dynamic objects, the red dots represent dynamic feature points, and the black dots are static feature points. The illustration emphasises significant disparities in the motion of dynamic and static entities as they change positions between two frames, ultimately reflecting the characteristic features of a dynamic scenes.

One approach has attracted considerable interest in tackling these challenges: multi-sensor fusion SLAM. This method integrates data from various sensors, including vision, LiDAR, IMU (inertial measurement unit), and even WIFI signals [8], to improve system performance. LiDAR is an effective tool for navigation and tracking due to its high precision. However, it is constrained by weather conditions, a restricted detection range, and the requirement for color information. These factors can result in subpar loop closure detection in environments such as long corridors and tunnels. Conversely, vision sensors excel in textured environments but are hindered by their lackluster performance in low-texture scenes or rapidly changing illumination conditions. Monocular cameras also face challenges with scale degradation. Despite these limitations, IMU sensors offer high frequency and reliability, albeit with accuracy over short periods being somewhat limited. The potential of multi-sensor fusion SLAM lies in its ability to mitigate these constraints by amalgamating data from various sensors, thereby enhancing both accuracy and robustness.

The second method integrates deep learning-driven semantic data with SLAM to enhance environmental comprehension. Dynamic SLAM systems that incorporate semantic segmentation, as exemplified by [9–11], leverage semantic information to augment the efficacy of SLAM in dynamic environments. Recent advancements in the realm of deep learning and computer hardware have propelled computers to achieve remarkable proficiency in classical image processing tasks. These tasks encompass image classification, object detection, and semantic segmentation. Panoptic segmentation represents an innovative approach that amalgamates these tasks, thereby facilitating pixel-level segmentation and object identification concurrently with object detection. This integration yields additional information for SLAM.

This paper introduces PSMD-SLAM, an innovative SLAM method that incorporates panoptic segmentation-assisted multi-sensor fusion for dynamic environments. This approach builds upon the LVI-SAM framework and significantly improves system performance by utilizing panoptic segmentation to enhance the effectiveness of multi-sensor fusion SLAM.

The primary contributions of this paper encompass the following:

1. Introducing a novel SLAM approach, PSMD-SLAM, which leverages the benefits of multi-sensor fusion and panoptic segmentation for dynamic environments.
2. Implementing dynamic object detection and removal based on panoptic segmentation, utilizing probability propagation-based and principal component analysis-based (PCA-based) clustering methods in visual and LiDAR data, respectively.
3. Proposing a fast and robust dynamic object six-degrees-of-freedom (6D) pose estimation module that fuses panoptic segmentation information with multi-sensor fusion data for accurate estimation.

The remainder of this manuscript is structured as follows: Section 2 delineates the pertinent literature, Section 3 elucidates the proposed framework, Section 4 showcases the experimental outcomes, and the Section 5 offers a synthesis.

2. Related Work

2.1. Multi-Sensor Fusion

Multi-sensor fusion is an essential technique for improving the accuracy and robustness of SLAM systems. Combining data from different sensors, such as cameras, LiDAR, and IMU, can help overcome the limitations of individual sensors and enable a more accurate estimation of the robot's pose.

One approach to multi-sensor fusion involves optimizing and enhancing the original SLAM framework, such as visual SLAM or LiDAR SLAM, by integrating new sensors while keeping the original framework unchanged. For instance, the LiDAR–inertial odometry (LIMO) proposed by Johannes Graeter et al. [12] enhances the visual odometry based on monocular VO by using depth information from LiDAR.

Another promising research direction is to reconstruct the SLAM framework to accommodate multiple sensors, which enables fuller exploitation of the advantages of different sensor types. For instance, Elena López et al. proposed a laser 2D LiDAR–vision–inertial SLAM approach [13], which is a loosely coupled EKF-based algorithm that requires an upper computer to assist in the computation. Similarly, Yuran Liang et al. utilized the error state extended Kalman filter (ESEKF) to fuse multiple sensors of IMU, vision sensors, LiDAR, radar, and GPS in a loosely coupled fusion with some scalability [14]. Additionally, LICFUSION [15] and LICFUSION 2.0 [16] integrate IMU measurements, sparse visual features, and LiDAR features and perform joint state optimization using a multistate constrained Kalman filter (MSCKF) framework. These multi-sensor fusion approaches have shown exciting potential in improving the accuracy and robustness of SLAM systems.

The factor graph model has been introduced and developed in the field of multi-sensor fusion SLAM. Liu et al. [17] proposed a SLAM framework that integrates the information of multiple sensors including a camera, LiDAR, IMU, and global positioning system (GPS) based on vision–LiDAR calibration. Different sensors are fused in a tightly coupled manner and finally optimized by a factor graph. NF-iSAM [18] leverages the expressive power of neural networks and trains normalizing flows to draw samples from the joint posterior of a non-Gaussian factor graph. By utilizing a Bayes tree, NF-iSAM is able to exploit the sparsity structure of SLAM, enabling efficient incremental updates for SLAM problems. Shan et al. proposed an LVI-SAM [19] method, which is a factor graph-based LiDAR–vision–inertial SLAM method that possesses global optimization based on scene re-identification and can keep working when LiDAR or vision sensors fail. Additionally, Wisth et al. proposed a tightly coupled SLAM system [20] that extracts feature from LiDAR and vision to jointly optimize information in a single integrated factor graph, and Lin and Zhang et al. proposed R^{3}LIVE [21], which constructs geometric structures and renders textures using LiDAR–inertial odometry (LIO) and visual–inertial odometry (VIO), respectively, and can reconstruct dense 3D RGB point clouds of the environment in real time.

The drawback of the aforementioned multi-sensor fusion methods is their increased computational complexity and potential challenges in real-time processing, particularly when dealing with large-scale scenes. Integrating data from multiple sensors and optimizing across different modalities can lead to higher computational demands, potentially limiting the real-time performance of the SLAM system, especially in resource-constrained environments.

2.2. Panoptic Segmentation

With the development of deep learning techniques, researchers have been trying to build all-in-one deep learning frameworks that can perform multiple tasks simultaneously. For instance, in the monocular visual sensing task, researchers aim to estimate scene depth, optical flow, and camera pose [22–24]. Similarly, the panoptic segmentation task combines semantic segmentation and instance segmentation [25–27].

