Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 4075910, 18 pages
https://doi.org/10.1155/2022/4075910

Research Article

A Variable Radius Side Window Direct SLAM Method Based on Semantic Information

Yan Chen[1], **Jianjun Ni**[1,2], **Emmanuel Mutabazi**[1], **Weidong Cao**[1,2] **and Simon X. Yang**[3]

[1]*College of Internet of Things Engineering, Hohai University, Changzhou 213022, China*
[2]*Jiangsu Key Laboratory of Power Transmission & Distribution Equipment Technology, Hohai University, Changzhou 213022, China*
[3]*Advanced Robotics and Intelligent Systems (ARIS) Laboratory, School of Engineering, University of Guelph, Guelph, ON, Canada*

Correspondence should be addressed to Jianjun Ni; njjhhuc@gmail.com

Received 5 May 2022; Accepted 28 June 2022; Published 22 August 2022

Academic Editor: Nian Zhang

Simultaneous Localization and Mapping (SLAM) is a challenging and key issue in the mobile robotic fields. In terms of the visual SLAM problem, the direct methods are more suitable for more expansive scenes with many repetitive features or less texture in contrast with the feature-based methods. However, the robustness of the direct methods is weaker than that of the feature-based methods. To deal with this problem, an improved direct sparse odometry with loop closure (LDSO) is proposed, where the performance of the SLAM system under the influence of different imaging disturbances of the camera is focused on. In the proposed method, a method based on the side window strategy is proposed for preprocessing the input images with a multilayer stacked pixel blender. Then, a variable radius side window strategy based on semantic information is proposed to reduce the weight of selected points on semistatic objects, which can reduce the computation and improve the accuracy of the SLAM system based on the direct method. Various experiments are conducted on the KITTI dataset and TUM RGB-D dataset to test the performance of the proposed method under different camera imaging disturbances. The quantitative and qualitative evaluations show that the proposed method has better robustness than the state-of-the-art direct methods in the literature. Finally, a real-world experiment is conducted, and the results prove the effectiveness of the proposed method.

1. Introduction

Simultaneous Localization and Mapping (SLAM) plays essential roles in robotic and other related fields [1–3]. In the robotic field, SLAM systems are used to solve the problem of robots about where they are. Based on the acquisition of its pose and surrounding environment, a robot can further solve where to go or what to do [4].

Many kinds of sensors are used in SLAM systems, such as LiDAR, camera, and inertial measurement unit [5, 6]. Commonly, SLAM algorithms are divided into laser SLAM and visual SLAM according to the sensor used [7, 8]. Due to the low cost of the camera, the large amount of information it carries, and the ease of use, visual SLAM has become more popular among researchers in recent years. Visual SLAM usually uses monocular cameras, binocular cameras, or RGB-D cameras to obtain environmental information. Compared with other types of cameras, the monocular camera is cheap and common. In addition, there are the most abundant data sources of the monocular camera. So the monocular SLAM plays an important role in the visual SLAM field and has been widely studied and applied [9, 10]. However, the monocular SLAM can obtain only image information without scale information, so it is more dependent on the quality of the image. Therefore, how to improve the robustness of monocular SLAM under different disturbances is a very challenging and important task in this field [11, 12].

There are three main implementation schemes in visual SLAM, namely feature-based method, direct method, and

semidirect method. The feature-based method finds feature points, matches them, calculates the pose, and constructs a map through geometric relations. The most commonly used methods for feature extraction are Scale Invariant Feature Transform (SIFT) [13], Speeded Up Robust Features (SURF) [14], and Oriented Fast and Rotated BRIEF (ORB) [15]. ORB is one of the best methods, which improves the speed and accuracy of FAST [16], and uses BRIEF [17] for the efficient computation of features. Accordingly, ORB-SLAM is currently the most popular visual SLAM solution [18, 19].

Unlike the feature-based method, the direct approach does not rely on the one-to-one matching of points. It optimizes the interframe pose by extracting pixels with apparent gradients and minimizing the photometric error function of the pixels, such as the large-scale direct monocular SLAM (LSD-SLAM) [20] and the direct sparse odometry (DSO) [21]. The semidirect method, such as the semidirect visual odometry (SVO) [22], uses a similar structure to the feature-based method and combines the tracking of the direct method and the motion optimization of the feature-based method. The feature-based method and the semidirect method both rely on low-level geometric feature extractors with high repeatability. They are not suitable for surfaces with many repetitive features or less texture. In contrast, the direct method can be used in a broader range of scenarios. In this paper, we focus on direct method solutions for the monocular SLAM. The main purpose of this paper is to improve the robustness of the direct methods under different disturbances.

The robustness of the direct method-based SLAM system is challenged by photometric calibration, dynamic objects, rolling shutter effect, camera imaging disturbances, and so on [23]. There have been many excellent works to improve the robustness of the direct method-based SLAM systems. For example, Zhu et al. [24] proposed a photometric transfer net (PTNet), which is trained to pixel-wisely remove brightness discrepancies between two frames without ruining the context information, to overcome the problem of brightness discrepancies. Liu et al. [25] proposed an enhanced visual SLAM algorithm based on the sparse direct method to solve the illumination sensitivity problem. Sheng et al. [26] filtered out the dynamic objects based on the semantic information to improve the positioning accuracy and robustness of DSO [21]. Zhou et al. [27] jointly optimized the 3D lines, points, and poses within a sliding window to consider the collinear constraint among the points to improve the robustness of the direct method.

The works introduced above can improve the robustness of the direct method to some extent. However, the research focusing on the influence of different camera imaging disturbances and semistatic objects is relatively lacking. During the long-term operation of the monocular SLAM system, the image quality of the camera will be affected by different disturbances from the external environment and internal sensors. In this paper, two main types of imaging disturbances are studied, namely, different noise on the camera and the brightness influence on the imaging process. The main noises on the camera include Gaussian noise and Salt-and-Pepper noise. Gaussian noise is often caused by the high temperature of the camera sensor running for a long time and mutual interference of internal circuit components [28]. Salt-and-Pepper noise is often caused by the faulty of the camera sensor, the wear of the camera lens, and the adsorption of dust in the air [29, 30]. The brightness influence on the imaging is a very common problem of the vision-based SLAM. For example, the accumulated irradiance exceeding the camera's dynamic range can cause the camera overexposure interference when the ambient brightness is not uniform [31, 32]. Another important influence on the robustness of the direct methods in the vision-based SLAM is the semistatic objects, which refer to objects that are static most of the time but will change at a certain moment, such as the cars parked on the side of the road. Semistatic objects are not suitable for being directly filtered out like dynamic objects because most of them are rich in texture and are suitable for estimating pose when they are static [33]. Thus, the main motivation of this paper is to study how to improve the robustness of the direct method-based SLAM system in different camera imaging disturbances and reduce the specific gravity of semistatic objects.

The main contributions of this paper are as follows: (1) A regional pixel information fusion method based on multiple average calculations is proposed to improve the robustness of the direct sparse odometry with loop closure- (LDSO-) based SLAM. (2) A side window strategy is introduced into the framework of the LDSO-based SLAM to enhance the edge-preserving property. (3) A method based on semantic information is presented to reduce the effects of nonstatic objects on the LDSO-based SLAM. So there are three main improvements of the proposed method, namely, a regional pixel information fusion method for robustness, a side window strategy for edge preserving, and the semantic-based strategy for the nonstatic objects. Compared with the existing methods, the proposed method improves the robustness of the direct method-based SLAM against multiple camera imaging disturbances, including Gaussian noise, Salt-and-Pepper noise, and camera overexposure, rather than just against a single disturbance.

The rest of this paper is organized as follows. Section 2 gives out an overview of the background. The proposed algorithm is presented in Section 3. In Section 4, detailed quantitative and qualitative experimental results are provided. The discussions of the proposed algorithm are carried out in Section 5. Finally, Section 6 concludes this paper and gives out the future work.

2. Background

Direct method-based SLAM systems jointly estimate the position and posture changes of the camera by minimizing the photometric error in the image alignment. It makes direct methods more accurate and robust than feature-based methods in scenes that lack texture or are full of repetitive textures. However, the monocular direct methods suffer from the accumulated drift of global translation, rotation, and scale without closed-loop detection. This leads to inaccurate long-term trajectory estimation and mapping. In this paper, Direct Sparse Odometry with Loop closure

(LDSO) [34] is focused on, which adds closed-loop detection to DSO for global optimization. The main process of LDSO is reviewed in this section.

2.1. Framework of LDSO. The algorithm framework of LDSO is shown in Figure 1. When a new frame of image is acquired, all the active 3D points in the current sliding window of the local bundle adjustment module are projected into this frame. The initial pose of this frame is estimated by direct image alignment. This frame is added to the local windowed bundle adjustment if it is judged as a new keyframe. Old or redundant keyframes and points are marginalized. The active keyframes and the marginalized keyframes rely on bag-of-words (BoW) for closed-loop detection and verification. If the closed-loop candidate is verified, it is added to the global pose graph for optimization.

