


RESEARCH ARTICLE

SLC-VIO: a stereo visual-inertial odometry based on structural lines and points belonging to lines

Chenchen Wei¹ , Yanfeng Tang¹, Lingfang Yang² and Zhi Huang^{1,*}

¹College of Mechanical and Vehicle Engineering, Hunan University, 2nd South Lushan Road, 410009, Changsha, China and

²College of Civil Engineering, Hunan University, 2nd South Lushan Road, 410009, Changsha, China

*Corresponding author. E-mail: huangzhi@hnu.edu.cn

Received: 3 June 2021; **Revised:** 26 October 2021; **Accepted:** 10 December 2021; **First published online:** 17 January 2022

Keywords: point belonging to a line, Manhattan world assumption, structural line features, simultaneous localization and mapping (SLAM), visual-inertial odometry (VIO)

Abstract

To improve mobile robot positioning accuracy in building environments and construct structural three-dimensional (3D) maps, this paper proposes a stereo visual-inertial odometry (VIO) system based on structural lines and points belonging to lines. The 2-degree-of-freedom (DoF) spatial structural lines based on the Manhattan world assumption are used to establish visual measurement constraints. The property of point belonging to a line (PPBL) is used to initialize the structural lines and establish spatial distance-residual constraints between point and line landmarks in the reconstructed 3D map. Compared with the 4-DoF spatial straight line, the 2-DoF structural line reduces the variables to be estimated and introduces the orientation information of scenes to the VIO system. The utilization of PPBL makes the proposed system fully exploit the prior geometric information of environments and then achieves better performance. Tests on public data sets and real-world experiments show that the proposed system can achieve higher positioning accuracy and construct 3D maps that better reflect the structure of scenes than existing VIO approaches.

1. Introduction

Simultaneous localization and mapping (SLAM), aiming to estimate the moving system's pose and construct a 3D map for the unknown environments, is widely used in applications such as self-driving cars, AGV, and unmanned aerial vehicles [1–3]. Among all SLAM techniques, the visual SLAM, which utilizes a camera as the primary sensor, has attracted more and more attention due to its simple configuration and low cost. In the last decade, various visual SLAM programs have been proposed, such as PTAM [4], SVO [5], and ORB-SLAM [6]. Compared with the pure vision methods, with the aid of an inertial measurement unit (IMU), the visual-inertial odometry (VIO) [7–9] can achieve better accuracy and robustness.

Most current popular SLAM/VIO systems only use point features as landmarks and rarely use the environment's prior geometric information. When the scene is poorly texture or has weak illumination, the quality of point features worsens and results in a large drift of the reconstructed map and low positioning accuracy. In such cases, line features are good complements to point landmarks. Compared with point features, line features can better depict the geometric structure information of the environment. More importantly, as shown in Fig. 1, in such a structural building scene that can be seen everywhere, most spatial straight lines are parallel or orthogonal to each other. These parallel or orthogonal structural lines encode the global orientation information of the scene. If these structural lines are used as landmarks in a VIO system, the accumulated orientation errors can be eliminated, thereby improving the positioning accuracy.

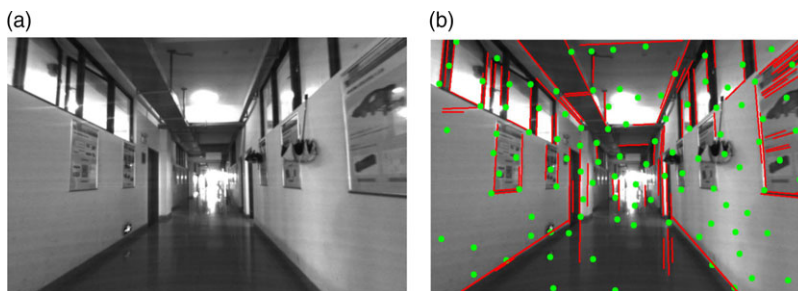


Figure 1. (a) Typical structured environment; (b) point features and line features extracted from the environment.

The Manhattan world assumption [10] can be used to describe such a structural scene. In a Manhattan world frame, the three axes are orthogonal to each other, and all structural lines are aligned with the directions of the three axes. A Manhattan world frame can be seen as a structural scene with a unified orientation. The structural scene is modeled as a Manhattan world, and its global orientation can be roughly estimated according to the image observations of structural lines and optimized later. With the Manhattan world assumption, the structural lines can be parameterized simply.

Previous works [11–13] have demonstrated that using both points and lines as landmarks in SLAM systems can achieve better performance than only using one of them. However, researchers seldomly consider the correlation between points and lines. They often separately establish the visual measurement residual constraints of points and lines and then add these residuals to a unified optimization framework. Such operations mean that the point features and line features are treated independently. However, as shown in Fig. 1(b), most point features belong to the line features. If the property of point belonging to a line (PPBL) can be included, more prior geometric information can be introduced into the VIO system. On the one hand, it facilitates the initialization of structural line features. On the other hand, the distance constraints between a feature point and the line landmarks it belongs to in the reconstructed 3D map can be established as residual items added to optimization, thereby further reducing mapping errors and improving the positioning accuracy.

Based on the ideas mentioned above, this paper presents the SLC-VIO system, a stereo VIO that takes structural lines and points as landmarks and utilizes the PPBL. The experiments have been conducted on both public data sets and in real-world scenes to test the performance of the proposed system. The results show that compared to the existing VIO methods that do not consider the structural regularity of environments, the proposed SLC-VIO achieves higher positioning accuracy and constructs 3D maps that better reflect the structure of scenes. The main contributions of this paper are as follows:

- (1) We take structural lines as additional landmarks in the optimization-based VIO system. The 2-DoF spatial structural lines are defined based on the Manhattan world assumption. Moreover, the Jacobian matrices related to structural lines are derived.
- (2) We take into account the property of PPBL. This property is used to initialize the spatial structural lines and establish distance-residual constraints between spatial point and line landmarks in the reconstructed map.
- (3) The positioning accuracy and mapping performance of the proposed SLC-VIO were tested on some public data sets and real-world experiments and compared with VINS-Fusion [8] and PL-VIO [14].

2. Related work

Although point feature-based methods are popular in visual SLAM/VIO systems, some researchers have also tried to use line features as landmarks for pose estimation in the early years. In 1997, Neira et al.

[15] first proposed a monocular SLAM system using vertical line segments as landmarks to construct a 2D map. Their system utilizes two endpoints to denote a spatial straight line and optimizes the estimated variables based on the EKF (Extended Kalman filter) framework. Gee et al. [16] proposed a real-time UKF-based (Unscented Kalman filter) SLAM system using line segments, where the spatial line segments are also denoted by two endpoints. Recently, Ruben et al. [12] proposed PL-SLAM, a stereo SLAM system using both point and line features. Their system uses LSD [17] and LBD [18] algorithms to detect and match 2D line features in images. The estimated variables are refined by the nonlinear optimization in the system, which minimizes a cost function composed of the re-projected errors of both points features and line features. However, they used two endpoints to denote a spatial straight line. The disadvantage of parameterizing spatial straight lines by endpoints is obvious: a spatial straight line has 4 DoF, while two endpoints introduce six parameters, thereby resulting in over-parameterization. It increases the number of variables to be optimized and makes the convergence worse during optimization.

To avoid over-parameterization, Bartoli et al. [19] proposed the orthonormal representation, which uses four parameters to define a spatial straight line. Since then, the orthonormal representation has been adopted in many SLAM systems, which takes the line features as landmarks. In these systems, they often use Plücker coordinates to denote spatial straight lines when calculating re-projected errors and then convert the Plücker coordinates to orthogonal representation during optimization. Apart from pure visual SLAM systems, some researchers have integrated line features into VIO systems. Zheng et al. [20] proposed a tightly coupled filtering-based stereo VIO system using both points and lines. Nevertheless, they still used two endpoints to represent a spatial straight line. In a recent work called PL-VIO [14], built upon VINS [8], a state-of-the-art VIO system, they used both points and lines as landmarks and adopted the Plücker coordinates and orthogonal representation for line parameterization.

In recent years, some researchers considered using structural regularity in man-made building scenes to improve SLAM/VIO performance. In some pure visual SLAM [21] systems or VIO [22] systems, they used vanishing points to reduce the accumulated orientation errors, thereby improving positioning accuracy. However, they only used line features detected in images to calculate vanishing points and then used the vanishing points to get the global orientation information of scenes. They did not use lines as landmarks. Kim et al. [23] proposed a SLAM method using vertical lines detected from an omni-direction camera image. In ref. [24], Zhang et al. proposed a monocular SLAM system that used vertical lines and floor lines as landmarks. Besides, they also used vanishing points to reduce accumulated heading error and to perform loop closing. Zhou et al. [25] proposed an EKF-based visual SLAM system using the building's structure lines. In their system, each structural line is represented by a point on a parameterizing plane and a dominant direction. Another VIO system [26] also uses similar methods for representing structural lines. In the recent work named StructVIO [27], Zou et al. proposed an EKF-based VIO system that adopts multiple Manhattan worlds to model the structural scenes. In their work, structural lines are defined in a local Manhattan world, which allows their system to deal with structural lines in multiple different orientations. However, among all these SLAM/VIO systems that adopt both points and lines or structural lines as landmarks, the correlation between points and lines is not considered.