Panoptic segmentation is a relatively new computer vision task that aims to segment all objects in an image or video, assigning each pixel to a specific class and instance. It provides a more comprehensive understanding of the visual world by combining instance segmentation and semantic segmentation. The instance segmentation component identifies and segments each individual object in the scene, while the semantic segmentation component labels each pixel with a category, such as "person", "car", or "building".

Initially, panoptic segmentation networks were often not able to operate online and were hardly accurate [25]. However, recent advancements have been made to address these issues. For instance, the panoptic segmentation network proposed by Hou et al. [28] can run at a rate of 10 frames per second but still suffers from accuracy deficiencies. While UPSNet [26] is dedicated to improving the accuracy of segmentation, it has poor real-time performance. On the other hand, panopticFCN [27] strikes a better balance between real-time performance and accuracy, achieving an operation rate of 20 frames per second and an accuracy of more than 40% PQ. This makes it suitable for integration into our panoptic SLAM systems.

2.3. SLAM with Semantic Prior

The ability of deep learning-based image segmentation methods (e.g., Yolov3 [29] and Mask R-CNN [30]) to provide semantic priors has inspired the development of SLAM. In the context of SLAM, "semantic prior" refers to the use of semantic information, such as object categories, to aid in the localization and mapping process. This can be especially useful in dynamic environments where objects are moving and can disrupt the SLAM process.

DynaSLAM [9] uses semantic prior information for dynamic scene verification. When objects in the image are identified as potentially moving, such as people, animals, and cars, features located on these objects are considered dynamic features and removed. While using semantic information to detect dynamic feature points is straightforward, it has limitations. The semantic dynamic feature points do not always match the actual dynamic feature points. To overcome this limitation, some approaches combine semantic information and multi-view geometry.

DS-SLAM [11] is one such method that uses SegNet to obtain semantic labels of feature points in a separate thread and further uses polar geometric constraints to check shift consistency after the feature points are classified as potential dynamic feature points to discriminate whether they are dynamic feature points or not. Dinh-Cuong Hoang [27] applies semantic segmentation and MASK R-CNN networks to implement acquisition and filtering methods for panoptic information to provide pose estimation of objects, respectively.

In addition, there are also object SLAM methods based on semantic prior, such as MaskFusion [31] and SO-SLAM [32]. These methods use semantic prior information to extract and model static objects as landmarks, including rectangular or quadratic curves. Then, they use these landmarks to complete the SLAM process. Of course, some works also incorporate dynamic objects that can be added to the back end of SLAM for joint optimization, such as [33,34]. However, the SLAM approaches mentioned above are typically limited to visual SLAM and are not common in multi-sensor fusion SLAM.

Compared to the aforementioned methods, this paper proposes a novel approach that begins with multi-sensor fusion SLAM and integrates the panoptic segmentation method. The proposed method achieves greater robustness and accuracy in removing the impact of dynamic objects on pose estimation and enables the estimation of the 6D pose of dynamic objects.

3. Method

3.1. PSMS-SLAM Framework

Our approach designs a multi-sensor fusion SLAM framework, PSMS-SLAM, combined with panoptic segmentation, which mainly contains five modules, and its main structure is shown in Figure 2.

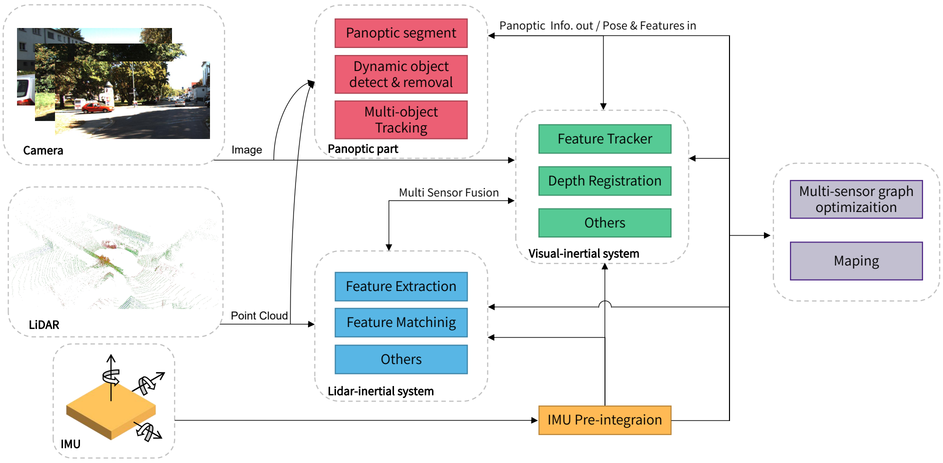


Figure 2. PSMS-SLAM system framework. The panoptic component is denoted in red, the vision component in green, the LiDAR component in blue, the IMU component in orange, and the back end in gray.

The IMU module uses the preintegration [35] method to perform IMU data preprocessing and provide the data to other pipelines. The LiDAR–inertial system module extracts feature points from the LiDAR point cloud, performs feature extraction and matching, and combines the IMU preintegration data to estimate the pose. The panoptic segmentation module is responsible for the panoptic segmentation of the image to obtain the panoptic information and to track the dynamic target pose in real time with this and global pose information. The visual–inertial system module accepts the image data using the Lucas–Kanade (LK) optical flow method to track the feature points of the previous frame, then uses RANSAC for the pairwise polar constraint to remove the exclusion points, and finally extracts the features from accelerated segment test (FAST) feature points in the image. In this module, the visual-based loop-closure, which utilizes the improved version of distributed bags of words library (DBOW2) method known for its high accuracy and robustness, is also implemented. Specifically, the IMU measurement constraint, visual odometry constraint, and LiDAR odometry constraint are jointly optimized using the factor graph method to obtain the final system poses, and the data from each module are accepted to complete the construction of the map.

The modules are also coupled to enhance the performance of each other: the visual–inertial system module receives data from the LiDAR to register the depth information into the feature points, completes the initialization using the estimated poses from the LiDAR–inertial system module, and receives information from the panoptic segmentation to reject the interference of dynamic objects on the accuracy.

3.2. Dynamic Objects Removal

PSMS-SLAM is a multi-sensor framework SLAM method, so below, we describe the algorithm in dynamic scenes in two parts, vision, and LiDAR.