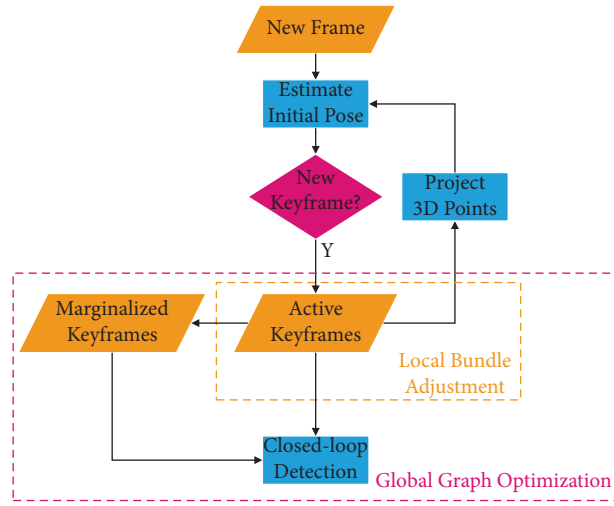


Figure 1: The framework of the LDSO method.

2.2. Local Bundle Adjustment. In the local bundle adjustment module based on sliding window, 5–7 keyframes are maintained. Their parameters are jointly optimized by minimizing the photometric error. The photometric error is defined as

$$\min \sum_{\mathbf{T}_i,\mathbf{T}_j,\mathbf{p}_k \in W} E_{i,j,k}, \tag{1}$$

where $W = \{\mathbf{T}_1, \ldots, \mathbf{T}_m, \mathbf{p}_1, \ldots, \mathbf{p}_n\}$ is the m keyframe poses represented as Euclidean transformation and n points of inverse depth parameterization in the sliding window. $E_{i,j,k}$ is calculated by

$$E_{i,j,k} = \sum_{\mathbf{p} \in N_{\mathbf{p}_k}} w_{\mathbf{p}} \left\| \left(I_j[\mathbf{p}'] - b_j\right) - \frac{t_j e^{a_j}}{t_i e^{a_i}} \left(I_i[\mathbf{p}] - b_i\right) \right\|_{\gamma}, \tag{2}$$

where $N_{\mathbf{p}_k}$ denotes the neighborhood pattern of $\mathbf{p}_k$; a and b are the affine light transform parameters; t denotes the exposure time; I is an image; $w_{\mathbf{p}}$ is a heuristic weighting factor; $\|\cdot\|_{\gamma}$ is the Huber norm; and $\mathbf{p}'$ denotes the reprojected pixel of $\mathbf{p}$ on I_j, which is calculated by

$$\mathbf{p}' = \prod \left(\mathbf{R}\Pi^{-1}\left(\mathbf{p}, d_{\mathbf{p}_k}\right) + \mathbf{t}\right), \tag{3}$$

where Π is the projection function from $\mathbb{R}^3$ to Ω; $\mathbf{R}$ and $\mathbf{t}$ are the relative rotation and translation between the two frames; and $d_{\mathbf{p}}$ is the inverse depth of point $\mathbf{p}$.

2.3. Closed-Loop Detection and Verification. In the LDSO SLAM, the DSO's point selection strategy has been modified to be more sensitive to corner points. The selected corner points are calculated as their ORB descriptors and packed into BoW. When the ORB descriptor of each keyframe is calculated, the closed-loop candidates of the keyframe are proposed by querying the BoW database. The similarity transformation from the closed-loop candidate to the current keyframe $\mathbf{S}_{cr}$ is optimized by minimizing 3D and 2D geometric constraints:

$$E_{loop} = \sum_{\mathbf{q}_i \in Q_1} w_1 \left\| \mathbf{S}_{cr}\Pi^{-1}\left(\mathbf{p}_i, d_{\mathbf{p}_i}\right) - \Pi^{-1}\left(\mathbf{q}_i, d_{\mathbf{q}_i}\right) \right\|_2 + \sum_{\mathbf{q}_j \in Q_2} w_2 \left\| \prod \left(\mathbf{S}_{cr}\Pi^{-1}\left(\mathbf{p}_j, d_{\mathbf{p}_j}\right)\right) - \mathbf{q}_j \right\|_2, \tag{4}$$

where Q_1 and Q_2 are the matched features in the current keyframe without and with depth, respectively; $\mathbf{p}_i$ denotes the reconstructed feature in the closed-loop candidates; $d_{\mathbf{q}}$ is the inverse depth of the feature $\mathbf{q}$; and w_1 and w_2 are the weights to balance the different measurement units.

It can be noticed from equation (2) that the pose estimation of LDSO relies on minimizing the photometric error of the selected points. If the selected points are disturbed by imaging disturbances, equation (2) is converted into

$$E_{i,j,k} = \sum_{\mathbf{p} \in N_{\mathbf{p}_k}} w_{\mathbf{p}} \left\| \left(I_j[\mathbf{p}'] - b_j\right) - \frac{t_j e^{a_j}}{t_i e^{a_i}} \left(I_i[\mathbf{p}] - b_i\right) \right\|_{\gamma} + E_n, \tag{5}$$

where E_n is the error due to imaging disturbances. As the intensity of the camera imaging disturbance increases, the optimization direction for minimizing the photometric error is more inclined to the error caused by the imaging disturbances rather than the estimated pose. Therefore, the robustness of LDSO in camera imaging disturbances is not strong enough.

3. Proposed Method

To enhance the robustness of the direct SLAM method, the points are fused with the surrounding pixels' information. The overview of the proposed method for obtaining and using fusion points is shown in Figure 2.

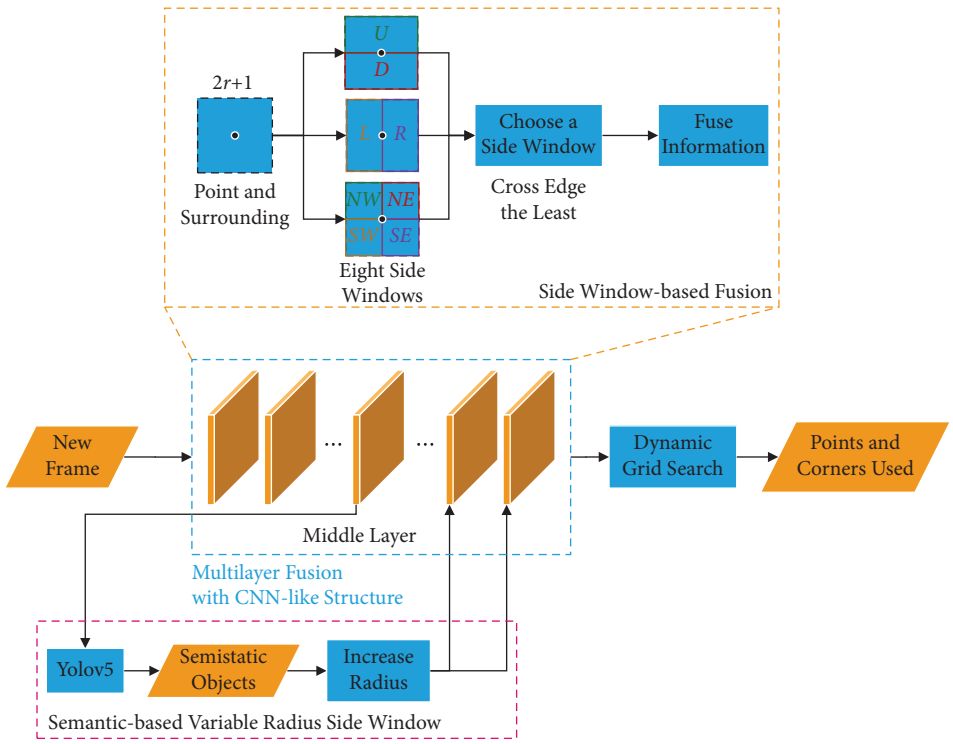


Figure 2: The overview of the proposed method for obtaining and using fusion points.

As shown in Figure 2, the area around each pixel is divided into blocks according to the side window strategy when a new frame arrives. The area block that crosses the image's edge the fewest times is chosen. This region block's pixel information is averaged into a single point. Multilayers of such pixel information fusion are superimposed to form a convolutional neural network (CNN) like structure [35, 36]. In the middle layer, semistatic objects are detected. The radiuses of the side window of the pixels belonging to the semistatic objects are increased in the back layers. The fusion points form the fused image. The points with sufficient gradient intensity and corners are selected using a dynamic grid search. These points are used in direct SLAM to improve the robustness of the system. The details of the proposed method are introduced as follows. The regional pixel information fusion is realized by multilayer fusion with a CNN-like structure. Then, the side window strategy is added to the fusion method for edge preservation. Finally, the radius of the side window is adjusted based on semantic information to reduce the weights of the semistatic objects.

