UV-SLAM: Unconstrained Line-Based SLAM Using Vanishing Points for Structural Mapping

Hyunjun Lim, *Student Member, IEEE*, Jinwoo Jeon, *Student Member, IEEE*,
and Hyun Myung, *Senior Member, IEEE*

***Abstract*—In feature-based simultaneous localization and mapping (SLAM), line features complement the sparsity of point features, making it possible to map the surrounding environment structure. Existing approaches utilizing line features have primarily employed a measurement model that uses line re-projection. However, the direction vectors used in the 3D line mapping process cannot be corrected because the line measurement model employs only the lines' normal vectors in the Plücker coordinate. As a result, problems like degeneracy that occur during the 3D line mapping process cannot be solved. To tackle the problem, this letter presents a UV-SLAM, which is an unconstrained line-based SLAM using vanishing points for structural mapping. This letter focuses on using structural regularities without any constraints, such as the Manhattan world assumption. For this, we use the vanishing points that can be obtained from the line features. The difference between the vanishing point observation calculated through line features in the image and the vanishing point estimation calculated through the direction vector is defined as a residual and added to the cost function of optimization-based SLAM. Furthermore, through Fisher information matrix rank analysis, we prove that vanishing point measurements guarantee a unique mapping solution. Finally, we demonstrate that the localization accuracy and mapping quality are improved compared to the state-of-the-art algorithms using public datasets.**

***Index Terms*—Localization, mapping, SLAM, visual-inertial SLAM.**

I. Introduction

THE feature-based visual simultaneous localization and mapping (SLAM) has been primarily developed based on point features because a point is the smallest unit that can express the characteristics of an image and has an advantage in low computation environment [1]. In addition, point features have been used for localization because they are easy to track. However, point features have several drawbacks [2]. First, point features are not robust in low-texture environments such as hallways. Moreover, they are weak against illumination change. Finally, point features are sparse, making it difficult to visualize the surrounding environment with a 3D map.

To supplement the point features, line-based methods have been proposed. Line features can additionally be used in environments with low textures, such as corridors. In addition, because the line consists of several points, there is a high probability that the characteristics will be maintained even when an illumination change occurs [5], [6]. Finally, because line features have structural regularities, the surrounding environment can be easily identified through 3D mapping [7].

With the above advantages, many studies have been conducted to apply line features to visual SLAM. First, a method using the Plücker coordinate and the orthonormal representation for representing 3D lines was proposed [8] to express a 3D line with a higher degree of freedom (DoF) than a 3D point. Based on this, line measurement model was defined in a similar way to the point measurement model. It re-projects a 3D line and calculates the difference from the new observed line. Most line-based algorithms adopt similar methods on existing point-based methods. Filtering-based approaches [9], [10] were developed from MSCKF [11]. Some optimization-based methods [12]–[14] exploited ORB-SLAM [15] and other approaches [2]–[4], [16] used VINS-Mono [17].

However, the above algorithms applying only line measurement model does not solve the problems in the 3D mapping of lines. Among the difficulties, a degeneracy problem occurs in the 3D mapping of line features. The degeneracy refers to a phenomenon in which 3D features cannot be uniquely determined through the triangulation of features [18]. When the observed 2D point is close to the epipole, a point feature cannot be determined as a single 3D point because the 3D point exists on the baseline. Similarly, when the observed 2D line passes close to the epipole, a line feature cannot be determined as a single 3D line because the 3D line exists on the epipolar plane. The mapping of line features is inaccurate because degeneracy occurs more frequently than point features do. However, because the line measurement model used in the existing algorithm employs only each line's normal vector in the Plücker coordinate, their direction vectors cannot be corrected. Fig. 1 shows the comparison of top views of 3D line mapping results between ALVIO [3], [4] and the proposed algorithm with the LiDAR point clouds overlaid in a real corridor environment. The ALVIO has poor mapping

Manuscript received September 9, 2021; accepted December 23, 2021. Date of publication January 6, 2022; date of current version January 13, 2022. This letter was recommended for publication by Associate Editor L. Zhang and Editor J. Civera upon evaluation of the reviewers' comments. This work was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) under Grant 2021-0-00230, (development of real · virtual environmental analysis based adaptive interaction technology) and in part by the Defense Challengeable Future Technology Program of Agency for Defense Development, Republic of Korea. The students are supported by the Korea Ministry of Land, Infrastructure and Transport (MOLIT) as Innovative Talent Education Program for Smart City and BK21 FOUR. *(Corresponding author: Hyun Myung.)*

Hyunjun Lim and Jinwoo Jeon are with the School of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea (e-mail: tp02134@kaist.ac.kr; zinuok@kaist.ac.kr).

Hyun Myung is with the School of Electrical Engineering, KI-AI, and KI-R, KAIST, Daejeon 34141, Republic of Korea (e-mail: hmyung@kaist.ac.kr).