(1) Visual–inertial system:

This part considers the frames that receive the panoptic segmentation information as a panoptic keyframe. Algorithm 1 shows the entire process for eliminating dynamic object feature points in the visual–inertial system.

Algorithm 1 Dynamic object feature points removal in the visual–inertial system

```
Require: Image frame $F$
Ensure: Processed feature points
Track feature points with LK method as point set $\mathbf{S}_1$
if $F$ is not key frame then
    Obtain new feature points as point set $\mathbf{S}_2$
    if Size of $\mathbf{S}_2$ is bigger than a threshold then
        Perform an LK optical flow match between the current frame and the nearest panoptic keyframe
        Obtain the dynamic probability of successfully matching feature points in $\mathbf{S}_2$ and add them to $\mathbf{S}_1$
    end if
else
    Fusion of panoptic segmentation information and image into a probability image
    Calculate the dynamic probability $p'_t$ of each feature point by local convolution
    Calculate the dynamic probability $p_t$ of all feature points by probability propagation
end if
```

When the panoptic segmentation information is obtained from the panoptic keyframes, we first map to a probability image according to the object categories. As described in [33], we define the dynamic probability of categories such as people and cars as 1.0 and the dynamic probability of buildings as 0. However, the feature point extraction window may capture some dynamic feature points outside the dynamic region of the image, which will move with the dynamic object.

To calculate the dynamic probability of feature points, we use a convolution kernel of size k that is aligned with the coordinates of the feature points to perform a local convolution.

$$M_{ij} = 1 - (\frac{1}{2})^{k-d_{ij}} \tag{1}$$

where k is the window size for feature point extraction. d_{ij} is the Manhattan distance from M_{ij} to the centre.

We then take the maximum value of the local convolution's result matrix as the feature points' dynamic probability.

To address the occasional false and missed detections in panoptic segmentation, we use a probabilistic propagation method to propagate the probability between adjacent panoptic keyframes, as shown in the following equation:

$$p_t = (1-\alpha)p_{t-1} + \alpha p'_t \tag{2}$$

Here, p_t is the final result of probability at panoptic key frame t. p'_t is the calculation result of panoptic key frame t. α is a probabilistic propagation parameter with values ranging from 0.2 to 0.5 and decreases as the camera frame rate increases.

We use the Lucas–Kanade optical flow method to track the obtained matches and obtain their dynamic probabilities for feature points that are not on panoptic keyframes. However, there may be cases where feature points are not detected in new image frames when tracked using the LK method. To address this, we extract new feature points for most frames in the visual–inertial process to reach a certain number. These new feature points are ignored until a new keyframe arrives, or when their number reaches a threshold, we use the LK optical flow method to perform matching and obtain the feature point matching information between that frame and the nearest panoptic keyframe. We then use the previous method to determine the dynamic probability. If a few feature points for which the match fails, these points will be ignored until the next panoptic key frame arrives.

(2) LiDAR–inertial system:

For the LiDAR–inertial system, we employ the feature extraction method from [19] to obtain two types of feature points: edge feature points and plane feature points.

For the edge feature points on the LiDAR frame i, we use the nearest-neighbor algorithm to find the five nearest feature points and then determined whether they were collinear. If they formed a straight line, we use a point-independent representation to represent the line using its normalized direction vector (a, b, c).

For the n edge feature points on frame i, we obtain n vectors $l_{1,i}$ to $l_{n,i}$. Next, we match the edge feature points of frame i with those of frame $i + 1$ using the scan-to-scan method from [7] to obtain the corresponding straight lines $l_{1,i+1}$ to $l_{n,i+1}$. We combine the data from m frames into a matrix M_l of the form

$$\begin{bmatrix} l_{1,i} & l_{1,i} & \cdots & l_{1,i+m} \\ l_{2,i+1} & l_{2,i+1} & \cdots & l_{2,i+m} \\ \vdots & \vdots & \ddots & \vdots \\ l_{n,i+m} & l_{n,i+1} & \cdots & l_{n,i+m} \end{bmatrix} \tag{3}$$

The method for the plane feature points is similar. For the plane feature points on frame i of the LiDAR frame, we use the nearest-neighbor algorithm to find the five nearest feature points and then determine whether they belong to the same plane. If so, we use (a, b, c, d) to represent the plane, where $ax + by + cz + d = 0$ is the standard form of the plane equation. We obtain a matrix M_p about the planes

$$\begin{bmatrix} p_{1,i} & p_{1,i} & \cdots & p_{1,i+m} \\ p_{2,i+1} & p_{2,i+1} & \cdots & p_{2,i+m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n,i+m} & p_{n,i+1} & \cdots & p_{n,i+m} \end{bmatrix} \tag{4}$$

We then use PCA to process the two matrices M_l and M_p. We perform SVD decomposition to obtain $dim \times m$ eigenvalues and eigenvectors for the two matrices, where m is the number of frames used, and dim is the dimension described (3 for M_l and 4 for M_p). The eigenvectors are then sorted in descending order of their corresponding eigenvalues to construct an orthogonal basis. We project the original data onto this basis to obtain the new data matrices X_l and X_p. Next, we compute the sample covariance matrices S_p and S_l using the following equation:

$$S = \frac{1}{n-1} \sum_{i=1}^{n} (X_{*,i} - \bar{X})(X_{*,i} - \bar{x})^T \tag{5}$$

where $X_{*,i}$ is the i-th column of X and n is the number of samples. Then, we use the following equations to calculate Hotelling's T-squared statistic value.

$$T^2 = n(\bar{x} - \mu)^T S^{-1} (\bar{X} - \mu) \tag{6}$$

where $\bar{x}$ is the sample mean of X, μ is the overall mean, and S is the sample covariance matrix. Then the clustering of T_l^2 and T_p^2 is performed using thedensity-based spatial clustering of applications with noise (DBSCAN) method. For the clustering results, we use the following equations to map these feature points extracted by LiDAR to the panoptic segmentation keyframes and find the corresponding class cases of the feature points.

$$p(u, v, 1) = F_{\text{LiDAR}}^{\text{World}} P_{\text{LiDAR}}(X, Y, Z) \tag{7}$$

When the ratio of static to dynamic feature points of a cluster is above 5, we identify it as a static cluster and the others as dynamic clusters. Figure 3 shows the results of the clustering algorithm, where it can be seen that the static principal component clustering works very well.