3. System overview

The framework of the proposed SLC-VIO system is shown in Fig. 2. It is divided into two sections: measurement processing and sliding window optimization. The system starts with the measurement processing, where the measurements from IMU and stereo images are processed. By propagating the IMU measurements forward, the initial value of the latest IMU pose can be obtained. In addition, the residual for preintegrating IMU measurements within two consecutive camera frames are added to sliding window optimization. The point and line features are detected and tracked by two separate threads in stereo image processing. Based on the detected 2D line features, the system detects the Manhattan world and identifies structural lines. Then, according to the image observations in images, the system

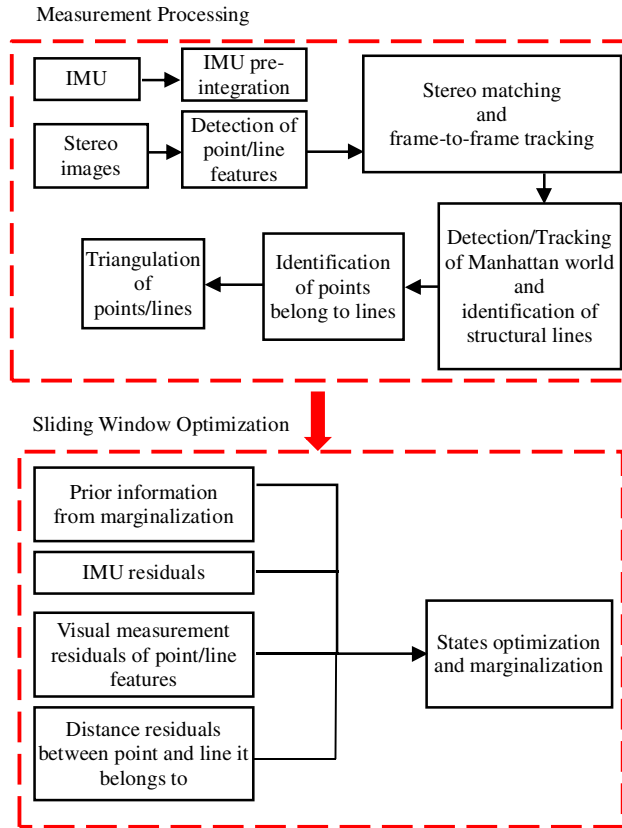


Figure 2. Flowchart of the proposed SLC-VIO.

identifies the PPBL. Finally, points and structural lines are triangulated to obtain an initial estimation of their spatial position.

The sliding window optimization proceeds with a sliding window-based tightly coupled optimization framework that fuses the preintegration constraints of IMU, the visual measurement constraints of point and structural line features, and the distance constraints between the point and the line landmarks it belongs to. In the sliding window, to limit the size of the state vector, marginalization is adopted. Some measurements related to marginalized states are converted into prior information.

4. Definition of structural lines

A structural scene is modeled as a Manhattan world. The Manhattan world frame $\{M\}$ is established with its origin coinciding with the origin of the global world frame $\{W\}$ where odometry starts. The three coordinate axes of the Manhattan world frame are aligned with the structural lines. Especially, the Z-axis of the Manhattan world frame is aligned with the vertical lines. The orientation of the Manhattan world frame in the global world frame can be initially estimated according to the image observations of structural lines. Figure 3 shows the Manhattan world model and the structural lines in it.

For a spatial structural line, regardless of its endpoints, there must be an intersection point on the Manhattan world frame's coordinate plane. For example, as shown in Fig. 3, L_0 is a vertical line and intersects the X-Y plane of the Manhattan world frame $\{M\}$ with point $P_{L_0}^M = (a_0, b_0, 0)^T$. The direction vector of L_0 in the Manhattan world frame is $d_{L_0}^M = (0, 0, 1)^T$. Similarly, for the horizontal structural lines such as L_1 and L_2 , their intersection points on the coordinate planes of $\{M\}$ are $P_{L_1}^M = (a_1, 0, b_1)^T$ and

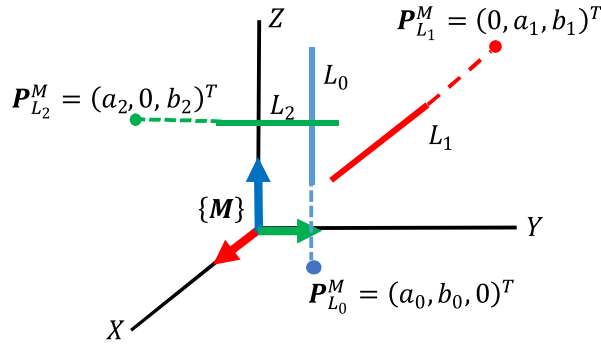


Figure 3. Illustration of the Manhattan world model and the structural lines. L_0 , L_1 , and L_2 are 3D structural lines. $P_{L_0}^M$, $P_{L_1}^M$, and $P_{L_2}^M$ are their respective intersection point on coordinate planes of the Manhattan world frame $\{M\}$.

$P_{L_2}^M = (0, a_2, b_2)^T$, respectively, and their direction vectors in $\{M\}$ are $d_{L_1}^M = (0, 1, 0)^T$ and $d_{L_2}^M = (1, 0, 0)^T$, respectively. Generally, the direction vector $d_{L_i}^M$ of a spatial structural line L_i can be obtained by identifying structural line features from all detected line features in the image and then used as fixed parameters. Each spatial structural line is considered as an infinitely long straight line. So, as long as the two nonzero parameters (a_i, b_i) of the intersection point $P_{L_i}^M$ are determined, the 3D structural line L_i in the Manhattan world can also be determined. As a result, a spatial straight line with 4-DoF becomes a structural line with only 2-DoF in the Manhattan world frame.

In the following descriptions, in order not to consider the camera intrinsic parameters, all image observations are transformed from pixel coordinates to homogeneous coordinates in the camera frame. To re-project a spatial structural line L_i onto the homogeneous coordinate plane in the camera frame $\{C_i\}$ (i is the sequence number of camera frames in sliding window), it requires to transform its intersection point $P_{L_i}^M$ and direction vector $d_{L_i}^M$ from $\{M\}$ to the camera frame $\{C_i\}$.

$$P_{L_i}^{C_i} = R_{WC_i}^{-1} (R_{WM} P_{L_i}^M - t_{WC_i}) \tag{1}$$

$$d_{L_i}^{C_i} = R_{WC_i}^{-1} R_{WM} d_{L_i}^M. \tag{2}$$

R_{WM} in (1) and (2), a rotation matrix, represents the orientation of the Manhattan world frame $\{M\}$ in the global world frame $\{W\}$. R_{WC_i} and t_{WC_i} represent the orientation and translation of the camera frame $\{C_i\}$ in the global world frame $\{W\}$, respectively. $P_{L_i}^{C_i} = (p_1, p_2, p_3)^T$ and $d_{L_i}^{C_i} = (d_1, d_2, d_3)^T$ are coordinates of the intersection point and direction vector in the camera frame $\{C_i\}$, respectively. The corresponding ones in homogeneous coordinates are $\mathcal{P}_{L_i}^{C_i} = (p_1/p_3, p_2/p_3, 1)^T$ and $d_{L_i}^{C_i} = (d_1/d_3, d_2/d_3, 1)^T$, respectively.

The homogeneous coordinate of a direction vector in the camera frame represents a vanishing point that is a common intersection point of a set of observed 2D line features. The spatial lines corresponding to the set of observed 2D line features are aligned with this direction vector. So, the homogenous coordinate $d_{L_i}^{C_i}$ is a vanishing point corresponding to the direction vector $d_{L_i}^M$.

Therefore, the theoretical re-projected line $l_i^{C_i}$ of the spatial structural line L_i on the homogeneous coordinate plane is given by

$$l_i^{C_i} = \mathcal{P}_{L_i}^{C_i} \times d_{L_i}^{C_i}. \tag{3}$$

With the above definitions, the re-projected line is expressed as

$$l_i^{C_i} = f(P_{L_i}^M, d_{L_i}^M, R_{WM}, R_{WC_i}, t_{WC_i}) \tag{4}$$

In a VIO system, the camera pose is usually represented by the IMU pose. Therefore, the above expression is further rewritten as

$$l_i^{C_i} = f(P_{L_i}^M, d_{L_i}^M, R_{WM}, R_{WI_i}, t_{WI_i}, R_{IC}, t_{IC}) \tag{5}$$

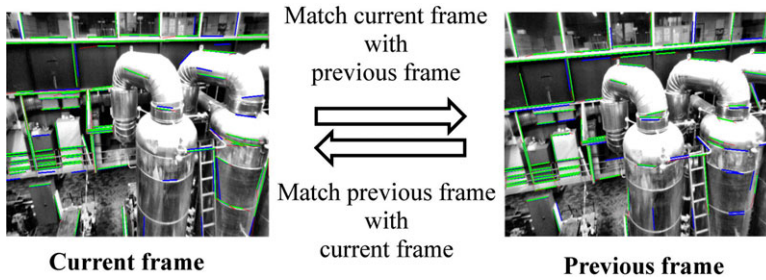


Figure 4. Illustration of the bidirectional matching strategy. The current frame is first matched with the previous frame. And then, the previous frame is matched with the current frame. The green lines are successful matches, and the blue lines are failed matches.

where \mathbf{R}_{Wl_i} and \mathbf{t}_{Wl_i} denote the IMU pose in the global world frame $\{W\}$. \mathbf{R}_{IC} and \mathbf{t}_{IC} are extrinsic parameters between the IMU and the camera, which can be obtained by calibration. If only considering the variables to be refined in the VIO system, the simplified version of the above expression is as follows:

$$l_i^{C_i} = f(\mathbf{P}_{L_i}^M, \mathbf{R}_{WM}, \mathbf{R}_{Wl_i}, \mathbf{t}_{Wl_i}) \quad (6)$$

As a result, the relationship between the spatial structural line L_i and the corresponding 2D re-projected line $l_i^{C_i}$ is established.

5. Measurement processing

The measurement processing involves both inertia and visual measurements. The visual measurements involve point and structural line features. Here we only present the details of the processing of structural line features. The processing of IMU and point features can be found in VINS [8].

5.1. Detection and tracking of line features

The LSD line detector [28] is employed to detect line features from images. For stereo matching and frame-to-frame tracking, the binary descriptor of the LBD method [18] is used to find correspondences among line features in different images. To improve the match of line features, as shown in Fig. 4, a bidirectional matching strategy is presented.

First, we match line features in the current frame with line features in the previous frame by the LBD descriptor with a relatively loose threshold to get as many candidate matches as possible. Second, some outliers are removed according to geometric constraints between the two matched line features, such as the difference of orientation and length, and the distance between their respective endpoints. Lastly, we match line features in the previous frame to line features in the current frame with the same method used in the first matching. For a line feature, only when the correspondences obtained in the bidirectional matching are the same, its match is regarded as correct. The bidirectional matching strategy is also used when matching left and right stereo images. The bidirectional matching strategy can improve the accuracy of matching considerably without reducing the number of candidate matches.