3.1. Regional Pixel Information Fusion Method. As we know, the main reason why the robustness of feature-based SLAM is better than that of direct SLAM is that the feature carries the general information of pixels in a local area instead of a single pixel [37]. Therefore, to improve the robustness of LDSO in different camera imaging disturbances, a regional pixel information fusion method is introduced into the LDSO algorithm. Namely, each pixel can fuse the information of surrounding pixels, and the fusion intensity decreases as the distance between the pixels increases.

The mean filter is one of the most common methods of fusing pixels. Unlike other filters such as the median, max, and min filters, which select one pixel and discard others, the mean filter considers information from all pixels. In addition, the mean filter is simple to implement. So, a 3×3 mean filter is used to fuse eight neighborhood pixels into one pixel in this study. At the same time, referring to the characteristics of the classic convolutional neural networks (CNN) [38], the mean filters are stacked in the structure of CNN. In CNN, the stacked convolutional layers are considered to extract high-level features of the image so that these feature points can be used for object classification operations. Each feature point obtained contains information about a local area. The CNN-like structure of the multilayer fusion used in this study is shown in Figure 2.

Remark 1. The main reason for using the 3×3 mean filter in this paper is that it is the minimum size that can cover eight neighborhood information. Using the stacking structure, the 3×3 receptive field can be easily expanded to 5×5, 7×7, and other larger receptive fields. By this stacking structure, the closer the points in this area are to the edge, the fewer times they are repeatedly used and the less they affect the obtained feature points. These characteristics are precisely in line with our needs for fusing regional pixel information.

3.2. Side Window Strategy for Pixel Fusion Area Selection. The consistent use of the square area as the pixel fusion range can conveniently improve the overall robustness of the visual odometry, but it will also cause a certain degree of damage to the edges of the image. The more the layers are stacked, the

greater the degree of damage. In image processing, this is called nonedge preservation [39]. As mentioned earlier, the points/features selected by LDSO are pixels with sufficient intensity gradients and corner features. Pixel fusion across the edges will reduce the gradient intensity of the pixels and blur the corner features. Since it makes the selected points difficult to gather at the edge, the point cloud map constructed is very unclear. Since the corner features are blurred and difficult to be extracted, it is difficult for LDSO to detect the closed-loop effectively.

To solve the above problems caused by the nonedge preservation of pixel fusion in the fixed square area, the side window strategy is introduced into LDSO [40]. The side window strategy treats each pixel as a potential edge point. Unlike the traditional pixel fusion method that takes the pixel's position as the center of the filter window, the side window strategy aligns the edge of the filter window with the pixel. Different from nonlinear anisotropic weightings such as the spatial weighting and gray value weighting of bilateral filters, which only reduce the diffusion of pixels along the edge normal direction, the side window strategy can cut off all the normal diffusion [41].

The details of the side window strategy proposed in our multilayer fusion are as follows:

(1) Each pixel and its surroundings are divided into eight side windows, as shown in Figure 2. They are the side windows in eight directions: up (U), down (D), left (L), right (R), northwest (NW), northeast (NE), southwest (SW), and southeast (SE). The center point p_i of the pixel fusion is located on the side or corner of the window. The radius r of the side window determines the range of the pixel fusion.

(2) The average value of the pixels in each side window is calculated as the output q_n of the side window, where $n \in \{U, D, L, R, \mathrm{NW}, \mathrm{NE}, \mathrm{SW}, \mathrm{SE}\}$.

(3) Compare the distance measured by L_1 norm between the output q_n of the eight side windows and the center point p_i. The fusion output p^{fusion} of the center point p_i and its surrounding pixels is $p^{\mathrm{fusion}} = q_s$, where

$$s = \underset{n\in\{U,D,L,R,NW,NE,SW,\mathrm{SE}\}}{\arg\min} \left\{\left|q_n - p_i\right|\right\}. \tag{6}$$

Remark 2. In the proposed multilayer superimposed pixel fusion strategy, the diffusion of pixels along the normal edge direction will be further amplified. And the side window strategy cuts off the possibility of pixels spreading along the normal direction of the edge, which is more suitable for our multilayer fusion.

The pseudocode of the proposed side window-based multilayer fusion method is summarized in Algorithm 1.

3.3. Semantic-Based Variable Radius Side Window Strategy. When humans use their eyes to estimate their position and remember the environment, they do not take all the objects they see into consideration. Instead, they focus on static objects such as walls and pillars and use semistatic objects that are stationary most of the time, such as cars parked on the side of the road, as a reference. Inspired by this, a semantic-based variable radius side window strategy is proposed to assign weights to static and semistatic objects.

First, in the first half of the stacked structure of pixel fusion, a smaller radius for the side window is used. In multilayer pixel fusion, due to the smaller coverage area, the side window with a smaller radius can make the image retain more details such as edges while reducing the impact of camera imaging disturbances. Subsequent object detection in a camera imaging disturbed environment is carried out on this basis.

Second, Yolov5 (one of the popular object detection deep networks) is used to distinguish static and semistatic objects in the input images. Yolov5 is the latest version of the Yolo object detection algorithm [42, 43]. The main reason for using the Yolov5 network is that Yolov5 can also maintain a higher processing frame rate under lower hardware conditions while achieving the accuracy of the current state-of-the-art technology. In this study, the pretrained Yolov5 model on the Microsoft COCO (Common Objects in Context) dataset is used to extract object location and category semantic information [44]. Common movable categories such as bicycles, cars, motorcycles, buses, and trucks in the COCO dataset are marked as semistatic objects.

Third, in the second half of the stacked structure of the pixel fusion, a slightly larger radius is used for the side windows of the regions where the semistatic objects are detected. A side window with a larger radius is more likely to contain more image edges. The selection principle of the side window is to select the side window whose output is most similar to the center pixel. The larger the edge gradient of the image within the coverage of the side window, the more dissimilar the output is from the center pixel. Therefore, the side window strategy is more inclined to retain the image edges with large gradients. Edges with smaller gradients in the side window will be blurred. With repeated pixel fusion, the obvious image edges in the semistatic object area will be preserved, while the pixel gradients inside will be reduced.

Remark 3. The specific gravity of the point in the semistatic object area selected by the LDSO with a high gradient intensity will decrease. The preserved obvious image edges can provide enough corner features for LDSO. In this way, a static object-based and semistatic object-assisted approach similar to the human positioning strategy is achieved.

A summary of the proposed points selection strategy based on the side window with semantic-based variable radius is given in Algorithm 2.

Overall, the workflow of the proposed variable radius side window direct SLAM method is summarized as follows:

Step 1. The radius parameters applicable to different regions are selected based on semantic information.

Input: Image I, Layer number L, Radius of side window r
Output: Set of fusion points
for $\forall l \in L$ **do**
  for $\forall\{x_i, y_i\} \in I$ **do**
    $S = \{(x_i - r)\colon (x_i + r), (y_i - r)\colon (y_i + r)\}$;
    % S is the surrounding of the pixel p_i
    Divide S into $\{U, D, L, R, \mathrm{NW}, \mathrm{NE}, \mathrm{SW}, \mathrm{SE}\}$;
    for $n \in \{U, D, L, R, \mathrm{NW}, \mathrm{NE}, \mathrm{SW}, \mathrm{SE}\}$ **do**
      $q_n = \mathrm{mean}(p_j), j \in n$;
    end for
    $s = \arg\min\{|q_n - p_i|\}$;
    % Select the side window s;
    $p^{\mathrm{fusion}} = q_s$;
  end for
end for

Algorithm 1: Side window-based multilayer fusion.

Input: Number of layers L, Desired number of points N_{des}
Output: Selected points
for $\forall l \in L$ **do**
  if $l < 1/2L$ **then**
    Use small radius side windows for multilayer fusion;
  end if
  if $l \geq 1/2L$ **then**
    Use Yolov5 to distinguish static and semistatic objects;
    Increase the radius of the side windows of the regions where the semistatic objects are detected;
  end if
end for
Split the image composed of fusion points into patches;
while $N_{\mathrm{sel}} < N_{\mathrm{des}}$ **do**
  Randomly select a patch M
  Compute the median of gradient as the region-adaptive threshold;
  Split M into $d \times d$ blocks;
  Select a point with the highest gradient which surpasses the gradient threshold from $d \times d$, 2 $d \times 2$ d, 4 $d \times 4$ d blocks separately;
end while

Algorithm 2: Semantic variable radius side window-based points selection.

Step 2. The different radius parameters are applied to the side window strategy to form a variable radius side window strategy.

Step 3. The semantic information-based variable radius side window strategy is applied to a multilayer stacked pixel blender to fuse regional pixel information.