In the subsequent module of the LIDAR inertial system, we use only the feature points in the static cluster for pose estimation and map building.

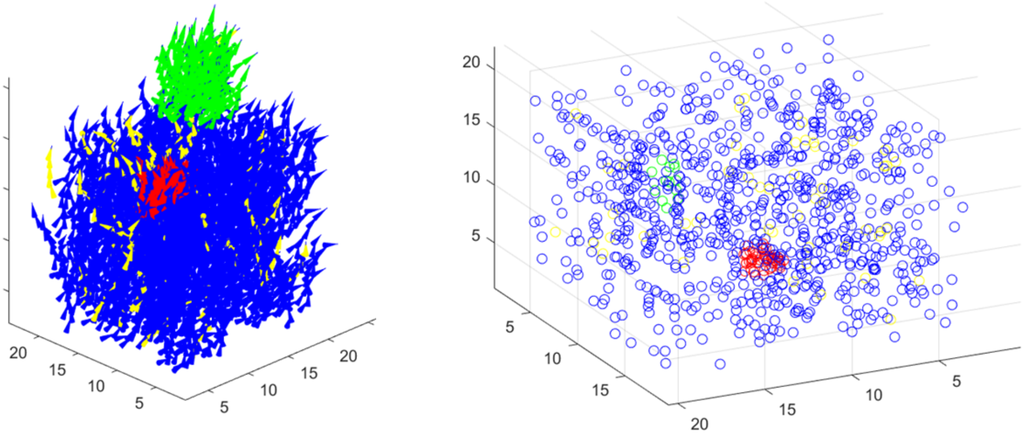


Figure 3. Cluster analysis outcome. On the left, the clustering results are presented, juxtaposed with the initial feature distribution on the right. The color-coded legend distinguishes static features in blue, features linked to one dynamic object in red, those associated with another dynamic object in green, and noise in yellow.

3.3. *Dynamic Object Pose Estimation Assisted by Multi-Sensor Fusion*

The dynamic object pose estimation, and the SLAM process are mutually reinforcing. Furthermore, we construct a dynamic object 6D pose estimation module in the proposed SLAM framework, whose main framework structure is shown in Figure 4.

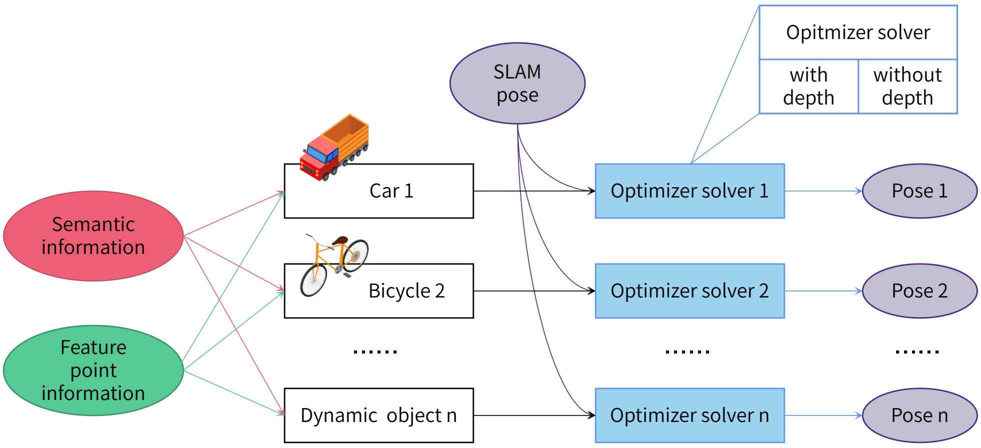


Figure 4. Framework of dynamic object pose estimation. The framework integrates various components, leveraging multi-sensor fusion, panoptic segmentation, and optimizer solvers for accurate and real-time estimation of the 6D pose of dynamic objects within the environment.

This module uses panoptic information to identify individual dynamic objects within the sensor's field of view and associate feature points with them. By exploiting the existing odometer's matches information of points, the module naturally acquires the matching relationship of dynamic objects between consecutive multiple frames of motion. To estimate the 6D positional information of dynamic objects, we input the SLAM system's body pose and the coordinates of continuously tracked feature points into a nonlinear optimization solver. Details of the solver's design are provided below. The dynamic object poses estimation, and SLAM processes mutually reinforce each other, improving the overall system's accuracy.

Benefiting from the advantages of multi-sensor fusion, when the dynamic object is in the LiDAR field-of-view range, we can obtain the depth information of the corresponding point by the following equation.

$$\begin{aligned} p_{\text{unit}}(\theta,\phi) &= f(P_{LiDAR}) \\ \phi &= \alpha + \frac{c \times s_v}{C} \\ \theta &= i \times \delta\theta, \end{aligned} \tag{8}$$

where c the channel of point, C the total number of channels of LiDAR, s_v the scale of vertical FOV, α the angle of first LiDAR ring, p_{unit} the coordinate of unit sphere, i the sequence of point, $\delta\theta$ the angle increment.

After acquiring a certain number of points with depth information, we can construct the following least squares method to find the pose of the corresponding dynamic target.

$$\min \sum_{i=1}^{m} \frac{1}{2} \| f(\widehat{p}_{\{i,k\}}) - T_d T_c f(\widehat{p}_{\{i,k+1\}}) \|_2^2 \tag{9}$$

However, the dynamic object may only sometimes be within the sensing range of the LiDAR, resulting in a lack of point depth information. To address this, the paper also proposes a method for estimating the pose of dynamic objects without depth information.

The pose estimation becomes particularly difficult for monocular cameras that lack depth information. The traditional para-polar geometry method does not apply to dynamic object pose estimation because the reverse-projected rays are disjoint. The second is the problem of scale information. The monocular camera needs scale information, and the relative scale of dynamic objects in the camera range is also a challenge that must be addressed.