5.2. Detection of the Manhattan world and identification of structural line features

Detection of the Manhattan world is conducted independently in the first ten left images during the initialization. Due to errors in the extraction of vanishing points, more than one Manhattan world could be detected. The most frequently detected Manhattan world is identified as the global Manhattan world. It involves clustering 2D line features into three groups to identify structural lines in three orthogonal

directions based on the vanishing points. Our system clusters line features by the RANSAC algorithm to identify structural line features and obtain a rough estimation of the corresponding vanishing points. After that, the vanishing points are refined by nonlinear least-squares optimization. The details about the identification of structural lines by the RANSAC clustering algorithm and the optimization of vanishing points can be found in ref. [29].

The normalized coordinates of a vanishing point are also the direction vectors of the corresponding spatial structural lines in the camera frame [30]. If the normalized coordinates of the three detected vanishing points are approximately orthogonal, the Manhattan world is detected in the current image. Due to errors in observation and calculation, the three vanishing points need to be orthogonalized by the Schmidt method.

$$\begin{aligned}
 V_X &= V'_X \\
 V_Y &= V'_Y - \frac{V_X^T V'_Y}{V_X^T V_X} V_X \\
 V_Z &= V'_Z - \frac{V_X^T V'_Z}{V_X^T V_X} V_X - \frac{V_Y^T V'_Z}{V_Y^T V_Y} V_Y
 \end{aligned} \tag{7}$$

where V'_i ($i = X, Y, Z$) are the normalized coordinates of the three detected vanishing points, V_i ($i = X, Y, Z$) are the results of the orthogonalization.

After that, the orientation R_{C_iM} of the detected Manhattan world frame $\{M\}$ relative to the current camera frame $\{C_i\}$ is denoted by

$$R_{C_iM} = [V_X, V_Y, V_Z] \tag{8}$$

where the column vectors of R_{C_iM} are three unit vectors obtained from the homogeneous coordinates of the three orthogonal vanishing points. Among them, the vertical direction is set as V_z , and the direction close to the camera heading is set to V_x . Then, the rotation matrix R_{WM} is given by

$$R_{WM} = R_{W_{C_i}} R_{C_iM}, \tag{9}$$

where the rotation matrix $R_{W_{C_i}}$ can be initially estimated by the EPnP [31] method.

The rotation matrix R_{WM} is obtained by the above procedures on each left image during system initialization. When the difference between the obtained rotation matrices is less than a preset threshold, it is regarded that the Manhattan worlds detected in each image are the same. In such a case, the Manhattan world is successfully detected, and its orientation is also roughly obtained. After initialization, the Manhattan world orientation R_{WM} is further refined in sliding window optimization.

A straightforward method, rather than clustering, is used to identify structural line features in the subsequent images. Once the rotation matrices R_{WM} of the Manhattan world and $R_{W_{C_i}}$ of the current camera frame have been estimated, the orientation of the Manhattan world $\{M\}$ relative to the current camera frame $\{C_i\}$ is given by

$$R_{C_iM} = R_{W_{C_i}}^{-1} R_{WM} \tag{10}$$

Then, with the rotation matrix $R_{W_{C_i}}$, we can get three vanishing points corresponding to three coordinate axis directions of the Manhattan world frame in the current image. Note that, when detecting a Manhattan world during initialization, three vanishing points are used to obtain the rotation matrix R_{C_iM} , whereas here, the rotation matrix R_{C_iM} is used to obtain three vanishing points.

For a newly detected line feature in the current image, an auxiliary line connecting the line feature's midpoint and one of the vanishing points is drawn. By checking the angle between the auxiliary line and the line feature, it can be determined that whether the line feature is a structural line that belongs to the vanishing point. In this way, the structural lines can be identified from all newly detected line features.

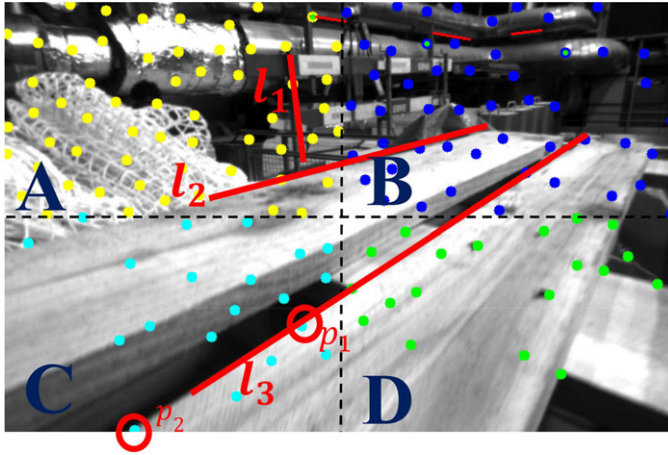


Figure 5. Illustration of the line features' search areas. The search area of l_1 , l_2 , and l_3 are area A, area A+B and area B+C+D, respectively. p_1 and p_2 are point features that may belong to l_3 .

5.3. Identification of points belonging to lines

The points belonging to lines are identified by two steps:

- (1) A point feature is classified into one of four areas according to its pixel coordinates. A line feature is classified into areas according to the coordinates of points on it, as illustrated in Fig. 5.
- (2) The PPBL between candidate point features and line features in the same area is checked and identified if both the following two conditions are satisfied:

Condition A: The vertical distance between the line feature and the point feature is less than a preset threshold.

Condition B: The point feature is inside the line feature, such as p_1 inside l_3 in Fig. 5; or the point feature is outside the line feature, such as p_2 outside l_3 in Fig. 5, but the distance from the point feature to the nearest endpoint of the line feature is less than a preset threshold.

When a point feature and a line feature satisfy the above two conditions, it is deemed that the point belongs to the line. However, due to the influence of visual angle, the PPBL in one image does not mean an authentic one. Only when the PPBL is detected in both left and right images and continues for at least four consecutive frames, the PPBL is verified.

5.4. Triangulation of structural lines

The direction of a structural line L_l in the Manhattan world frame $\{M\}$ can be directly obtained. Therefore, triangulation is carried out to acquire the initial value of the intersection point $P_{L_l}^M$. There are two ways to triangulate the structural lines, one is to triangulate by the points belonging to lines and the other is to triangulate by the midpoints. And for a structural line, if there exist triangulated point features belonging to it, its triangulation can be simplified.

As shown in Fig. 6, P_0 belongs to line L_0 and has been triangulated. To triangulate the structural line L_0 , P_0 is transformed from the global world frame $\{W\}$ to the Manhattan world frame $\{M\}$.

$$P_{P_0}^M = R_{WM}^{-1} P_{P_0}^W \tag{11}$$

$P_{P_0}^W$ in (10) are the coordinates of P_0 in the global world frame $\{W\}$. And $P_{P_0}^M = (p_1, p_2, p_3)^T$ are the coordinates of P_0 in the Manhattan word frame $\{M\}$. Assuming that L_0 is identified as a structural line

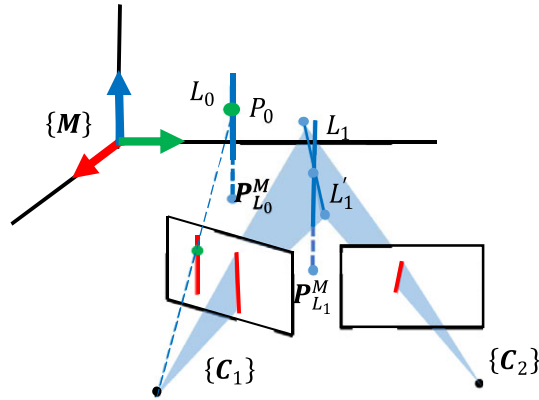


Figure 6. Triangulation of structural lines. P_0 has been identified as belonging to structural line feature L_0 . The red lines are observations of structural lines in the image, and the blue dotted lines are the results of the triangulation.

parallel to the Z-axis of the Manhattan world frame $\{M\}$, so its corresponding intersection point is $P_{L_0}^M = (p_1, p_2, 0)^T$. Similarly, for structural lines in other directions, the initial value of their intersection points $P_{L_l}^M$ can also be easily obtained according to triangulated points belonging to them.

For other structural lines, like L_1 in Fig. 6, if there are no triangulated points belonging to them, an initial estimation for the endpoints is obtained by a conventional method [14]. Due to the calculation and observation errors, the direction of the 3D line L'_1 composed of these two endpoints is not precisely parallel to the direction obtained in the identification procedures. So, in the Manhattan world frame $\{M\}$, the midpoint of these two endpoints is projected onto the plane perpendicular to the structural line. And then, the projection point is taken as the intersection point $P_{L_1}^M = (a_0, b_0, 0)^T$ ($a_0 = \frac{P_{sX}^M + P_{eX}^M}{2}$, $b_0 = \frac{P_{sY}^M + P_{eY}^M}{2}$) of the structural line L_1 .

6. Sliding window optimization

After the initial values of camera pose and landmarks position are estimated by the measurement processing, the state variables are optimized in a tightly coupled sliding window. The landmarks consist of spatial point features and structural line features.