Step 4. The points are selected according to Algorithm 2 on the points fused with local information.

Step 5. The selected points are used to estimate the camera pose by minimizing equation (2) and perform global optimization by minimizing equation (4) when loop closures are detected.

4. Experimental Results and Analysis

In this section, the proposed method is comprehensively evaluated on outdoor datasets (KITTI dataset) and indoor datasets (TUM RGB-D dataset), which are introduced as follows:

(1) KITTI dataset [45, 46]: this dataset is currently the most extensive dataset in the world for evaluating computer vision algorithms in autonomous driving scenarios. It contains real image data collected in outdoor scenes such as urban areas, villages, and highways. The "00–10" sequences in this dataset provide ground truth, which are used in this study.

(2) TUM RGB-D dataset [47, 48]: this dataset provides RGB-D data and ground-truth data intending to establish a novel benchmark for the evaluation of

(a) (b) (c)

Figure 3: Comparison of the example scene before and after adding noise. (a) The original image. (b) The image after adding Gaussian noise. (c) The image after adding Salt-and-Pepper noise. Note that the effect of the added noise is noticeable.

visual odometry and visual SLAM systems. In this paper, the sequences "freiburg1_xyz," "freiburg2_xyz," "freiburg2_rpy," "freiburg1_desk," and "freiburg1_desk2" are selected, which were all acquired in the office interior scene with rich texture.

The main reason for using the two datasets is that both of them provide ground truth, which is required for the quantitative evaluation. Because there is a certain natural camera overexposure problem in the two datasets [49], they are used directly to test the proposed method under the disturbance of camera overexposure. In addition, Gaussian noise and Salt-and-Pepper noise are added to the two datasets in these experiments to further test the proposed method under different camera sensor noises. In this paper, the variance of Gaussian noise added is 0.003, and the rate of Salt-and-Pepper noise added is 10%. The noise addition operation and the noise-adding parameters in this study are relatively common in the literature [50, 51]. Figure 3 shows an example scene before and after adding two kinds of noise.

4.1. Quantitative Evaluation. In this study, the proposed method is based on the side window fusion strategy on the direct method-based SLAM. Here, it is compared with the general direct sparse odometry method (DSO) and the general direct sparse odometry with loop closure (LDSO). In this paper, the large-scale direct monocular SLAM (LSD-SLAM) is not compared because its tracking robustness is not as good as DSO [52]. To further discuss the performance of our method, ORB-SLAM3 is also added for comparison, which is one of the state-of-the-art methods based on the feature-based method [53, 54]. The root mean squared error of absolute trajectory error ($RMSE_{ATE}$) is used to evaluate the performance of these methods [55].

4.1.1. On the KITTI Dataset. Firstly, some comparison experiments are conducted on the KITTI dataset to show the robustness of the proposed method in the face of different camera imaging disturbances. The results with no noise added, Gaussian noise, and Salt-and-Pepper noise are listed in Tables 1–3, respectively. The missing values in the tables mean tracking failures.

The results in Table 1 show that our method can achieve similar or better performance compared with the other direct methods in the sequences without added noise. The results on the sequences without added noise show that the performance of the proposed method is obviously better than the general LDSO method on the sequences "KITTI_00" and "KITTI_02," where the RMSE values of the proposed method are 32.42% and 51.91% less than the general LDSO method. The main reason is that the sequences "KITTI_00" and "KITTI_02" have a large number of scenes in the shade of trees (see Figures 4(a) and 4(c)), and frequent changes in ambient light bring more frequent camera overexposure problems to the images. The results show that the proposed method can deal with the camera overexposure interference on the direct methods effectively.

In the sequences with Gaussian noise, we can see that the performance of the general direct methods decreases obviously on all of the sequences in the KITTI dataset, but the proposed method is not seriously affected by the Gaussian noise (see Table 2). In particular, the other direct methods fail to track in sequence "KITTI_03," "KITTI_04," and "KITTI_09" while our method still works. The results in Table 2 show that the proposed method outperforms the general LDSO method by more than 13.7% on all of the sequences in the KITTI dataset. In the sequences with Salt-and-Pepper noise, DSO and LDSO are entirely inoperable, while our method obtains good performance (see Table 3).

Compared with ORB-SLAM3, our method obtains slightly better performance on the sequences without added noise, except sequences "KITTI_08," "KITTI_09," and "KITTI_10." The main reason is that these sequences contain very rich textures that are more suitable for feature-based methods. In particular, ORB-SLAM3 will track failure in the sequence "KITTI_01," whether the noise is added or not. This is due to the fact that the sequence "KITTI_01" is a very texture-deficient highway scene and is not suitable for feature-based SLAM methods (see Figure 4(b)).

Table 1: $RMSE_{ATE}$ on KITTI dataset with no noise added.

Method	No noise added											
	KITTI_00	KITTI_01	KITTI_02	KITTI_03	KITTI_04	KITTI_05	KITTI_06	KITTI_07	KITTI_08	KITTI_09	KITTI_10	Average
DSO [21]	115.035	31.811	152.463	2.030	0.755	49.981	54.004	17.576	114.391	70.534	14.661	56.658
LDSO [34]	7.360	9.972	47.245	2.342	0.800	4.166	12.805	1.691	114.739	69.803	14.815	25.976
ORB-SLAM3 [54]	9.265	—	22.025	2.117	1.223	4.034	16.196	1.688	38.114	7.243	7.771	10.968
Ours	4.974	9.710	22.722	2.183	0.857	3.540	12.798	1.789	99.579	52.469	14.210	20.439

Note. "—" means tracking failure. The average value is calculated based on the number of successes.

Table 2: $RMSE_{ATE}$ on KITTI dataset with Gaussian noise.

Method	Gaussian noise											
	KITTI_00	KITTI_01	KITTI_02	KITTI_03	KITTI_04	KITTI_05	KITTI_06	KITTI_07	KITTI_08	KITTI_09	KITTI_10	Average
DSO [21]	115.771	56.143	185.187	—	—	50.185	59.382	38.812	127.674	—	15.287	81.055
LDSO [34]	22.543	23.052	169.247	—	—	44.010	58.729	53.481	130.993	—	16.277	64.792
ORB-SLAM3 [54]	10.645	—	59.868	2.860	1.911	9.250	19.249	1.932	42.931	8.223	8.776	16.565
Ours	17.772	13.023	120.380	2.133	1.093	5.740	13.491	1.973	102.206	52.664	14.042	31.320

Note. "—" means tracking failure. The average value is calculated based on the number of successes.

Table 3: $RMSE_{ATE}$ on KITTI dataset with Salt-and-Pepper noise.

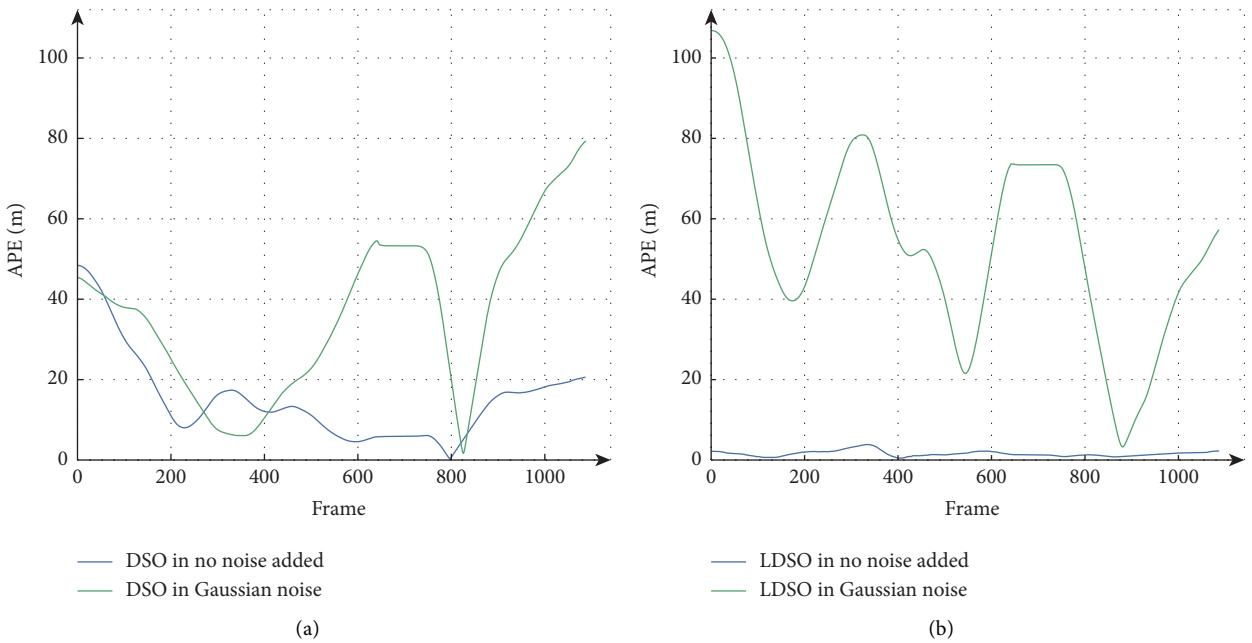
Method	Salt-and-Pepper noise											
	KITTI_00	KITTI_01	KITTI_02	KITTI_03	KITTI_04	KITTI_05	KITTI_06	KITTI_07	KITTI_08	KITTI_09	KITTI_10	Average
DSO [21]	—	—	—	—	—	—	—	—	—	—	—	—
LDSO [34]	—	—	—	—	—	—	—	—	—	—	—	—
ORB-SLAM3 [54]	—	—	—	—	—	—	—	—	—	—	—	—
Ours	19.798	10.464	108.448	2.252	0.806	11.581	12.463	2.238	101.590	52.177	14.937	30.614

Note. "—" means tracking failure.