Digital Object Identifier 10.1109/LRA.2022.3140816

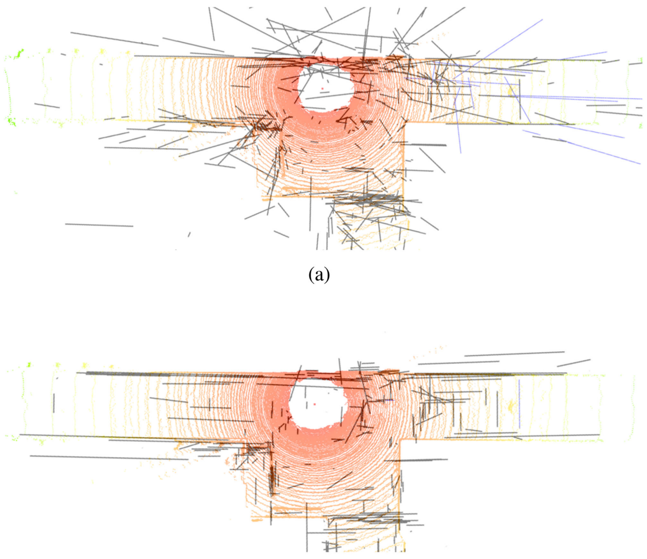


Fig. 1. Top views of 3D line mapping results with LiDAR point clouds overlaid in real corridor environment (a) ALVIO [3], [4]. (b) UV-SLAM.

result despite the lines having structural regularities as shown in Fig. 1(a).

In this letter, we propose a UV-SLAM, an algorithm that solves the above problems using the lines' structural regularities as shown in Fig. 1(b). The main contributions of this letter are as follows:

- To the best of our knowledge, the proposed UV-SLAM is the first optimization-based monocular SLAM using vanishing point measurements for structural mapping without any restriction such as camera motion and environment. In particular, our algorithm does not use the Manhattan world assumption in the process of extracting the vanishing points and using them as measurements.
- We define a novel residual term and Jacobian of the vanishing point measurements based on the most common methods of expressing 3D line features: the Plücker coordinate and the orthonormal representation.
- We prove that the proposed method guarantees the observability of 3D lines through Fisher information matrix (FIM) rank analysis. Through this, problems that occur in the 3D mapping of line features are proven to be solved.

The rest of this letter is organized as follows. Section II gives a review of related works. Section III explains the proposed method in depth. Section IV analyzes the Fisher information matrix rank to prove the validity of the proposed method. Section V provides the experimental results. Finally, Section VI concludes by summarizing our contributions and discussing future work.

II. Related Works

A. Degeneracy of Line Features

Some works looked into the degeneracy for line features [19], [20]. In addition, Yang *et al.* investigated degenerate motions using two distinct line triangulation methods [10]. Subsequently, they analyzed various features' observability and degenerate camera motions in the inertial measurement unit (IMU) aided navigation system [21]. However, when line degeneracy occurs in these investigations, all of the degenerate lines have been eliminated. As a result, there is a limitation in that information loss occurs due to the removed lines.

To handle degenerate lines, Ok *et al.* reconstructed 3D line segments using imaginary points when the lines are close to the epipolar line [22]. However, this approach can be used only when degenerate lines intersect with other lines, and it restricts applicability. Our previous work solved the degeneracy by using structural constraints in parallel conditions, investigating the fact that degenerate lines frequently occur in pure translation motions [4]. However, this method has a limitation in that it can be used only in pure translational motions. Therefore, in order to improve the quality of line mapping, a method that can be used independent of camera motion is required.

B. Line-Based SLAM With Manhattan or Atlanta World Assumption

In [23], [24], rotation matrix was estimated using the Manhattan world assumption. Based on this, decoupled methods have been proposed to estimate the translation after calculating the rotation through the vanishing points [25]–[27]. In addition, some approaches applied line features to SLAM by using the Manhattan or Atlanta world assumption [28], [29]. These methods use a novel 2-DoF line representation to exploit lines with structural regularities only. Furthermore, there is a study using 2-DoF line representation to classify structural lines and non-structural lines [30]. However, as these approaches use structural lines with dominant direction only, they are practical only in an indoor environment where the assumptions are mostly correct. Therefore, there is a need for a novel algorithm that is not restricted by assumptions.

C. Vanishing Point Measurements

Some approaches use vanishing point measurements without assumptions. In [31], parallel lines were clustered based on vanishing points. Then, residuals were constructed using the conditions that parallel lines should be in one plane and their cross product should be zero. However, accurate mapping results could not be obtained when the initial estimation was inaccurate as degeneracy occurred.

Moreover, there is a paper using vanishing points as an observation model. In [32], the residuals are defined to apply unbounded vanishing point measurements to line-based SLAM. Unfortunately, it does not provide a proof that vanishing point measurements improve the localization accuracy and line mapping results of line-based SLAM.