Generally, SLAM systems use cameras with frame rates of 10 to 20 frames or even higher, so we assume that the 6D positional transformation T_d of the dynamic object is the same in three consecutive frames, as shown in Figure 5. In such a short time interval, even if the acceleration reaches 1G, the change in its velocity is still less than 2 m/s. For a slower car, the change in its velocity is even more minor. Therefore, we can express it by the following equation.

$$P_{\{i,k\}} = T_d P_{\{i,k+1\}} = T_d^2 P_{\{i,k+2\}} \tag{10}$$

where $P_{\{i,k\}} = [x,y,z]^T$ is the coordinate of the point P_i on the k th frame.

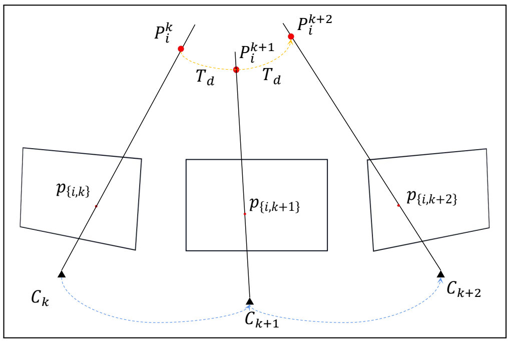


Figure 5. Schematic diagram illustrating the 6D pose consistency in dynamic object pose estimation. The dynamic object depicted in the figure maintains the same 6D pose across three consecutive frames.

Then, for the complete set of points $\mathbf{P}(Pi, 0 < i \leq n)$ of dynamic objects, a least squares problem is similarly constructed, and the problem is solved using nonlinear optimization as follows:

$$\min \sum_{i=1}^{n} \sum_{t=0}^{2} \| p_{\{i,k+t\}} - h\left[(T_d)^t P_{i,k}^{w} \right] \|_2^2 \tag{11}$$

The variable to be solved is T_d, where $p_{\{i,k+t\}}$ is the pixel coordinate (u, v) of the point P_i at the $k + t$ th frame image; h represents the camera model function, where the camera internal reference, correction factor, and camera pose are preset values.

3.4. Panoptic Segmentation

Our method employs panopticFCN as a panoptic segmentation network, which can obtain both pixel-level instance information and semantic information. By training on the COCO dataset, we can identify 80 thing classes and 53 stuff classes and can add a unique identity to each of them. Based on these segmentation results, we can know the priori properties of each pixel in the image. For example, pavement and wall classes hardly produce motion and changes, while humans and vehicles tend to produce motion and appearance changes.

Panoptic segmentation eventually returns a pixel-level image segmentation result, as shown in Figure 6, as well as a set of data information recording the category of things and object numbers corresponding to each region. This information is fed into the other modules as a data stream, which also presents a potential pipeline synchronisation problem. On the other hand, there is the possibility that the image segmentation rate is lower than the image frame input rate when the device's graphics performance is inferior.

In addressing this problem, we skip some of the input image frames according to the rate of image segmentation. On the other hand, the other modules receive the discrete image segmentation data and carry out some processing to avoid the system going into a blocking state, as described in Section 3.2.

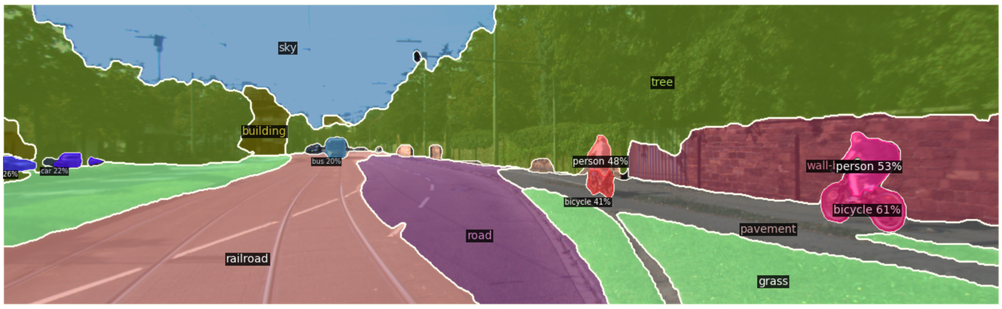


Figure 6. Panoptic segmentation image result.

4. Experiment

4.1. Experimental Description

PSMS-SLAM is evaluated using a series of experiments comparing the performance evaluation of classical SLAM methods on the same computer equipped with an AMD 5950X CPU, which is manufactured by Advanced Micro Devices, Inc. (AMD, Santa Clara, CA, USA), with the silicon die of the chip fabricated at the foundry of Taiwan Semiconductor Manufacturing Company (TSMC) in Hsinchu, Taiwan, China. The computer also includes a GeForce 3090 GPU, manufactured by NVIDIA Corporation (Santa Clara, CA, USA), with the silicon die of the GPU fabricated at the foundry of Samsung in Hwaseong and Pyeongtaek, South Korea. All programs in the experiments were executed using the Robot Operating System (ROS).

The M2DGR dataset [23] is a multi-modal and multi-scene SLAM dataset proposed by Shanghai Jiao Tong University. It comprises a diverse sensor suite, including six fisheye cameras, an upward-facing RGB fisheye camera, an infrared camera, an event camera, a visual–inertial sensor, an IMU, a 32-line LiDAR, a consumer-grade GNSS receiver, and a GNSS-IMU navigation system with real-time kinematic (RTK) signals. Ground truth trajectories are provided by laser total station, motion capture system, RTK signals, and high-precision IMUs. The dataset includes data sequences from indoor and outdoor environments, alternating indoor–outdoor environments, and various complex scenarios, such as night scenes, twilight scenes, and completely dark indoor scenes, which make it highly challenging. The M2DGR dataset provides a flexible experimental environment that can accommodate a variety of SLAM approaches with different sensor configuration requirements. Table 1 shows the details of the different sequences in the M2DGR dataset.

Table 1. M2DGR dataset details.

Dataset	Distance (km)	Time (s)	Scene Classification
walk 01	3.724	454	static
Street 02	2.453	110	dynamic, long straight
Street 06	5.067	485	dynamic
Street 10	0.561	80	static
Gate 03	0.393	27	long straight

In addition, the panoptic segmentation network PanopticFCN is trained only on the COCO dataset to weed out problems such as overfitting caused by deep learning.