(a) (b)

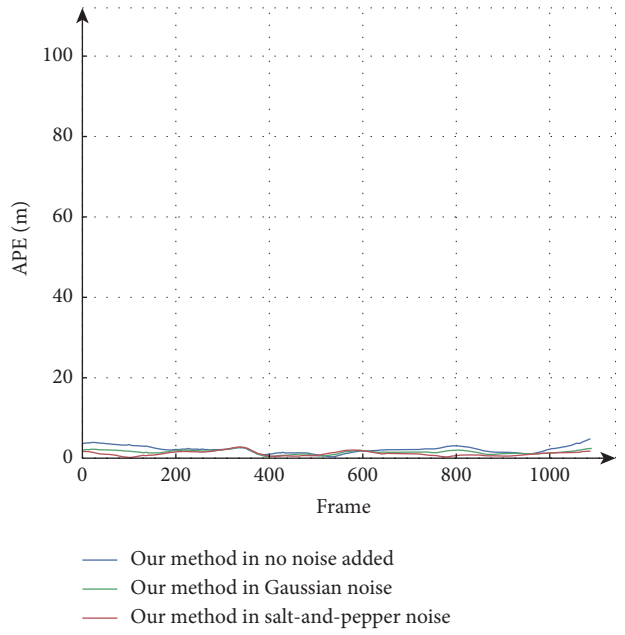
(c)

Figure 4: Example scenes for sequences "KITTI_00," "KITTI_01," and "KITTI_02" in the KITTI dataset: (a) is from the sequence "KITTI_00"; (b) is from the sequence "KITTI_01"; (c) is from the sequence "KITTI_02." The sequences "KITTI_00" and "KITTI_02" are the sequences with more camera overexposure interference, while "KITTI_01" is the sequence with little camera overexposure interference.



(a) (b)

Figure 5: Continued.



(c)

Figure 5: Comparison of APE with respect to translation in different noises on the sequence "KITTI_07": (a) APE of DSO. (b) APE of LDSO. (c) APE of our method. Note that the performance gap of our method is significantly smaller than that of DSO and LDSO. In particular, DSO and LDSO do not work in the sequence with Salt-and-Pepper noise, and the results in this case cannot be added for comparison. Besides, the performance of our method is better than the others overall.

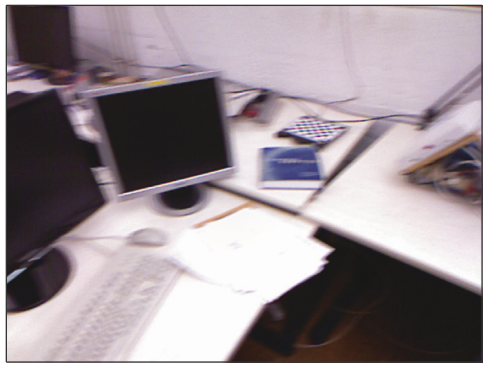


Figure 6: Blurred image with smears in TUM RGB-D dataset.

Table 4: $RMSE_{ATE}$ on TUM RGB-D dataset with no noise added.

Method	No noise added				
	fr1_xyz	fr2_xyz	fr2_rpy	fr1_desk	fr1_desk2
LDSO [34]	0.061	0.011	0.046	0.774	0.904
Ours	0.063	0.012	0.043	0.780	0.905

Although ORB-SLAM3 performs better in the face of Gaussian noise (see Table 2), ORB-SLAM3 is unavailable under the influence of Salt-and-Pepper noise (see Table 3). By contrast, the results show that our method performs more consistently in different camera imaging disturbances than other methods (see Tables 2 and 3).

To compare the robustness in different camera imaging disturbances more clearly, the absolute pose errors (APE) with respect to translation on the example sequence "KITTI_07" in different noises are shown in Figure 5. Here, the main reason for using the sequence "KITTI_07" as the example is that this sequence has a medium sequence length in the KITTI dataset. In the next part of this paper, the sequence "KITTI_07" is also used as the study object, where the reason is not further explained. The lack of the APE curves of DSO and LDSO in the sequence with Salt-and-Pepper noise is due to their inability to work. Notice that our method has more consistent APE curves in different noises, and all the APEs of our method are less than 5.0%. This experiment highlights that our strategy effectively improves

Table 5: $RMSE_{ATE}$ on TUM RGB-D dataset with Gaussian noise.

Method	Gaussian noise				
	fr1_xyz	fr2_xyz	fr2_rpy	fr1_desk	fr1_desk2
LDSO [34]	—	0.096	—	0.518	—
Ours	0.156	0.010	0.060	0.801	0.756

Note. "—" means tracking failure.

Table 6: $RMSE_{ATE}$ on TUM RGB-D dataset with Salt-and-Pepper noise.

Method	Salt-and-Pepper noise				
	fr1_xyz	fr2_xyz	fr2_rpy	fr1_desk	fr1_desk2
LDSO [34]	—	—	—	0.841	—
Ours	0.129	0.011	0.058	0.796	0.871

Note. "—" means tracking failure.