6.1. Sliding window formulation

Figure 7 illustrates the sliding window formulation. The state vector in the sliding window is defined as

$$\begin{aligned} \chi &= [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{R}_{WM}, \lambda_0, \lambda_1, \dots, \lambda_m, L_0, L_1, \dots, L_k] \\ \mathbf{x}_i &= [\mathbf{R}_{Wl_i}, t_{Wl_i}, \mathbf{v}_{Wl_i}, \mathbf{b}_a, \mathbf{b}_g], i \in [0, n] \\ L_l &= [\theta_l, \rho_l]^T = \left[\tan^{-1}(b_l/a_l), 1/\sqrt{a_l^2 + b_l^2} \right]^T, l \in [0, k] \end{aligned} \tag{12}$$

where \mathbf{x}_i represents the IMU states, including rotation \mathbf{R}_{Wl_i} , position t_{Wl_i} , velocity \mathbf{v}_{Wl_i} in the global world frame, biases of acceleration \mathbf{b}_a , angular velocity \mathbf{b}_g in the IMU body frame at the i^{th} time step when an image is captured, and n is the number of keyframes in the sliding window. Since the use of polar coordinate and inverse depth $[\theta_l, \rho_l]^T$ has better optimization performance [27], we convert Cartesian

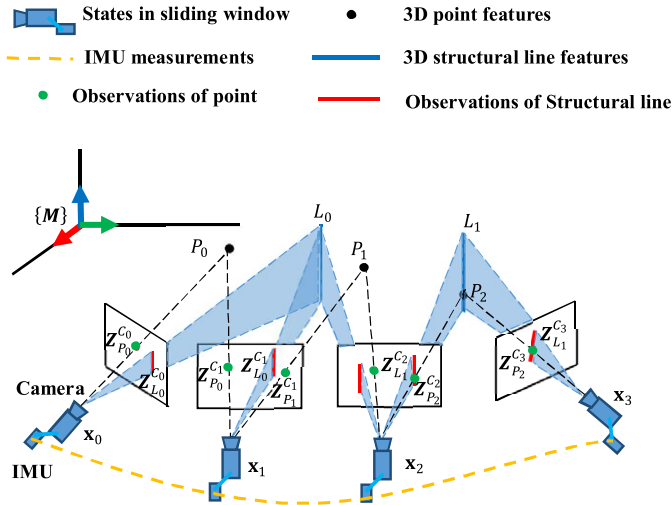


Figure 7. Illustrating the sliding window formulation. Among the points and lines, P_2 belongs to the structural line L_1 . The state variables to be optimized are the IMU states, the spatial position of points and structural lines, and the orientation of the Manhattan world frame.

coordinate parameters $[a_i, b_i]^T$ into $[\theta_i, \rho_i]^T$. This work is under a keyframe-based paradigm, and the selection strategy of keyframe is the same as VINS [7]. m and k are the numbers of spatial point features and structural line features observed by keyframes in the sliding window, respectively. \mathbf{R}_{WM} represents the orientation of the Manhattan world frame $\{M\}$ in the global world frame $\{W\}$. λ_p is the inverse depth of the p^{th} spatial point feature from its first observed keyframe. L_l represents two nonzero parameters of the intersection point $P_{L_l}^M$ of the l th spatial structural line in the Manhattan world frame $\{M\}$. Considering numerical stability [32], the inverse depth representation (θ_i, ρ_i) , instead of (a_i, b_i) , is used as a parameter of structural lines.

All the state variables mentioned above are optimized in the sliding window by minimizing the sum of cost functions:

$$\min_{\chi} \left\{ \|\mathbf{r}_p - \mathbf{H}_p \chi\|_{\Sigma_p}^2 + \sum_{i \in \mathcal{B}} \|\mathbf{r}_B(\mathbf{Z}_{b_i b_{i+1}}, \chi)\|_{\Sigma_{b_i b_{i+1}}}^2 + \sum_{(i,j) \in \mathcal{P}} \rho \left(\|\mathbf{r}_{\mathcal{P}}(\mathbf{Z}_{P_j}^{C_i}, \chi)\|_{\Sigma_{P_j}^{C_i}} \right) + \sum_{(i,l) \in \mathcal{L}} \rho \left(\|\mathbf{r}_{\mathcal{L}}(\mathbf{Z}_{L_l}^{C_i}, \chi)\|_{\Sigma_{L_l}^{C_i}} \right) + \sum_{(j,l) \in \mathcal{C}} \rho \left(\|\mathbf{r}_{\mathcal{C}}(\mathbf{Z}_{P_j}^{C_i}, \chi)\|_{\Sigma_{P_j}^{C_i}} \right) \right\} \quad (13)$$

where $\{\mathbf{r}_p, \mathbf{H}_p\}$ are the prior information and information matrix obtained after marginalizing out a camera frame. IMU measurements and features are selectively marginalized from the sliding window. Meanwhile, the measurements corresponding to marginalized states are converted into a prior. $\mathbf{r}_B(\mathbf{Z}_{b_i b_{i+1}}, \chi)$ is the residual for IMU measurement, and \mathcal{B} is the set of all IMU measurements in the sliding window. $\mathbf{r}_{\mathcal{P}}(\mathbf{Z}_{P_j}^{C_i}, \chi)$ and $\mathbf{r}_{\mathcal{L}}(\mathbf{Z}_{L_l}^{C_i}, \chi)$ are respective residuals for visual measurements of point features and structural line features. \mathcal{P} and \mathcal{L} are respective sets of point features and structural line features observed by keyframes. $\mathbf{r}_{\mathcal{C}}(\mathbf{Z}_{P_j}^{C_i}, \chi)$ is the residual for the spatial distance between the point and the line it belongs to. \mathcal{C} is the set of points and lines. ρ is the robust kernel function used to suppress outliers. The Ceres solver [33] is used to solve this nonlinear optimization problem.

The residual terms related to IMU measurements and visual measurements of point features are established with methods similar to VINS [8]. Therefore, in the following sections, we only present the details of residuals related to structural line features.

6.2. Structural line measurement residual

For the l th structural line, according to the inverse depth representation $\mathbf{L}_l = [\theta_l, \rho_l]^T$, the nonzero parameters $[a_l, b_l]^T$ of the intersection point $\mathbf{P}_{L_l}^M$ is obtained by

$$\begin{bmatrix} a_l \\ b_l \end{bmatrix} = \begin{bmatrix} \cos\theta_l / \rho_l \\ \sin\theta_l / \rho_l \end{bmatrix} \tag{14}$$

With the parameters $[a_l, b_l]^T$ and the direction vector $\mathbf{d}_{L_l}^M$ of the structural line in the Manhattan world $\{\mathbf{M}\}$, its intersection point $\mathbf{P}_{L_l}^M$ can be directly obtained, as is described in Section 3. Then, the corresponding re-projected 2D line $\mathbf{l}_l^{C_i} = (l_1, l_2, l_3)^T$ on the homogeneous coordinate plane of the i th camera frame $\{\mathbf{C}_i\}$ is obtained by (6). The structural line measurement residual is defined as the re-projected error, that is, the distance between the endpoints of $\mathbf{Z}_{L_l}^{C_i}$ and the re-projected 2D line $\mathbf{l}_l^{C_i}$. The residual $r_{\mathcal{L}}(\mathbf{Z}_{L_l}^{C_i}, \chi)$ is given by

$$r_{\mathcal{L}}(\mathbf{Z}_{L_l}^{C_i}, \chi) = \begin{bmatrix} s^T \mathbf{l}_l^{C_i} / \sqrt{l_1^2 + l_2^2} \\ \mathbf{e}^T \mathbf{l}_l^{C_i} / \sqrt{l_1^2 + l_2^2} \end{bmatrix} \tag{15}$$

where $\mathbf{s} = (s_1, s_2, 1)^T$, $\mathbf{e} = (e_1, e_2, 1)^T$ are the coordinates of two endpoints on the homogeneous coordinate plane of $\{\mathbf{C}_i\}$.

For this residual term, the state variables to be optimized include the IMU state \mathbf{x}_i , the rotation matrix \mathbf{R}_{WM} , and \mathbf{L}_l . The corresponding Jacobian matrices can be obtained by the chain rule:

$$J_L = \frac{\partial r_{\mathcal{L}}}{\partial \mathbf{l}_l^{C_i}} \frac{\partial \mathbf{l}_l^{C_i}}{\partial (\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})} \begin{bmatrix} \frac{\partial (\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})}{\partial \mathbf{x}_i} & \frac{\partial (\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})}{\partial \mathbf{R}_{WM}} & \frac{\partial (\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})}{\partial \mathbf{L}_l} \end{bmatrix}. \tag{16}$$

with

$$\begin{aligned} \frac{\partial r_{\mathcal{L}}}{\partial \mathbf{l}_l^{C_i}} &= \begin{bmatrix} \frac{s_1}{(l_1^2 + l_2^2)^{\frac{1}{2}}} + \frac{-l_1 s^T \mathbf{l}_l^{C_i}}{(l_1^2 + l_2^2)^{\frac{3}{2}}} \frac{s_2}{(l_1^2 + l_2^2)^{\frac{1}{2}}} + \frac{-l_2 s^T \mathbf{l}_l^{C_i}}{(l_1^2 + l_2^2)^{\frac{3}{2}}} \frac{1}{(l_1^2 + l_2^2)^{\frac{1}{2}}} \\ \frac{e_1}{(l_1^2 + l_2^2)^{\frac{1}{2}}} + \frac{-l_1 \mathbf{e}^T \mathbf{l}_l^{C_i}}{(l_1^2 + l_2^2)^{\frac{3}{2}}} \frac{e_2}{(l_1^2 + l_2^2)^{\frac{1}{2}}} + \frac{-l_2 \mathbf{e}^T \mathbf{l}_l^{C_i}}{(l_1^2 + l_2^2)^{\frac{3}{2}}} \frac{1}{(l_1^2 + l_2^2)^{\frac{1}{2}}} \end{bmatrix}_{2 \times 3} \\ \frac{\partial \mathbf{l}_l^{C_i}}{\partial (\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})} &= \begin{bmatrix} 0 & \frac{1}{p_3} & \frac{-p_1}{p_3^2} & 0 & \frac{-1}{d_3} & \frac{d_1}{d_3^2} \\ \frac{-1}{p_3} & 0 & \frac{p_1}{p_3^2} & \frac{1}{d_3} & 0 & \frac{-d_1}{d_3^2} \\ \frac{d_2}{p_3} & \frac{-d_1}{p_3} & \frac{-d_2 p_1 + d_1 p_2}{p_3^2} & \frac{-p_2}{d_3} & \frac{p_1}{d_3} & \frac{p_2 d_1 + p_1 d_2}{d_3^2} \end{bmatrix}_{3 \times 6}, \tag{17} \end{aligned}$$

where $\mathbf{P}_{L_l}^{C_i} = (p_1, p_2, p_3)^T$ and $\mathbf{d}_{L_l}^{C_i} = (d_1, d_2, d_3)^T$ are intersection point and direction vector of the l th structural line in the i th camera frame $\{\mathbf{C}_i\}$, respectively, which are obtained by (1) and (2).