4.2. Comparative Evaluation

Table 2 shows the performance evaluation results on the M2DGR dataset. We compare four different methods: LeGO-LOAM, VINS-Mono, LVI-SAM, and our proposed method, PSMS-SLAM. The evaluation is performed on five sequences: Walk 01, Street 02, Street 06, Street 10, and Gate 03.

Table 2. Comparison of accuracy based on the absolute trajectory error (ATE) in meters on the M2DGR dataset.

M2DGR	Walk 01	Street 02	Street 06	Street 10	Gate 03
LeGO-LOAM [36]	3.29	20.02	1.25	17.93	0.12
VINS-Mono [6]	57.90	24.16	124.36	23.57	6.25
LVI-SAM [19]	3.28	3.53	0.43	3.95	0.11
PSMS-SLAM	3.30	3.14	0.45	2.87	0.11

As illustrated in Table 2, our method yields competitive or superior trajectory accuracy compared with the other three methods. In particular, our method obtains comparable results to LeGO-LOAM and LVI-SAM on Walk 01 and Gate 03, respectively. Furthermore, our method achieves state-of-the-art performance on Street 10 with an ATE of merely 2.87 m, demonstrating that our method significantly outperforms LVI-SAM in a dynamic environment.

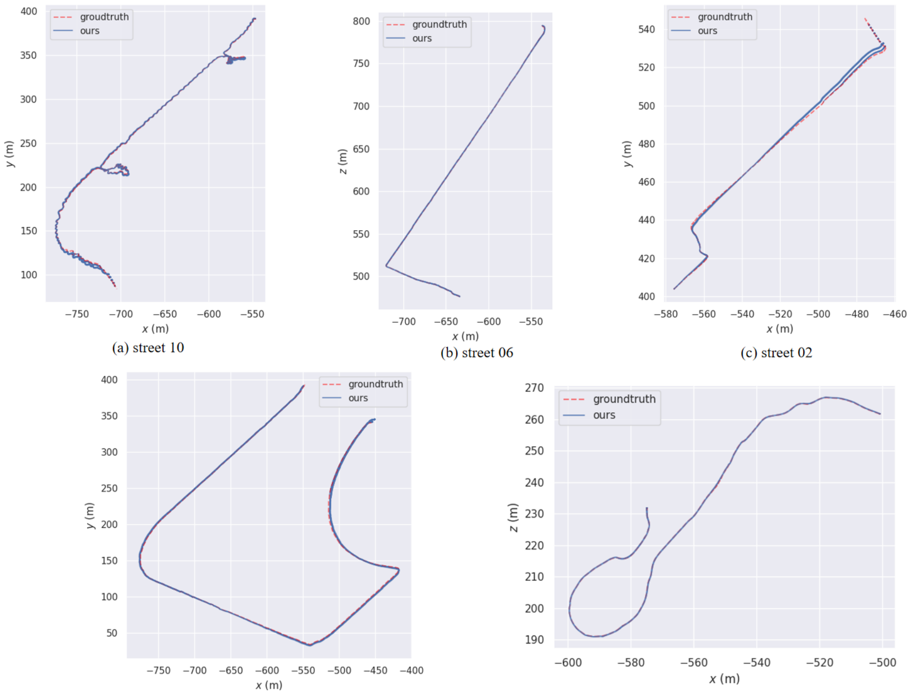
The experimental results also demonstrate the importance of multi-sensor fusion SLAM, especially VINS-Mono, which has the highest error in all scenes except Street 06. This is mainly due to the degradation of the single sensor in long straight lines and light-shadow scenes.

Overall, the experimental results show that our proposed method can effectively handle the challenges of more realistic and complex realistic environments represented by the M2DGR dataset and achieve competitive performance.

4.3. Visualization

The visualization results play a pivotal role in gauging the efficacy and robustness of our proposed PSMS-SLAM method. Figure 7 provides an in-depth illustration of the trajectory obtained through our system's operation on the M2DGR dataset, offering a direct visual comparison against the ground truth. This trajectory visualization is a key metric for evaluating the accuracy and dependability of our SLAM system, providing a tangible representation of how well it reconstructs the environment and estimates the camera pose over time.

In Figure 7, the trajectory traces the path followed by the camera throughout the dynamic scenes, offering a side-by-side comparison with the ground truth. Discrepancies between the estimated trajectory and the ground truth trajectory can be visually identified, providing insights into the system's performance under varying conditions and highlighting any potential challenges faced during the mapping and localization processes.



(a) street 10 (b) street 06 (c) street 02

(d) walk 01 (e) gate 03

Figure 7. The trajectory generated by our PSMS-SLAM method on the M2DGR dataset, providing a side-by-side comparison with the ground truth trajectory.

Moving forward, Figure 8 delves into the visualization results with a specific emphasis on our dynamic object pose estimation method. This figure serves as a comprehensive showcase of the system's capabilities by presenting both velocity and depth information associated with the dynamic objects detected by our PSMS-SLAM system.

The inclusion of velocity information allows for the observation of the speed and directional movement of each identified dynamic object. This aspect is crucial for applications requiring precise understanding and prediction of object movements, such as in robotics and autonomous navigation. By visually inspecting the velocity information in Figure 8, we gain valuable insights into how well our system can handle the dynamic nature of objects within the environment.

Moreover, the visualization of depth information in Figure 8 provides a spatial context, depicting the relative distances and positions of the dynamic objects within the scene. This depth information is fundamental for creating accurate 3D maps and understanding the layout of the environment. Our method leverages multi-sensor fusion to enhance the accuracy of depth estimation, ensuring a more faithful representation of the scene's geometry.

By offering a visual representation of the 6D pose estimation in real time, Figure 8 allows for a qualitative assessment of our system's performance in handling the challenges presented by dynamic entities in the environment. The dynamic object pose estimation module proves to be robust, capturing not only the positional information but also the dynamic behaviors of objects, which is crucial for applications in SLAM.