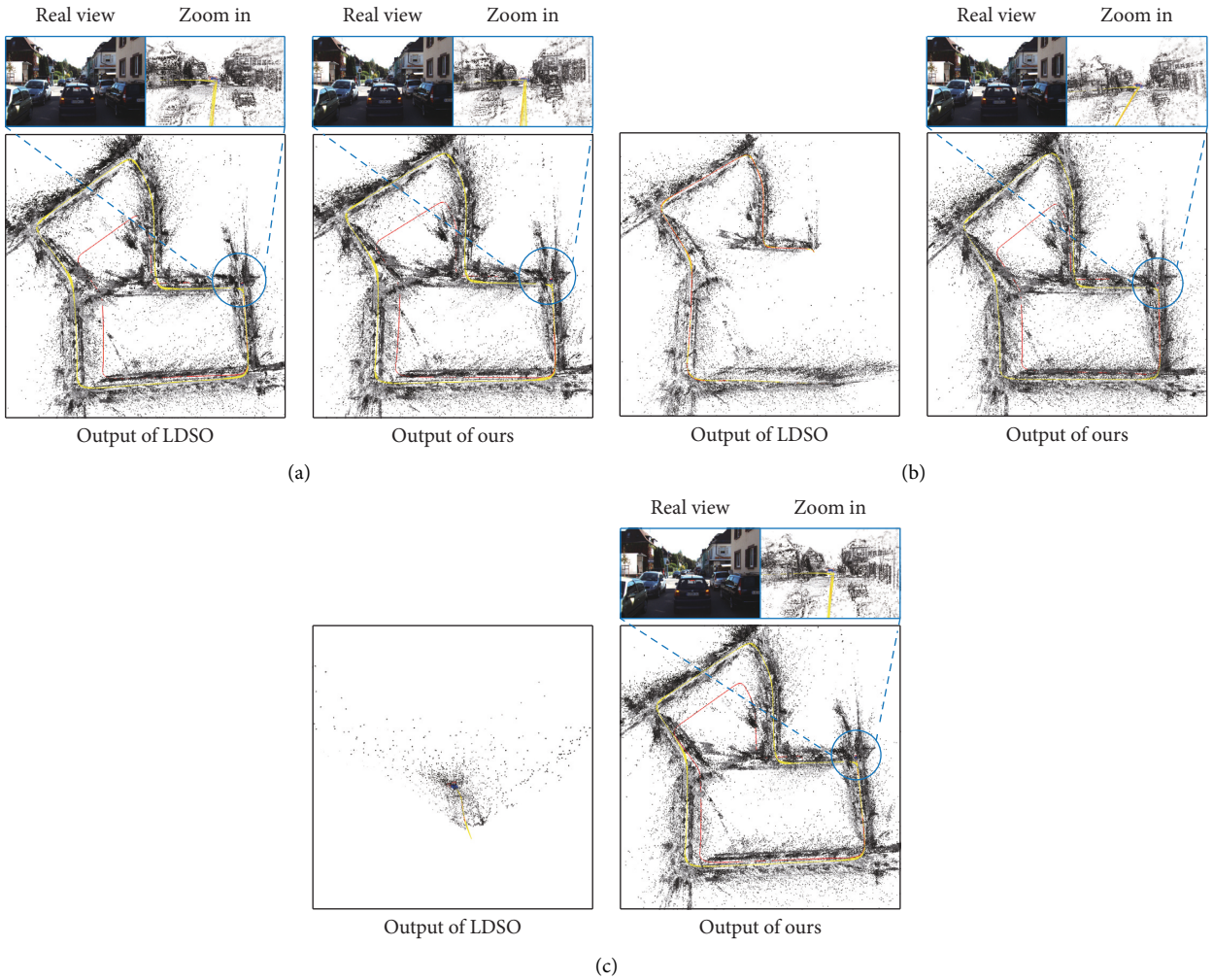


Figure 7: Sample outputs of the sequence "KITTI_07": (a), (b), and (c) are the outputs on the sequence with no added noise, Gaussian noise, and Salt-and-Pepper noise, respectively. Left: LDSO's outputs. Right: our method's outputs. Note that, in the sequence without adding noise, the quality of our method's trajectory estimation and point cloud map construction is similar to that of LDSO. In the sequence with Gaussian noise added to LDSO, the closed-loop cannot be detected, and the trajectory estimation in the second half is wrong. LDSO does not work in the sequence with Salt-and-Pepper noise added. Our strategy achieves a more robust performance under different noise interferences.

the robustness of direct SLAM when facing different camera imaging disturbances outdoors.

4.1.2. On the TUM RGB-D Dataset. Secondly, some experiments are conducted on the TUM RGB-D dataset to verify whether our strategy has the effect of improving robustness in indoor environments. Since DSO and LDSO perform very similarly in this case, our method is only compared with LDSO. In this dataset, there are many blurred images with smears, as shown in Figure 6. In this experiment, because the selected sequences are relatively short, the difference in RMSE is not apparent. Thus, we mainly compare whether the tracking of the SLAM system based on different methods is successful. The results are shown in Tables 4–6.

The results in this experiment show that the SLAM system will fail easily after adding noise to the images. Note that, in the sequences in which no noise is added, both our method and LDSO can track successfully (see Table 4). After adding different noises to the sequences, LDSO becomes more prone to failure tracking, while our method still tracks successfully (see Tables 5 and 6). This experiment highlights that our approach can still improve the robustness of direct SLAM under different camera imaging disturbances when faced with a poor indoor image input.

4.2. Qualitative Evaluation. This section mainly conducts a qualitative evaluation of the completeness and the clarity of the predicted trajectory map and the constructed point cloud map in the camera imaging disturbances. Examples of the point cloud map constructed on the sequence "KITTI_07" are shown in Figure 7. The results show that our method is similar to LDSO in the absence of noise interference. When disturbed by Gaussian noise and Salt-and-Pepper noise, LDSO is negatively affected to varying degrees, while our method has a better and more stable

performance in the trajectory prediction and point cloud map construction. The main reason is that our method uses the multilayer pixel fusion features based on the side window strategy instead of directly using the original pixels, which can improve the robustness of the direct method-based SLAM in different camera imaging disturbances.

5. Discussion

The total performances of the proposed method have been proved on different datasets by some comparison experiments in Section 4. In this section, some additional comparison experiments are conducted to discuss the performance of our method in different intensities of camera imaging disturbances. In addition, the performance of the key improvement of the proposed method, namely, the points selection strategy, is further discussed. At last, the proposed method is tested in real-world applications to demonstrate the effectiveness of the proposed method.

5.1. Performance in Camera Imaging Disturbances of Different Intensities. Firstly, the performance of our method in the camera imaging disturbances of different intensities is discussed, where some expanded comparison experiments are conducted under the sensor noise of different intensities and the camera overexposure with different frequencies.

5.1.1. About Different Noise Intensities. The performance of our method in the camera sensor noise of different intensities is discussed on the sequence "KITTI_07." The comparison experiments are carried out separately in Gaussian noise and Salt-and-Pepper noise with different intensities. The variance of Gaussian noise ranges from 0.001 to 0.009 and is incremented by a step size of 0.002. The rate of Salt-and-Pepper noise added ranges from 2% to 10%, and the step size is 2%. The results are shown in Tables 7 and 8. For Gaussian noise, DSO tracking fails when the variance is greater than 0.005. LDSO tracking fails when the variance is greater than 0.003. Our method tracks successfully at all noise intensities and performs stably when the variance is below 0.005. This reflects that our method is more robust than other direct methods in different intensities of Gaussian noise. For Salt-and-Pepper noise, both DSO and LDSO fail to track when the noise addition rate is greater than 2%. Our method can track successfully and perform stably at all noise addition rates. It can be seen that our method is more robust than other direct methods in different intensities of Salt-and-Pepper noise. ORB-SLAM3 can also track successfully in all intensities of Gaussian noise and performs stably when the variance is below 0.007. While ORB-SLAM3 outperforms our method in robustness under different intensities of Gaussian noise, it fails to track at all addition rates of Salt-and-Pepper noise.

Table 7: $RMSE_{ATE}$ comparison in Gaussian noise of different intensities.

Variance	DSO [21]	LDSO [34]	ORB-SLAM3 [54]	Ours
0.001	24.396	2.504	1.872	1.655
0.003	38.812	53.481	1.932	1.973
0.005	45.968	—	2.101	2.946
0.007	—	—	2.566	12.343
0.009	—	—	10.242	13.816

Note. "—" means tracking failure.

Table 8: $RMSE_{ATE}$ comparison in Salt-and-Pepper noise of different intensities.

Addition rate (%)	DSO [21]	LDSO [34]	ORB-SLAM3 [54]	Ours
2	35.500	35.525	—	1.409
4	—	—	—	1.602
6	—	—	—	1.755
8	—	—	—	2.225
10	—	—	—	2.238

Note. "—" means tracking failure.

Table 9: $RMSE_{ATE}$ comparison under interference of camera overexposure at different frequencies.

Interval frames	DSO [21]	LDSO [34]	ORB-SLAM3 [54]	Ours
30	32.946	11.866	—	11.522
25	34.257	12.248	—	11.836
20	—	22.240	—	11.885
15	—	—	—	13.333
10	—	—	—	—

Note. "—" means tracking failure.

5.1.2. About Different Overexposure Frequencies. To discuss the performance of our method under the interference of camera overexposure, the sequence "KITTI_01," which suffers little from camera overexposure, is experimented with adding simulated camera overexposure disturbance at different frequencies. The camera overexposure addition operation in this study is similar to other pieces of literature [56]. The number of interval frames at which overexposure interference is added ranges from 30 to 10 and is decreased by a step size of 5. The results are shown in Table 9.

The results in Table 9 show that our method performs close to LDSO when the camera overexposure interference is not very serious. However, when the overexposure interference interval is 20 frames, the proposed method outperforms the general LDSO method by more than 46%. In addition, LDSO starts to fail to track when the overexposure interference interval is lower than 15 frames, while our method can still work when the overexposure interference interval is bigger than 10 frames. ORB-SLAM3 fails to track in the sequence "KITTI_01" under the added camera overexposure interference. The results of this experiment show that the proposed method has better performance under the camera overexposure interference.