The Jacobian matrices of $(\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})$ with respect to \mathbf{x}_i , \mathbf{R}_{WM} , and \mathbf{L}_l are defined as follows:

$$\begin{aligned} \frac{\partial (\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})}{\partial \mathbf{x}_i} &= \begin{bmatrix} \mathbf{R}_{IC}^{-1} [\mathbf{R}_{Wl_i}^{-1} (\mathbf{R}_{WM} \mathbf{P}_{L_l}^M - \mathbf{t}_{Wl_i})]^\wedge & -\mathbf{R}_{IC}^{-1} \mathbf{R}_{Wl_i}^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{R}_{IC}^{-1} [\mathbf{R}_{Wl_i}^{-1} (\mathbf{R}_{WM} \mathbf{d}_{L_l}^M)]^\wedge & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}_{6 \times 15} \\ \frac{\partial (\mathbf{P}_{L_l}^{C_i}, \mathbf{d}_{L_l}^{C_i})}{\partial \mathbf{R}_{WM}} &= \begin{bmatrix} -\mathbf{R}_{IC}^{-1} \mathbf{R}_{Wl_i}^{-1} \mathbf{R}_{WM} [\mathbf{P}_{L_l}^M]^\wedge \\ -\mathbf{R}_{IC}^{-1} \mathbf{R}_{Wl_i}^{-1} \mathbf{R}_{WM} [\mathbf{d}_{L_l}^M]^\wedge \end{bmatrix}_{6 \times 3} \end{aligned}$$

$$\frac{\partial(\mathbf{P}_{L_i}^{C_i}, d_{L_i}^{C_i})}{\partial L_i} = \begin{bmatrix} \mathbf{R}_{IC}^{-1} \mathbf{R}_{W_{L_i}}^{-1} \mathbf{R}_{WM} \\ \mathbf{0} \end{bmatrix}_{6 \times 3} \begin{bmatrix} -\sin \theta_l / \rho_l & -\cos \theta_l / \rho_l^2 \\ \cos \theta_l / \rho_l & -\sin \theta_l / \rho_l^2 \\ 0 & 0 \end{bmatrix}_{3 \times 2}, \tag{18}$$

where $[\cdot]^\wedge$ represents the skew-symmetric matrix of a three-dimension vector.

The covariance matrix $\Sigma_{L_i}^{C_i}$ used to normalize the structural line measurement residuals is defined as

$$\Sigma_{L_i}^{C_i} = \begin{bmatrix} \sigma_{L_i}^2 & 0 \\ 0 & \sigma_{L_i}^2 \end{bmatrix}_{2 \times 2} \tag{19}$$

where σ_{L_i} is set by assuming that the measurement noise of endpoints is 1 to 2 pixels.

6.3. Distance residual between points and lines

When the j th point feature has been identified as belonging to the l th structural line feature, the distance residual between them is given by

$$r_c(\mathbf{Z}_{P_j}^{C_i}, \chi) = \Pi(\mathbf{P}_{L_i}^M) - \Pi(\mathbf{P}_{P_j}^M) \tag{20}$$

$$\mathbf{P}_{P_j}^M = \mathbf{R}_{WM}^{-1} \left(\mathbf{R}_{W_{L_i}} \left(\mathbf{R}_{IC} \mathbf{Z}_{P_j}^{C_i} \frac{1}{\lambda_j} + \mathbf{t}_{IC} \right) + \mathbf{t}_{W_{L_i}} \right) \tag{21}$$

$$\Pi(\mathbf{P}_{L_i}^M) = \begin{bmatrix} a_l \\ b_l \end{bmatrix} = \begin{bmatrix} \cos \theta_l / \rho_l \\ \sin \theta_l / \rho_l \end{bmatrix} \tag{22}$$

where $\mathbf{Z}_{P_j}^{C_i}$ is the image observation of the j th point feature in the i th camera frame. $\mathbf{P}_{P_j}^M$ is the spatial coordinate of the j th point feature in the Manhattan world frame $\{\mathbf{M}\}$. $\Pi(\mathbf{P}_{P_j}^M)$ represents projecting $\mathbf{P}_{P_j}^M$ onto the plane of $\{\mathbf{M}\}$ perpendicular to the l th structural line and then resize the 3D coordinates to the 2D ones by removing zero-value components. Similarly, $\Pi(\mathbf{P}_{L_i}^M)$ represents resizing the 3D coordinates of $\mathbf{P}_{L_i}^M$ to 2D ones.

For this residual term, the state variables to be optimized are $[\mathbf{x}_i \ \mathbf{R}_{WM} \ L_l \ \lambda_j]$. The related Jacobian matrices are defined as follows:

$$\mathbf{J}_c = \frac{\partial r_c}{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)} \begin{bmatrix} \frac{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)}{\partial \mathbf{x}_i} & \frac{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)}{\partial \mathbf{R}_{WM}} & \frac{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)}{\partial L_l} & \frac{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)}{\partial \lambda_j} \end{bmatrix} \tag{23}$$

with

$$\begin{aligned} \frac{\partial r_c}{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)} &= \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \end{bmatrix}_{2 \times 6} \\ \frac{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)}{\partial \mathbf{x}_i} &= \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{R}_{WM}^{-1} \mathbf{R}_{W_{L_i}} \left[\mathbf{R}_{IC} (\mathbf{Z}_{P_j}^{C_i} / \lambda_j) + \mathbf{t}_{IC} \right]^\wedge \mathbf{R}_{WM}^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}_{6 \times 15} \\ \frac{\partial (\mathbf{P}_{L_i}^M, \mathbf{P}_{P_j}^M)}{\partial \mathbf{R}_{WM}} &= \begin{bmatrix} \mathbf{0} \\ \left[\mathbf{R}_{WM}^{-1} \mathbf{R}_{W_{L_i}} \left(\mathbf{R}_{IC} (\mathbf{Z}_{P_j}^{C_i} / \lambda_j) + \mathbf{t}_{IC} \right) + \mathbf{t}_{W_{L_i}} \right]^\wedge \end{bmatrix}_{6 \times 3} \end{aligned}$$

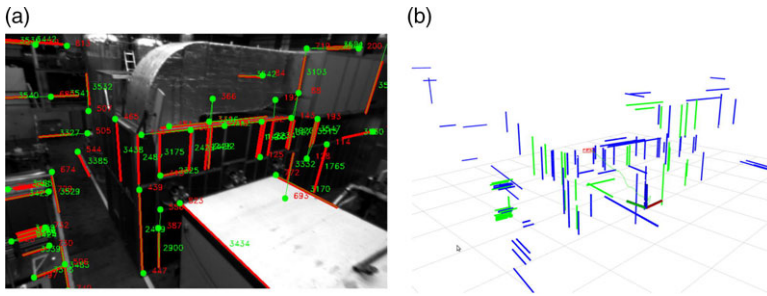


Figure 8. (a) Identified points and lines they belong to in one image. (b) The corresponding spatial structural lines to be optimized in the sliding window. The blue lines are new landmarks added to the map, and the green lines are existing ones.

$$\frac{\partial \left(P_{L_i}^M, P_{P_j}^M \right)}{\partial L_i} = \begin{bmatrix} \mathbf{0} \\ -\sin \theta_i / \rho_i & -\cos \theta_i / \rho_i^2 \\ \cos \theta_i / \rho_i & -\sin \theta_i / \rho_i^2 \\ 0 & 0 \end{bmatrix}_{6 \times 2}$$

$$\frac{\partial \left(P_{L_i}^M, P_{P_j}^M \right)}{\partial \lambda_j} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{R}_{WM}^{-1} \mathbf{R}_{WLi} \mathbf{R}_{IC} \mathbf{Z}_{P_j}^{Ci} / \lambda_j^2 \end{bmatrix}_{6 \times 1}. \tag{24}$$

Similar to the covariance matrix of structural lines measurement, the covariance matrix $\Sigma_{L_i}^{P_j}$ used to normalize the distance residual is defined as a 2×2 diagonal matrix by assuming that the spatial distance error is about 0.1–0.2 m.

7. Experimental results

To evaluate the performance of SLC-VIO, we first tested it on Euroc [34] data sets and TUM VI [35] data sets, and then conducted a real-world experiment using our devices. Two state-of-the-art VIO methods, VINS [8] and PL-VIO [14], were also implemented with their open-source code for comparison purposes. VINS [7] is a typical VIO system that only uses point features as visual measurement. The stereo version VINS-Fusion and monocular version VINS-Mono were adopted for comparative experiments, and the loop closing was disabled to evaluate the odometry performance only. PL-VIO uses both point features and line features as visual measurements. It uses Plücker coordinates to describe spatial lines and does not consider structural regularity and correlation between points and lines. All experiments were conducted on a computer with an AMD Ryzen Core 3600 CPU (@ 3.6GHz) and 16GB RAM.

7.1. Tests on Euroc data sets

The Euroc data sets consist of stereo images (frame rate: 20 FPS) and synchronized IMU measurements (sample rate: 200 Hz) [34]. They were collected by the visual-inertial sensor mounted on a micro-aerial vehicle (MAV) flying in a machine hall. As shown in Figs. 8 and 10, the machine hall is a typical structured environment with plenty of structural lines. More importantly, some scenes with weak illumination are also included in the data sets, which may challenge the positioning accuracy of VIO systems. Besides, the Euroc data sets also provide the ground truth trajectories. At the beginning of each sequence, the UAV is on a wooden frame. Due to the lack of structural lines in three orthogonal directions, it is difficult for SLC-VIO to detect a Manhattan world during initialization. Therefore, the beginning of each sequence is skipped until the machine hall can be observed.

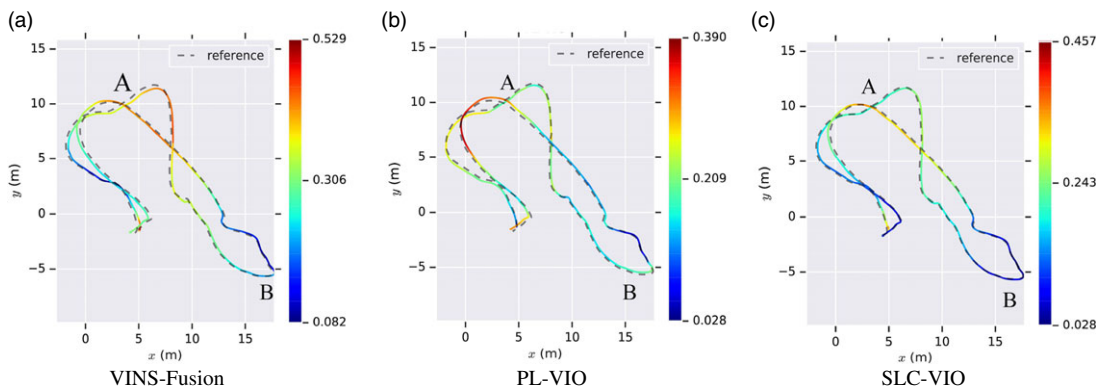


Figure 9. Estimated trajectories by VINS-Fusion, PL-VIO, and our SLC-VIO.