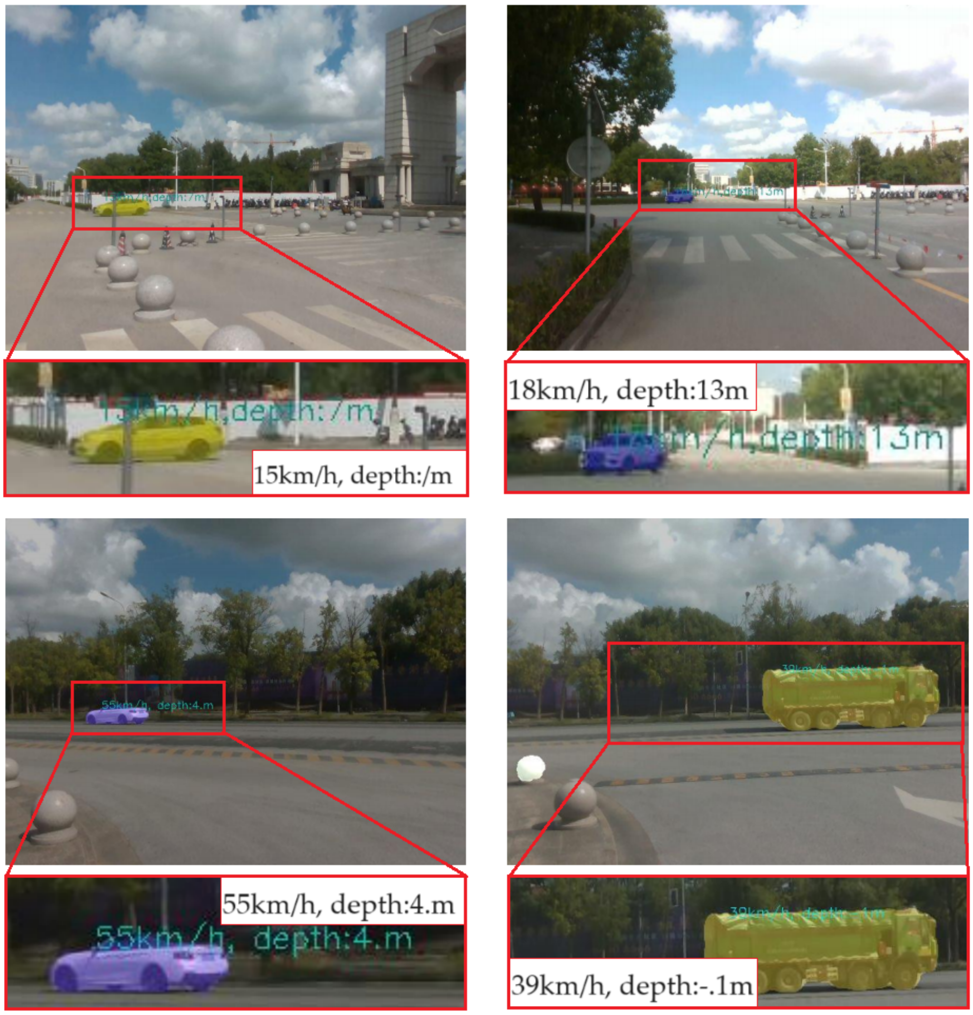


Figure 8. Visualization results of the dynamic target tracking. The visualization results of dynamic target tracking encapsulate the outcomes of our PSMS-SLAM method's performance in tracking dynamic objects within the environment.

5. Conclusions

This paper introduced PSMD-SLAM, a pioneering SLAM method that combines panoptic segmentation and multi-sensor fusion for dynamic environments. Leveraging the LVI-SAM framework, our approach significantly boosts system performance. Key contributions include the introduction of PSMD-SLAM, dynamic object detection and removal through panoptic segmentation, and a robust 6D pose estimation module. The PSMS-SLAM framework integrates IMU, LiDAR–inertial, panoptic segmentation, visual–inertial, and back-end modules, enhancing each other's performance. The visual–inertial system, treating frames with panoptic segmentation as keyframes, reinforces dynamic object pose estimation and the SLAM process. Overall, PSMD-SLAM offers a promising solution for robust SLAM in dynamic environments by effectively incorporating panoptic segmentation and multi-sensor fusion. During the research process of PSMD-SLAM, the key challenge encountered is the accuracy and real-time issues of scene semantic recognition. In response to this challenge, future research will focus on improving the accuracy and real-time performance of panoptic segmentation. This research direction will contribute to enhancing the robustness and practicality of SLAM systems in dynamic environments.

Author Contributions: Conceptualization, J.C. and C.S.; Methodology, C.S. and B.Z.; Software, C.S. and B.Z.; Validation, F.W.; Investigation, F.H.; Data curation, F.W. and F.H.; Writing—original draft, C.S. and B.Z.; Writing—review & editing, J.C. and F.H.; Visualization, B.Z.; Project administration, C.S.; Funding acquisition, C.S. and J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by Guangdong Major Project of Basic and Applied Basic Research (2023B0303000016), Shenzhen Technology Project (JCYJ20220818101211025) and CAS Key Technology Talent Program.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available on request. The data presented in this study are available on request from the corresponding author, C.S or B.Z. We plan to release our code at the appropriate time.