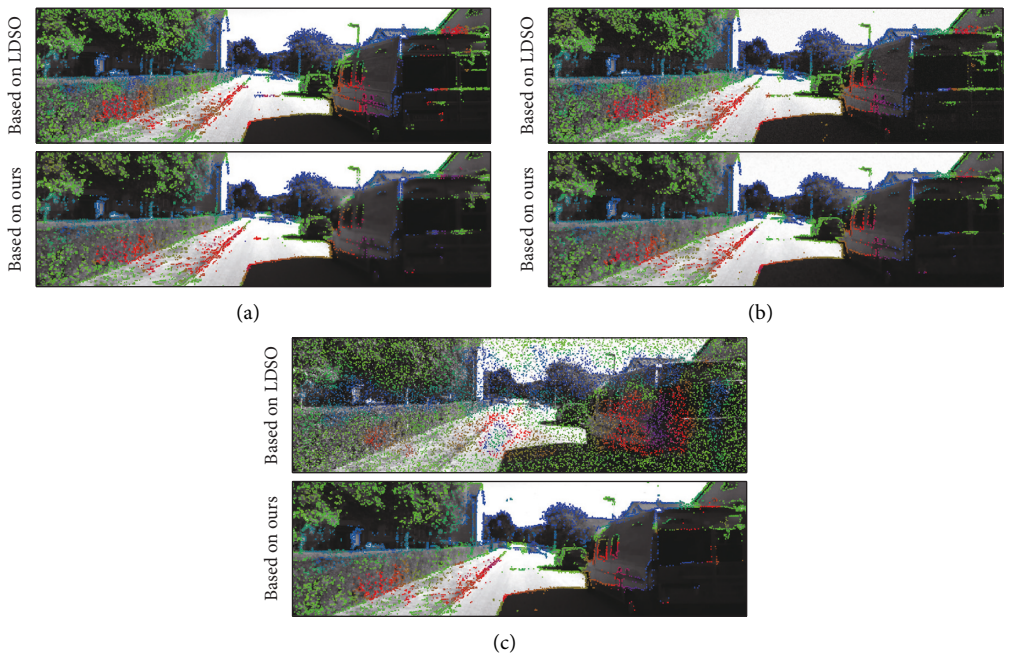
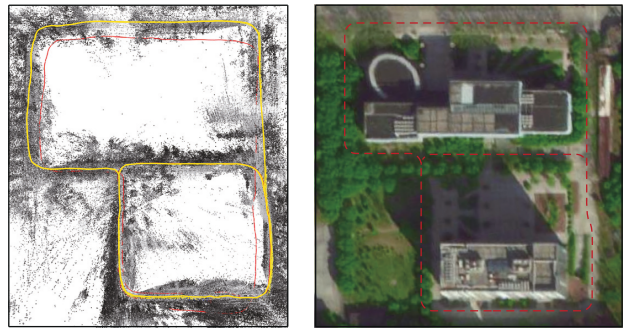


Figure 8: Point selection results of our strategy and LDSO in different noises: (a), (b), and (c) are the images from the KITTI dataset with no added noise, Gaussian noise, and Salt-and-Pepper noise, respectively. Top rows: point selection results of LDSO. Bottom rows: point selection results of our strategy. Note that the points selected by our strategy are more consistent in different noises. Moreover, on semistatic objects such as cars parked on the side of the road, the points selected by our approach are significantly less than those by LDSO and are mainly distributed on the apparent edges.

Table 10: $RMSE_{ATE}$ comparison of whether using semantic-based variable radius side window.

Method	No noise added		Gaussian noise		Salt-and-Pepper noise	
	KITTI_07	KITTI_08	KITTI_07	KITTI_08	KITTI_07	KITTI_08
FR-SW	2.256	106.652	2.794	112.754	2.471	106.093
SVR-SW	1.789	99.579	1.973	102.206	2.238	101.590

Figure 9: Some images in the real scene added with noise. (a) Added with Gaussian noise. (b) Added with Salt-and-Pepper noise.



(a) (b)

Figure 10: Result of experiment in real scene. (a) The trajectory estimated by our method, which is marked with a yellow curve. (b) The approximate trajectory on the satellite map, which is marked with a red dashed line.

5.2. About Points Selection Strategy. Secondly, the effect of the points selection strategy of our method to improve the robustness of direct method-based SLAM is discussed. Figure 8 shows the selection of points in an example scene with different types of noise. Here, our points selection strategy is compared with that of the general LDSO. It is easy to notice that the points selected by our strategy are more consistent in different noises. It is not easy for LDSO to detect closed loops under the influence of Gaussian noise. Gaussian noise creates texture in untextured areas. These textures are selected as the basis for closed-loop detection, which easily leads to the failure of closed-loop detection. In Salt-and-Pepper noise, LDSO is entirely inoperable. The reason is that the image-gradient-based features selected by LDSO are easily located at the position of the Salt-and-Pepper noise (see Figure 8(c)). These randomly generated noise positions cannot be used as the basis for estimating camera pose. As shown in Figure 8(a), the points selected by our method are significantly less than that by LDSO and are mainly distributed on the apparent edges of the semistatic objects such as cars parked on the side of the road. The consistent selection of points of our method improves the robustness of direct method-based SLAM.

The comparison results of $\mathrm{RMSE}_{\mathrm{ATE}}$ based on the proposed semantic-based variable radius side window (SVR-SW) and the fixed radius side window in the general LDSO (FR-SW) are shown in Table 10. Here, the sequences "07" and "08" of the KITTI dataset are used, which contain more semistatic objects. It can be noticed that the proposed SVR-SW strategy achieves better performance on different noises. The main reason is that the semantic-based variable radius side window can reduce the weight of selected points of semistatic objects to improve the performance of direct method-based SLAM in scenes with more semistatic objects.

5.3. Experiment in Real Scene. Thirdly, to discuss the performance of our method in real scenes, an experiment is conducted on a real-world dataset collected outdoors by the Zenmuse X5S camera mounted on the DJI Inspire 2 drone [57]. In reality, the camera imaging disturbances often do not exist all the time but are sudden and random. For simulation of this situation, Gaussian noise and Salt-and-Pepper noise are artificially added to parts of this dataset. Some images added with noise are shown in Figure 9, which have obvious brightness changes due to the shade of trees and lots of semistatic objects in the real scene, such as bicycles and cars. The real-world dataset is collected along the road to easily judge whether our method estimates the correct trajectory using the satellite map. The experimental result of this self-collected real dataset is shown in Figure 10. It can be seen that the trajectory estimated by our method does not deviate from the road due to the camera imaging disturbances, including the artificially added noise and the natural brightness changes. Our method performs good robustness on different camera imaging disturbances in real scenes.

6. Conclusion

The robustness in the camera imaging disturbances of the direct method-based SLAM is studied in this paper, and a concept of side windows is introduced into this visual SLAM system. Based on this concept, a multilayer stacked pixel blender is used to process the input images, which can significantly reduce the blurring effects on the edges of the images. In addition, the size of the fusion window can be adjusted based on semantic information to reduce the proportion of selected points on semistatic objects. At last, to more clearly evaluate the robustness of the proposed method under different camera imaging disturbances, the public datasets enhanced with different camera imaging disturbances are used to perform detailed quantitative and qualitative experiments. The results demonstrate that our strategy can improve the robustness of the direct method-based SLAM against the different camera imaging disturbances, including various sensor noises and camera overexposure. Furthermore, the results of the real-world experiment show that the proposed method can work efficiently in real-world applications. In the future, how to further improve the robustness of the visual SLAM method while improving efficiency by using different fusion methods should be studied, such as deep neural networks.

Data Availability

Publicly available datasets were analyzed in this study. These data can be found at http://www.cvlibs.net/datasets/kitti/eval_odometry.php and https://vision.in.tum.de/data/datasets/rgbd-dataset/download.

Conflicts of Interest

The authors declared that they have no conflicts of interest in this work.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61873086) and the Science and Technology Support Program of Changzhou (CE20215022).

References

[1] J. Chang, N Dong, and D. Li, "A real-time dynamic object segmentation framework for SLAM system in dynamic scenes," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, 2021.

[2] J. Ni, L. Wu, X. Yang, and X. Y. Simon, "Bioinspired intelligent algorithm and its applications for mobile robot control: a survey," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 3810903, 16 pages, 2016.

[3] Y. Ying, H. Yan, Z. Li, K. Feng, and X. Feng, "Loop closure detection based on image covariance matrix matching for visual SLAM," *International Journal of Control, Automation and Systems*, vol. 19, no. 11, pp. 3708–3719, 2021.

[4] J. Ni, C. Wang, X. Fan, and X. Y. Simon, "A bioinspired neural model based extended Kalman filter for robot SLAM," *Mathematical Problems in Engineering*, 905826, vol. 2014, 11 pages, 2014.

[5] H. Deilamsalehy and C. H. Timothy, "Sensor fused three-dimensional localization using IMU, camera and LiDAR," in *Proceedings of the IEEE Sensors*, Orlando, FL, USA, October 2016.

[6] W. Xie, P. Xiaoping Liu, and M. Zheng, "Moving object segmentation and detection for robust RGBD-SLAM in dynamic environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, 2021.

[7] J. Zhao, T. Li, Y. Tong, L. Zhao, and S. Huang, "2D laser SLAM with closed shape features: fourier series parameterization and submap joining," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1527–1534, 2021.

[8] J. Ni, Y. Chen, K. Wang, and X.Y Simon, "An improved vision-based SLAM approach inspired from animal spatial cognition," *International Journal of Robotics and Automation*, vol. 34, no. 5, pp. 491–502, 2019.

[9] T. H. Nguyen, T.-M. Xie, and L. Xie, "Tightly-coupled ultra-wideband-aided monocular visual SLAM with degenerate anchor configurations," *Autonomous Robots*, vol. 44, no. 8, pp. 1519–1534, 2020.

[10] Z. Liang and C. Wang, "A semi-direct monocular visual SLAM algorithm in complex environments," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 101, no. 1, 2021.