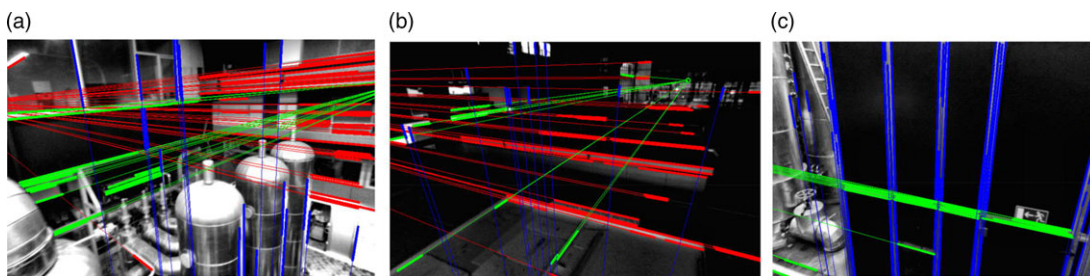


Figure 10. Illustration of scenes in Euroc data sets, where most line features are aligned with orthogonal directions. The blue lines are identified as vertical lines. The red and green lines are identified as horizontal lines.

To separately demonstrate the effects of structural lines and PPBL, we first tested SL-VIO (no PPBL) that only utilizes structural lines. Then, SLC-VIO that utilizes structural lines and PPBL was tested. The default parameters provided by authors in VINS-Fusion [7] and PL-VIO were used. The absolute pose error (APE), that is, the position difference between the ground truth and the estimated ones, was used to evaluate the positioning accuracy.

Table I presents the root-mean-square error (RMSE) of APE on five sequences. It shows that the proposed SLC-VIO achieves the best performance on almost all sequences, except for MH_01_easy. The results that SL-VIO (no PPBL) is better than VINS-Fusion, VINS-Mono, and PL-VIO on most sequences illustrate the advantage of using structural lines. Besides, by comparing SLC-VIO with SL-VIO(no PPBL), it can be found that the positioning accuracy of VIO system is further improved by using PPBL. It is also found that, compared to VINS-Mono, VINS-Fusion does not show a distinct advantage in accuracy since VINS-Fusion aims to improve the robustness and applicability. On the last two sequences, VINS-Fusion achieves better performance than the monocular version. The reason could be that VINS-Mono skipped some frames to ensure real-time performance and failed to match or track sufficient point features in cases of poor illumination.

Figure 8(a) demonstrates the points belonging to line features on MH_02_easy. Figure 8(b) shows the corresponding 3D structural lines in the reconstructed map. In Fig. 8(b), most of the structural lines marked in blue are constrained by distance residuals. It can be found that there are sufficient points and lines they belong to in the environment.

To visually compare the accuracy of VINS-Fusion, PL-VIO, and SLC-VIO, the estimated trajectories of the three methods on MH_04_difficult sequence are presented in Fig. 9. The amplitude of errors is denoted by colors. It can be seen that the trajectory estimated by SLC-VIO is the closest to the ground truth, especially in area A characterized by weak illumination (see in Fig. 10(b)) and area B characterized by sparse structural lines (see in Fig. 10(c)). The results demonstrate that the utilization of structural

Table I. RMSE on Euroc data sets (unit: meters).

Sequence	VINS-Fusion	VINS-Mono	PL-VIO	SL-VIO (no PPBL)	SLC-VIO
MH_01_easy	0.194	0.206	0.095	0.130	0.120
MH_02_easy	0.182	0.172	0.207	0.238	0.166
MH_03_medium	0.130	0.140	0.106	0.103	0.089
MH_04_difficult	0.306	0.352	0.223	0.215	0.182
MH_05_difficult	0.203	0.346	0.256	0.229	0.179

Table II. Average time consumption on each frame (unit: millisecond).

Sequence	VINS-Fusion	PL-VIO	SLC-VIO
MH_01_easy	56	100	72
MH_02_easy	55	101	72
MH_03_medium	56	93	69
MH_04_difficult	52	88	64
MH_05_difficult	51	98	68

Table III. The average execution time of each key operation in SLC-VIO (unit: millisecond).

Operations	Time	
Measurement processing	Detection and tracking of point features	19
	Detection and tracking of line features	66
	Identification of structural line features	<1
	Identification of points belongs to lines	<1
	Triangulation of point/line features	<1
	Merging of redundant line features	2
Sliding window optimization	57	

lines and PPBL can further improve the positioning accuracy of the VIO system in a weak-illumination environment.

The average time consumption was evaluated on Euroc data sets, and the results are shown in Table II. It can be concluded that the computation efficiency of VIN-Fusion is the highest since it only takes point features as landmarks. Whereas in PL-VIO and SLC-VIO, both point features and line features are used as landmarks, the efficiency is relatively low. Moreover, with the 2-DoF spatial structural lines, the number of variables to be optimized in SLC-VIO is less than that in PL-VIO. Therefore, the efficiency of SLC-VIO is higher than PL-VIO.

Table III presents the average execution time of each key operation in SLC-VIO. The detection and tracking of line features and sliding window optimization are the most time-consuming processes. The processing time of sliding window optimization depends on the number of features in an image and fluctuates in the range of 42–62 ms. The time consumption is ensured by limiting the maximum number of features (the maximum number of points and lines is 150 and 35, respectively) extracted from an image. Since the measurement processing and sliding window optimization run in parallel, the efficiency of SLC-VIO is mainly determined by these two processes. SLC-VIO adopts LSD and LBD algorithms to detect and track line features, which can be further accelerated by GPU. Therefore, the efficiency of this algorithm might be improved with the aid of hardware acceleration. In sliding window optimization, marginalization is another time-consuming operation due to the dense Hessian matrix. This problem can be potentially solved by discarding part of point and line features to obtain a sparse Hessian matrix. Skipping frames is another solution to ensure real-time performance. In this implementation, frames are skipped to ensure the oncoming frame is processed timely if the actual frame rate exceeds 10 Hz.

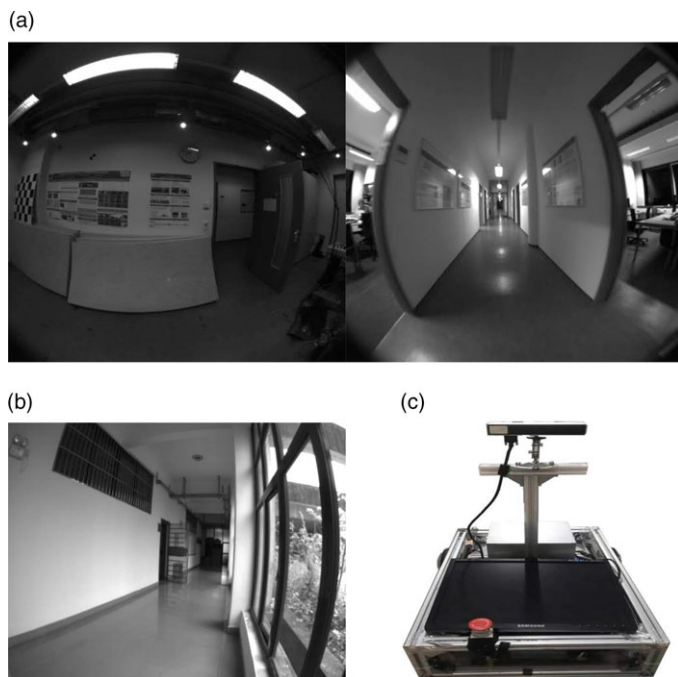


Figure 11. (a) Illustration of scenes in TUM VI data sets. (b) Illustration of the corridor in the real-world experiment. (c) The automated vehicle used in the real-world experiment.

Generally, this method can improve the efficiency of SLC-VIO with little influence on the positioning accuracy. However, the number of matched and tracked point/line features may decrease if some frames are skipped while the camera is moving at high speed, and the accuracy deteriorates in these cases.

7.2. Tests on TUM VI data sets and real-world experiment

Because the scenes in TUM VI [35] data sets (see in Fig. 11(a)) and real-world experiments (see in Fig. 11(b)) are similar low-texture corridors, the two experiments are described and analyzed together. Since there are not enough point features, such low-texture corridors are also a great challenge for visual SLAM/VIO systems.

The TUM VI data sets are collected by a handheld device and also provide the ground truth trajectories. Since the images' distortions in TUM data set are relatively large, we used an equidistant camera model [36] rather than a pinhole camera model to cope with the distortions. As shown in Fig. 11(c), the real-world experiment was carried out on an automated vehicle equipped with a visual-inertial sensor. The automated vehicle ran in the corridor at a speed of 1 m/s to collect data. Data acquisition started and ended at the same location. Unlike the public data sets, there is no ground truth for the evaluation of positioning accuracy in the real-world experiment. However, since the actual starting point and endpoint are the same, the positioning error can be determined by the distance between the starting point and the endpoint of the estimated trajectory.

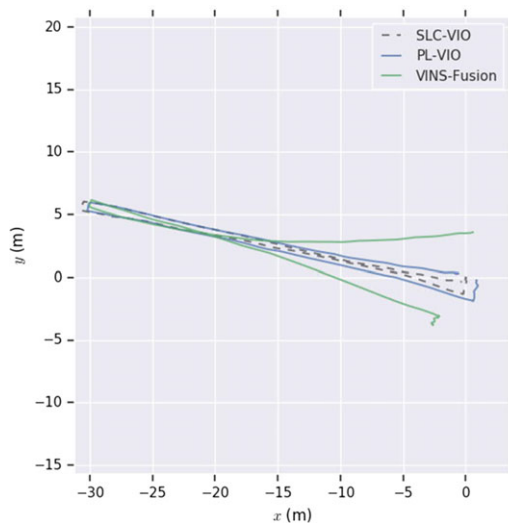
Table IV presents the RMSE of APE tested on TUM VI data sets. Figure 12 and Table V show the estimated trajectories and positioning errors of the three VIO systems in a real-world experiment, respectively. It can be seen that in these two experiments, due to the lack of point features, the positioning error of VINS-Fusion is significantly larger than that of PL-VIO and SLC-VIO. This result demonstrates the advantages of using line features for VIO system in low-texture environments. The result that SLC-VIO performed better than PL-VIO indicates that using structural line features as landmarks and considering the PPBL can further improve the accuracy of VIO system in structured environments, especially in environments with low texture.