Conflicts of Interest: Author Bo Zeng was employed by the company China Mobile Financial Technology Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Thrun, S.; Leonard, J.J. Simultaneous Localization and Mapping. In *Springer Handbook of Robotics*; Siciliano, B., Khatib, O., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 871–889.
2. Chatterjee, A.; Rakshit, A.; Singh, N.N. Simultaneous Localization and Mapping (SLAM) in Mobile Robots. In *Vision Based Autonomous Robot Navigation. Studies in Computational Intelligence*; Springer: Berlin/Heidelberg, Germany, 2013; Volume 455, pp. 167–206.
3. Ouyang, M.; Shi, X.; Wang, Y.; Tian, Y.; Shen, Y.; Wang, D.; Wang, P.; Cao, Z. A Collaborative Visual SLAM Framework for Service Robots. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 8679–8685.
4. Zheng, S.; Wang, J.; Rizos, C.; Ding, W.; El-Mowafy, A. Simultaneous Localization and Mapping (SLAM) for Autonomous Driving: Concept and Analysis. *Remote Sens.* **2023**, *15*, 1156. [CrossRef]
5. Piao, J.; Kim, S. Adaptive Monocular Visual–Inertial SLAM for Real-Time Augmented Reality Applications in Mobile Devices. *Sensors* **2017**, *17*, 2567. [CrossRef] [PubMed]
6. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [CrossRef]
7. Zhang, J.; Singh, S. LOAM: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*; Berkeley, CA, USA, 2014; Volume 2, pp. 1–9. Available online: https://www.ri.cmu.edu/pub_files/2014/7/Ji_LidarMapping_RSS2014_v8.pdf (accessed on 20 April 2024).
8. Kudo, T.; Miura, J. Utilizing WiFi signals for improving SLAM and person localization. In Proceedings of the 2017 IEEE/SICE International Symposium on System Integration (SII), Taipei, Taiwan, 11–14 December 2017; pp. 487–493.
9. Bescos, B.; Fácil, J.M.; Civera, J.; Neira, J. DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robot. Autom. Lett.* **2018**, *3*, 4076–4083. [CrossRef]
10. Bescos, B.; Campos, C.; Tardós, J.D.; Neira, J. DynaSLAM II: Tightly-coupled multi-object tracking and SLAM. *IEEE Robot. Autom. Lett.* **2021**, *6*, 5191–5198. [CrossRef]
11. Yu, C.; Liu, Z.; Liu, X.-J.; Xie, F.; Yang, Y.; Wei, Q.; Fei, Q. DS-SLAM: A semantic visual SLAM towards dynamic environments. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1168–1174.
12. Graeter, J.; Wilczynski, A.; Lauer, M. Limo: Lidar-monocular visual odometry. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 7872–7879.
13. López, E.; García, S.; Barea, R.; Bergasa, L.M.; Molinos, E.J.; Arroyo, R.; Romera, E.; Pardo, S. A multi-sensorial simultaneous localization and mapping (SLAM) system for low-cost micro aerial vehicles in GPS-denied environments. *Sensors* **2017**, *17*, 802. [CrossRef] [PubMed]
14. Liang, Y.; Müller, S.; Schwendner, D.; Rolle, D.; Ganesch, D.; Schaffer, I. A scalable framework for robust vehicle state estimation with a fusion of a low-cost IMU, the GNSS, radar, a camera and lidar. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 1661–1668.
15. Zuo, X.; Geneva, P.; Lee, W.; Liu, Y.; Huang, G. Lic-fusion: Lidar-inertial-camera odometry. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 5848–5854.

16. Zuo, X.; Yang, Y.; Geneva, P.; Lv, J.; Liu, Y.; Huang, G.; Pollefeys, M. Lic-fusion 2.0: Lidar-inertial-camera odometry with sliding-window plane-feature tracking. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 5112–5119.
17. Liu, X.; Wen, S.H.; Jiang, Z.M.; Tian, W.B.; Qiu, T.Z.; Othman, K.M. A multisensor fusion with automatic vision–LiDAR calibration based on factor graph joint optimization for SLAM. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 9513809. [CrossRef]
18. Huang, Q.; Pu, C.; Fourie, D.; Khosoussi, K.; How, J.P.; Leonard, J.J. NF-iSAM: Incremental smoothing and mapping via normalizing flows. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 1095–1102.
19. Shan, T.; Englot, B.; Ratti, C.; Rus, D. Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 5692–5698.
20. Wisth, D.; Camurri, M.; Das, S.; Fallon, M. Unified multi-modal landmark tracking for tightly coupled lidar-visual-inertial odometry. *IEEE Robot. Autom. Lett.* **2021**, *6*, 1004–1011. [CrossRef]
21. Lin, J.; Zhang, F. R 3 LIVE: A robust, real-time, RGB-colored, LiDAR-inertial-visual tightly-coupled state estimation and mapping package. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; pp. 10672–10678.
22. Song, C.; Niu, M.; Liu, Z.; Cheng, J.; Wang, P.; Li, H.; Hao, L. Spatial-temporal 3D dependency matching with self-supervised deep learning for monocular visual sensing. *Neurocomputing* **2022**, *481*, 11–21. [CrossRef]
23. Yin, J.; Li, A.; Li, T.; Yu, W.; Zou, D. M2dgr: A multi-sensor and multi-scenario slam dataset for ground robots. *IEEE Robot. Autom. Lett.* **2021**, *7*, 2266–2273. [CrossRef]
24. Luo, C.; Yang, Z.; Wang, P.; Wang, Y.; Xu, W.; Nevatia, R.; Yuille, A. Every Pixel Counts ++: Joint Learning of Geometry and Motion with 3D Holistic Understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2624–2641. [CrossRef] [PubMed]
25. Kirillov, A.; He, K.; Girshick, R.; Rother, C.; Dollar, P. Panoptic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
26. Xiong, Y.; Liao, R.; Zhao, H.; Hu, R.; Bai, M.; Yumer, E.; Urtasun, R. Upsnet: A unified panoptic segmentation network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8818–8826.
27. Hoang, D.-C.; Lilienthal, A.J.; Stoyanov, T. Panoptic 3d mapping and object pose estimation using adaptively weighted semantic information. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1962–1969. [CrossRef]
28. Hou, R.; Li, J.; Bhargava, A.; Raventos, A.; Guizilini, V.; Fang, C.; Lynch, J.; Gaidon, A. Real-time panoptic segmentation from dense detections. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
29. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
30. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
31. Runz, M.; Buffier, M.; Agapito, L. MaskFusion: Real-Time Recognition, Tracking and Reconstruction of Multiple Moving Objects. In Proceedings of the 2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Munich, Germany, 16–20 October 2018; pp. 10–20.
32. Liao, Z.; Hu, Y.; Zhang, J.; Qi, X.; Zhang, X.; Wang, W. SO-SLAM: Semantic object SLAM with scale proportional and symmetrical texture constraints. *IEEE Robot. Autom. Lett.* **2022**, *7*, 4008–4015. [CrossRef]
33. Song, C.; Zeng, B.; Su, T.; Zhang, K.; Cheng, J. Data association and loop closure in semantic dynamic SLAM using the table retrieval method. *Appl. Intell.* **2022**, *52*, 11472–11488. [CrossRef]
34. Yang, S.; Scherer, S. CubeSLAM: Monocular 3-D Object SLAM. *IEEE Trans. Robot.* **2019**, *35*, 925–938. [CrossRef]
35. Forster, C.; Carlone, L.; Dellaert, F.; Scaramuzza, D. IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. In *Robotics: Science and Systems XI*; Mit Press: Cambridge, MA, USA, 2015.
36. Shan, T.; Englot, B. Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4758–4765.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.