[11] H.-J. Liang, J Sanket, C. Aloimonos, and Y. Aloimonos, "Salientdso: bringing attention to direct sparse odometry," *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 4, pp. 1619–1626, 2019.

[12] J.-W. Kam, H.-S. Kim, S.-J. Lee, and S.-S. Hwang, "Robust and fast collaborative augmented reality framework based on monocular SLAM," *IEIE Transactions on Smart Processing and Computing*, vol. 9, no. 4, pp. 325–335, 2020.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[14] S. Sarhan, A. A. Nasr, and M. Y. Shams, "Multipose face recognition-based combined adaptive deep learning vector quantization," *Computational Intelligence and Neuroscience*, vol. 2020, Article ID 8821868, 11 pages, 2020.

[15] E. Rublee, R. Vincent, K. Kurt, and B. Gary, "ORB: an efficient alternative to SIFT or SURF," in *Proceedings of the 2011 International conference on computer vision*, pp. 2564–2571, IEEE, Barcelona, Spain, November 2011.

[16] E. Rosten, R. Porter, and T. Drummond, "Faster and better: a machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2008.

[17] M. Calonder, V. Lepetit, and M. Ozuysal, "BRIEF: computing a local binary descriptor very fast," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1281–1298, 2011.

[18] Ke Wang, S. Ma, R. Fan, and J. Lu, "SBAS: salient bundle adjustment for visual SLAM," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, 2021.

[19] J. Ni, T. Gong, Y. Gu, J. Fan, and X. Fan, "An improved deep residual network-based semantic simultaneous localization and mapping method for monocular vision robot," *Computational Intelligence and Neuroscience*, vol. 2020, Article ID 7490840, 14 pages, 2020.

[20] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: large-scale direct monocular SLAM," in *Proceedings of the 13th European Conference on Computer Vision, ECCV 2014*, pp. 834–849, Springer, Zurich, Switzerland, September 2014.

[21] 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, 2017.

[22] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "SVO: semidirect visual odometry for monocular and multicamera systems," *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 249–265, 2016.

[23] N. Yang, R. Wang, X. Cremers, and D. Cremers, "Challenges in monocular visual odometry: photometric calibration, motion bias, and rolling shutter effect," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2878–2885, 2018.

[24] K. Zhu, X. Jiang, Z. Gao, H. Fujita, and J.-N. Hwang, "Photometric transfer for direct visual odometry," *Knowledge-Based Systems*, vol. 213, Article ID 106671, 2021.

[25] P. Liu, X. Yuan, C. Song, C. Liu, and Z. Li, "Real-time photometric calibrated monocular direct visual SLAM," *Sensors*, vol. 19, no. 16, p. 3604, 2019.

[26] C. Sheng, S. Pan, W. Gao, Y. Tan, and T. Zhao, "Dynamic-DSO: direct sparse odometry using objects semantic information for dynamic environments," *Applied Sciences*, vol. 10, no. 4, p. 1467, 2020.

[27] L. Zhou, S. Kaess, and M. Kaess, "DPLVO: direct point-line monocular visual odometry," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7113–7120, 2021.

[28] C. Li, W. Li, Z. Wang, and Y. Wan, "Research on image feature extraction and matching algorithms for simultaneous localization and mapping," in *Proceedings of the 2021 IEEE*

Third International Conference on Communications, Information System and Computer Engineering, CISCE, pp. 370–376, Beijing, China, May 2021.

[29] H. M. Bruno and E. Colombini, "LIFT-SLAM: a deep-learning feature-based monocular visual SLAM method," *Neurocomputing*, vol. 455, pp. 97–110, 2021.

[30] C. Rafael Steffens, L. R. Vieira Messias, P. Lilles, J. Drews, and S. S. da Costa Botelho, "Can exposure, noise and compression affect image recognition? An assessment of the impacts on state-of-the-art ConvNets," in *Proceedings of the 2019 Latin American Robotics Symposium (LARS), 2019 Brazilian Symposium on Robotics (SBR) and 2019 Workshop on Robotics in Education (WRE)*, pp. 61–66, Rio Grande, Brazil, October 2019.

[31] B Paul, R. Wang, and D. Cremers, "Online photometric calibration of auto exposure video for realtime visual odometry and SLAM," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 627–634, 2018.

[32] J. Kim and A. Kim, "Light condition invariant visual SLAM via entropy based image fusion," in *Proceedings of the 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence, URAI*, pp. 529–533, Jeju, Republic of Korea, July 2017.

[33] Y. Kim, W. Chung, and D. Hong, "Indoor parking localization based on dual weighted particle filter," *International Journal of Precision Engineering and Manufacturing*, vol. 19, no. 2, pp. 293–298, 2018.

[34] X. Gao, R. Wang, N. Demmel, and D. Cremers, "LDSO: direct sparse odometry with loop closure," in *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pp. 2198–2204, Madrid, Spain, October 2018.

[35] L. Liu, "Image classification in htp test based on convolutional neural network model," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 6370509, 8 pages, 2021.

[36] J. Ni, Y. Chen, and Y. Chen, "A survey on theories and applications for self-driving cars based on deep learning methods," *Applied Sciences-Basel*, vol. 10, no. 8, 2020.

[37] 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, Article ID 103597, 2020.

[38] A. Krizhevsky, I. Hinton, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[39] D. Xu, X. Wang, G. Sun, and H. Li, "Towards a novel image denoising method with edge-preserving sparse representation based on laplacian of B-spline edge-detection," *Multimedia Tools and Applications*, vol. 76, no. 17, Article ID 17839, 2017.

[40] H. Yin, Y. Qiu, and G. Qiu, "Side window guided filtering," *Signal Processing*, vol. 165, pp. 315–330, 2019.

[41] L. Xiao, C. Fan, H. Ouyang, A. F. Abate, and S. Wan, "Adaptive trapezoid region intercept histogram based Otsu method for brain MR image segmentation," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 4, pp. 2161–2176, 2022.

[42] D. Zheng, L. Li, S. Chai et al., "A defect detection method for rail surface and fasteners based on deep convolutional neural network," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 2565500, 15 pages, 2021.

[43] J. Ni, K. Shen, Y. Chen, W. Cao, and S. X Yang, "An improved deep network-based scene classification method for self-driving cars," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, 2022.

[44] T.-Y Lin, M. Maire, and S. Belongie, "Microsoft COCO: common objects in context," in *Proceedings of the 13th European Conference on Computer Vision, ECCV 2014*, pp. 740–755, Zurich, Switzerland, September 2014.

[45] R. Jiang, H. Zhou, H. Wang, and S. S. Ge, "Road-constrained geometric pose estimation for ground vehicles," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 2, pp. 748–760, 2020.

[46] A. Geiger, P. Lenz, C. Urtasun, and R. Urtasun, "Vision meets robotics: the kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[47] S. Yang and S. Scherer, "Monocular object and plane SLAM in structured environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3145–3152, 2019.

[48] J. Cheng, H. Zhang, and Q.-H. Meng, "Improving visual localization accuracy in dynamic environments based on dynamic region removal," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 3, pp. 1585–1596, 2020.

[49] 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, 2020.

[50] S.-J. Ryu, M.-J. Lee, and H.-K. Lee, "Detection of copy-rotate-move forgery using zernike moments, Information Hiding," in *Proceedings of the International workshop on information hiding*, pp. 51–65, Springer, Calgary, Canada, June 2010.

[51] L. Calatroni and K. Papafitsoros, "Analysis and automatic parameter selection of a variational model for mixed Gaussian and salt-and-pepper noise removal," *Inverse Problems*, vol. 35, no. 11, 2019.

[52] G. Chahine and C. Pradalier, "Survey of monocular slam algorithms in natural environments," in *Proceedings of the 2018 15th Conference on Computer and Robot Vision, CRV*, pp. 345–352, Toronto, Canada, May 2018.

[53] Y. Miura and J. Miura, "RDS-SLAM: real-time dynamic SLAM using semantic segmentation methods," *IEEE Access*, vol. 9, Article ID 23772, 2021.

[54] C. Campos, R. Elvira, and J. J. G. . Rodríguez José, M. M. Montiel and D. T. Juan, Orb-Slam3: An accurate open-source library for visual, visual–inertial, and multimap SLAM," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[55] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proceedings of the International Conference on Intelligent Robot Systems (IROS)*, Vilamoura-Algarve, Portugal, October 2012.

[56] C. Hu, B. B. Sapkota, J. Alex Thomasson, and M. V. Bagavathiannan, "Influence of image quality and light consistency on the performance of convolutional neural networks for weed mapping," *Remote Sensing*, vol. 13, no. 11, 2021.

[57] S. Hasan, M. Digman, and D. L. Brian, "Utility of a commercial unmanned aerial vehicle for in-field localization of biomass bales," *Computers and Electronics in Agriculture*, vol. 180, 2021.