Table IV. RMSE on TUM VI data sets (unit: meters).

	VINS-Fusion	PL-VIO	SLC-VIO
RMSE	0.306	0.235	0.219

Table V. Positioning error in real-world experiments.

Methods	Total length (m)	Positioning error (m)	Percentage error (%)
VINS-Fusion	42.347	10.425	24.63
PL-VIO	51.76	1.068	2.06
SLC-VIO	49.762	0.539	1.08

**Figure 12.** The estimated trajectories in the real-world experiment.

Figures 13 and 14, respectively, show the reconstructed 3D map by VINS-Fusion, PL-VIO, and SLC-VIO in the tests of the TUM VI data sets and real-world experiments. It can be seen that, compared with the map composed of only sparse point features, the map composed of line features can better reflect the environment's geometric structure information. And in Fig. 14(a), there is an apparent drift in the 3D map obtained by VINS-Fusion. It demonstrates that the mapping performance of the VIO system using only point features will deteriorate in a low-texture environment. In contrast, as shown in Fig. 14(b)–(c), in the reconstructed 3D maps utilizing line features, the drift is significantly reduced. Besides, as shown in Figs. 13(b) and 14(b), there are many disordered line features in the 3D map obtained by PL-VIO. In contrast, as shown in Figs. 13(c) and 14(c), the 3D map obtained by SLC-VIO more correctly reflects the geometric structure of scenes. The reason is that after using the prior environmental geometric information, the 2-DoF structural line in SLC-VIO has better convergence than the 4-DoF straight line represented by Plücker coordinates in PL-VIO during optimization. This result illustrates the advantages of using structural lines to reconstruct a 3D map.

8. Conclusion

This paper presents the SLC-VIO system, which is a stereo VIO using both point features and structural line features as landmarks and considering the property of PPBL. The man-made structure environment is modeled as a Manhattan world, and then 2-DoF spatial structure lines are defined in it. By adding

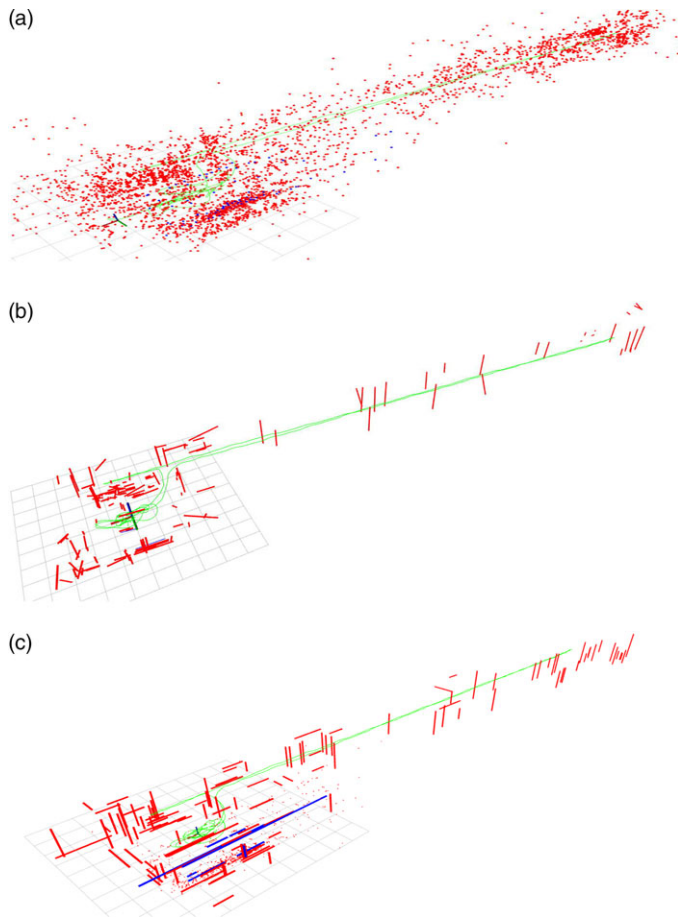


Figure 13. Reconstructed 3D maps by VINS-Fusion (a), PL-VIO (b), and SLC-VIO (c) on TUM VI data sets. The blue lines in (c) are new landmarks added to the map, and the red points or lines are the old landmarks in the map. The green line is the trajectory of camera.

the image observations of structural lines to visual measurements and establishing distance-residual constraints between points and lines in the reconstructed 3D map, the proposed system makes full use of the prior geometric information of structured environments and thereby achieves better positioning accuracy.

The proposed system was tested on public data sets and in the real-world environment and compared with the state-of-the-art VIO methods, including VINS [8] and PL-VIO [14]. The results illustrate that taking structural line features as landmarks and considering the PPBL can significantly reduce the drift in the reconstructed 3D maps and improve the positioning accuracy, especially in low-texture or poor illumination environments. However, in environments where it is hard to find line features, the proposed system degenerates into a VIO system that uses only point features, just like VINS [7]. In addition, it can be seen that the spatial structural line with 2-DoF has better convergence and optimization efficiency than the general spatial line with 4-DoF represented by Plücker coordinates during optimization.

Author Contributions. All authors contributed to the study conception, design, and implementation. Material preparation, data collection, and analysis were performed by Chenchen Wei, Yanfeng Tang, and Zhi Huang. The original draft of the manuscript was written by Chenchen Wei and reviewed and edited by Lingfang Yang and Zhi Huang. All authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

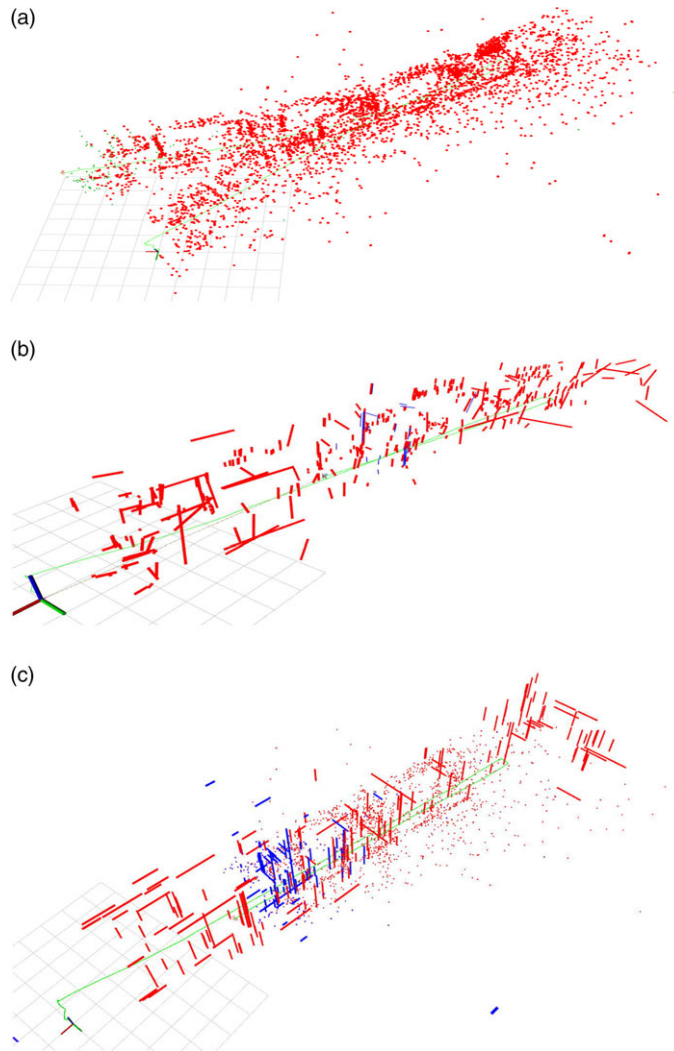


Figure 14. Reconstructed 3D maps by VINS-Fusion (a), PL-VIO (b), and SLC-VIO (c) in the real-world experiment. The blue lines in (c) are new landmarks added to the map, and the red points or lines are the old landmarks in the map. The green line is the trajectory of camera.

Financial Support. This work was supported by the Natural Science Foundation of Hunan province (Grant numbers [2018JJ3062]).

Ethical Considerations. None.

Conflicts of Interest. The authors declare none.

References

- [1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, Leonard and J. John, “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” *IEEE Trans. Robot.* **32**(6), 1309–1332 (2016). doi: [10.1109/TRO.2016.2624754](https://doi.org/10.1109/TRO.2016.2624754).
- [2] J. Artieda, José M. Sebastian, Pascual Campoy, Juan F. Correa, Iván F. Mondragón, Carol Martínez and Miguel Olivares, “Visual 3D SLAM from UAVs,” *J. Intell. Robot. Syst. Theory Appl.* **55**(4–5), 299–321 (2009). doi: [10.1007/s10846-008-9304-8](https://doi.org/10.1007/s10846-008-9304-8).

- [3] M. Yekkehfallah, M. Yang, Z. Cai, L. Li and C. Wang, "Accurate 3D localization using RGB-TOF camera and IMU for industrial mobile robots," *Robotica* **39**(10), 1816–1833 (2021). doi: [10.1017/S0263574720001526](https://doi.org/10.1017/S0263574720001526).
- [4] G. Klein and D. Murray, "Parallel Tracking and Mapping on a Camera Phone," *Science & Technology Proceedings - IEEE 2009 International Symposium on Mixed and Augmented Reality, ISMAR 2009* (2009) pp. 83–86. doi: [10.1109/ISMAR.2009.5336495](https://doi.org/10.1109/ISMAR.2009.5336495).
- [5] C. Forster, M. Pizzoli and D. Scaramuzza, "SVO: Fast Semi-Direct Monocular Visual Odometry," *IEEE International Conference on Robotics and Automation (ICRA)* (2014) pp. 15–22.
- [6] R. Mur-Artal, J. M. M. Montiel and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.* **31**(5), 1147–1163 (2015). doi: [10.1109/TRO.2015.2463671](https://doi.org/10.1109/TRO.2015.2463671).
- [7] G. Nützi, S. Weiss, D. Scaramuzza and R. Siegwart, "Fusion of IMU and vision for absolute scale estimation in monocular SLAM," *J. Intell. Robot. Syst. Theory Appl.* **61**(1–4), 287–299 (2011). doi: [10.1007/s10846-010-9490-z](https://doi.org/10.1007/s10846-010-9490-z).
- [8] T. Qin, P. Li and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.* **34**(4), 1004–1020 (2018). doi: [10.1109/TRO.2018.2853729](https://doi.org/10.1109/TRO.2018.2853729).
- [9] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *Int. J. Rob. Res.* **34**(3), 314–334 (2015). doi: [10.1177/0278364914554813](https://doi.org/10.1177/0278364914554813).
- [10] J. M. Coughlan and A. L. Yuille, "Manhattan World: Compass direction from a single image by Bayesian inference," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2 (1999) pp. 941–947. doi: [10.1109/iccv.1999.790349](https://doi.org/10.1109/iccv.1999.790349).
- [11] A. Pumarola, A. Vakhitov, A. Agudo, A. Sanfeliu and F. Moreno-Noguer, "PL-SLAM: Real-Time Monocular Visual SLAM with Points and Lines," *2017 IEEE International Conference on Robotics and Automation (ICRA)* (2017) pp. 4503–4508. doi: [10.1109/ICRA.2017.7989522](https://doi.org/10.1109/ICRA.2017.7989522).
- [12] R. Gomez-Ojeda, F. A. Moreno, D. Zuiga-Nol, D. Scaramuzza and J. Gonzalez-Jimenez, "PL-SLAM: A stereo SLAM system through the combination of points and line segments," *IEEE Trans. Robot.* **35**(3), 734–746 (2019). doi: [10.1109/TRO.2019.2899783](https://doi.org/10.1109/TRO.2019.2899783).
- [13] L. Zhao, S. Huang, L. Yan and G. Dissanayake, "A new feature parametrization for monocular SLAM using line features," *Robotica* **33**(3), 513–536 (2015). doi: [10.1017/S026357471400040X](https://doi.org/10.1017/S026357471400040X).
- [14] Y. He, J. Zhao, Y. Guo, W. He and K. Yuan, "PL-VIO: Tightly-coupled monocular visual-inertial odometry using point and line features," *Sensors (Switzerland)* **18**(4), 1–25 (2018). doi: [10.3390/s18041159](https://doi.org/10.3390/s18041159).
- [15] J. A. Castellanos and J. D. Tardós, *Mobile Robot Localization and Map Building*, vol. **39**, no. 6 (Springer, Berlin, 1999) pp. 275–284. doi: [10.1007/978-1-4615-4405-0](https://doi.org/10.1007/978-1-4615-4405-0).
- [16] A. P. Gee and W. Mayol-Cuevas, "Real-Time Model-Based SLAM Using Line Segments," *Lecture Notes in Computer Science*, vol. **4292** (2006) pp. 354–363. doi: [10.1007/11919629_37](https://doi.org/10.1007/11919629_37).
- [17] R. G. von Gioi, J. Jakubowicz, J.-M. Morel and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(4), 722–732 (2010). doi: [10.1109/TPAMI.2008.300](https://doi.org/10.1109/TPAMI.2008.300).
- [18] L. Zhang and R. Koch, "An efficient and robust line segment matching approach based on LBD descriptor and pairwise geometric consistency," *J. Vis. Commun. Image Represent.* **24**(7), 794–805 (2013). doi: [10.1016/j.jvcir.2013.05.006](https://doi.org/10.1016/j.jvcir.2013.05.006).
- [19] A. Bartoli and P. Sturm, "The 3D line motion matrix and alignment of line reconstructions," *Int. J. Comput. Vis.* **57**(3), 159–178 (2004). doi: [10.1023/B:VISI.0000013092.07433.82](https://doi.org/10.1023/B:VISI.0000013092.07433.82).
- [20] F. Zheng, G. Tsai, Z. Zhang, S. Liu, C. C. Chu and H. Hu, "Trifo-VIO: Robust and Efficient Stereo Visual Inertial Odometry Using Points and Lines," *IEEE International Conference on Intelligent Robots and Systems* (2018) pp. 3686–3693. doi: [10.1109/IROS.2018.8594354](https://doi.org/10.1109/IROS.2018.8594354).
- [21] Y. H. Lee, C. Nam, K. Y. Lee, Y. S. Li, S. Y. Yeon and N. L. Doh, "VPass: Algorithmic Compass using Vanishing Points in Indoor Environments," *IEEE-RSJ International Conference on Intelligent Robots and System* (2009) pp. 936–941. doi: [10.1109/IROS.2009.5354508](https://doi.org/10.1109/IROS.2009.5354508).
- [22] F. Camposeco and M. Pollefeys, "Using Vanishing Points to Improve Visual-Inertial Odometry," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015 (2015) pp. 5219–5225. doi: [10.1109/ICRA.2015.7139926](https://doi.org/10.1109/ICRA.2015.7139926).
- [23] S. Kim and S. Y. Oh, "SLAM in indoor environments using omni-directional vertical and horizontal line features," *J. Intell. Robot. Syst. Theory Appl.* **51**(1), 31–43 (2008). doi: [10.1007/s10846-007-9179-0](https://doi.org/10.1007/s10846-007-9179-0).
- [24] G. Zhang, D. H. Kang and I. H. Suh, "Loop Closure Through Vanishing Points in a Line-Based Monocular SLAM," *Proceedings - IEEE International Conference on Robotics and Automation* (2012) pp. 4565–4570. doi: [10.1109/ICRA.2012.6224759](https://doi.org/10.1109/ICRA.2012.6224759).
- [25] H. Zhou, D. Zou, L. Pei, R. Ying, P. Liu and W. Yu, "StructSLAM: Visual SLAM with building structure lines," *IEEE Trans. Veh. Technol.* **64**(4), 1364–1375 (2015). doi: [10.1109/TVT.2015.2388780](https://doi.org/10.1109/TVT.2015.2388780).
- [26] J.-P. Tardif, "Non-Iterative Approach for Fast and Accurate Vanishing Point Detection," *IEEE International Conference on Computer Vision (ICCV)* (2009) pp. 1250–1257. doi: [10.1109/ICCV.2009.5459328](https://doi.org/10.1109/ICCV.2009.5459328).
- [27] D. Zou, Y. Wu, L. Pei, H. Ling and W. Yu, "StructVIO: Visual-inertial odometry with structural regularity of man-made environments," *IEEE Trans. Robot.* (2019). doi: [10.1109/TRO.2019.2915140](https://doi.org/10.1109/TRO.2019.2915140).
- [28] Z. Huang, B. Fan and X. Song, "Robust lane detection and tracking using multiple visual cues under stochastic lane shape conditions," *J. Electron. Imaging* **27**(02), 1 (2018). doi: [10.1117/1.jei.27.2.023025](https://doi.org/10.1117/1.jei.27.2.023025).
- [29] M. Nieto and L. Salgado, "Non-Linear Optimization for Robust Estimation of Vanishing Points," *Proceedings - International Conference on Image Processing. ICIP* (2010) pp. 1885–1888. doi: [10.1109/ICIP.2010.5652381](https://doi.org/10.1109/ICIP.2010.5652381).
- [30] A. M. Andrew, *Multiple View Geometry in Computer Vision*, vol. **30**, no. 9–10 (Cambridge University Press, Cambridge, 2001).

- [31] V. Lepetit, F. Moreno-Noguer and P. Fua, “EPnP: An accurate $O(n)$ solution to the PnP problem,” *Int. J. Comput. Vis.* **81**(2), 155–166 (2009). doi: [10.1007/s11263-008-0152-6](https://doi.org/10.1007/s11263-008-0152-6).
- [32] J. Civera, A. J. Davison and J. M. M. Montiel, “Inverse depth parametrization for monocular SLAM,” *IEEE Trans. Robot.* **24**(5), 932–945 (2008). doi: [10.1109/TRO.2008.2003276](https://doi.org/10.1109/TRO.2008.2003276).
- [33] A. Sameer, M. Keir, “Ceres Solver,” p. 2018 (2010).
- [34] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. Achtelik and R. Siegwart, “The EuRoC micro aerial vehicle datasets,” *Int. J. Rob. Res.* **35**(10), 1157–1163 (2016). doi: [10.1177/0278364915620033](https://doi.org/10.1177/0278364915620033).
- [35] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stuckler and D. Cremers, “The TUM VI Benchmark for Evaluating Visual-Inertial Odometry,” (2018). doi: [10.1109/IROS.2018.8593419](https://doi.org/10.1109/IROS.2018.8593419).
- [36] F. Huang, Y. Wang, X. Shen, C. Lin and Y. Chen, “Method for calibrating the fisheye distortion center,” *Appl. Opt.* **51**(34), 8169–8176 (2012). doi: [10.1364/AO.51.008169](https://doi.org/10.1364/AO.51.008169).

Author Biography

Chenchen Wei is a graduate student in the College of Mechanical and Vehicle Engineering, Hunan University, China. His research interests include visual SLAM and sensor fusion.

Yanfeng Tang is a graduate student in the College of Mechanical and Vehicle Engineering, Hunan University, China. His research interests include visual SLAM and control of autonomous vehicles.

Lingfang Yang received the Ph.D. degree in civil engineering from Hunan University, Changsha, China. She is an assistant professor with the College of Civil Engineering, Hunan University. Her research interests include intelligent transportation, deep learning, and data mining.

Zhi Huang received the Ph.D. degree in mechanical engineering from Hunan University, Changsha, China. He is an Associate Professor with the College of Mechanical and Vehicle Engineering, Hunan University. His research interests include advanced assistant driving, active safety, electric vehicle, embedded system, and vehicle dynamics control.