III. Proposed Method

A. Framework of Algorithm

The overall framework of the UV-SLAM is shown in Fig. 2. The proposed method is based on VINS-Mono [17], and the way IMU and point measurements are used is similar to it. The point features are extracted from Shi-Tomasi [33] and are tracked by KLT [34]. In addition, the IMU measurement model is defined by the pre-integration method [35]. Finally, our

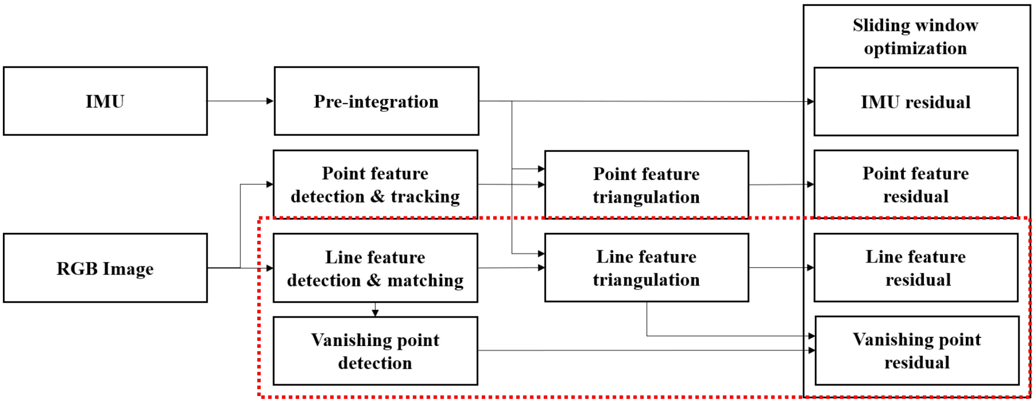

Sliding window optimization
IMU
Pre-integration
IMU residual
Point feature detection & tracking
Point feature triangulation
Point feature residual
RGB Image
Line feature detection & matching
Line feature triangulation
Line feature residual
Vanishing point detection
Vanishing point residual

Fig. 2. Block diagram illustrating the framework of UV-SLAM. The dashed box represents the newly added blocks in this letter. When an RGB image is received, detection and matching of line features are carried out. Afterward, the vanishing points are detected, and each line is clustered. After creating 3D lines through the triangulation process, the residuals of both the lines and the vanishing points are defined. Finally, localization and mapping results can be obtained through sliding window optimization.

optimization-based method employs two-way marginalization with Schur complement [36].

To add line features to the monocular visual-inertial odometry (VIO) system, the line features are extracted from line segment detector (LSD) [37] and are tracked by line binary descriptor (LBD) [38]. Whereas 3D points can be intuitively expressed in (x, y, z), 3D lines require a complicated way to express themselves. Therefore, the proposed algorithm employs a Plücker coordinate and a orthonormal representation used in [8]. The Plücker coordinate is an intuitive way to represent 3D lines, and a 3D line is represented as follows:

$$\mathbf{L}(\mathbf{n}, \mathbf{d})^\top \in \mathbb{R}^6, \tag{1}$$

where $\mathbf{n}$ and $\mathbf{d}$ represent normal and direction vectors, respectively. The Plücker coordinate is used in the triangulation and re-projection process. Whereas 3D lines are actually 4-DoF, the lines in the Plücker coordinate are 6-DoF. Therefore, over-parameterization problem occurs in the optimization process of VIO or visual SLAM. To solve this, the orthonormal representation is employed, which is a 4-DoF representation of lines. It is used in the optimization process and can be expressed as follows:

$$\mathbf{o} = [\boldsymbol{\psi}, \phi], \tag{2}$$

where $\boldsymbol{\psi}$ is the 3D line's rotation matrix in Euler angles with respect to the camera coordinate system, and ϕ is the parameter representing the minimal distance from the camera center to the line. The conversion between the Plücker coordinate and the orthonormal representation is given in [8].

In addition, the UV-SLAM can determine whether the extracted lines have structural regularities or not. For vanishing point detection, the proposed algorithm use J-linkage [39], which can find multiple instances in the presence of noise and outliers. The overall process is as follows: First, vanishing point hypotheses are created through random sampling for all lines extracted from the image. Subsequently, after merging similar ones through comparison between the hypotheses, vanishing points are calculated. Because the J-linkage can find all possible vanishing points through the hypotheses, it can find more vanishing points than other algorithms with the Manhattan world assumption.

B. State Definition

In this letter, $(\cdot)^w$, $(\cdot)^c$, and $(\cdot)^b$ represent the world coordinate, camera coordinate, and body coordinate, respectively. In addition, $(\cdot)_b^w$ reflects the coordinate transformations of a rotation matrix, quaternion, or translation from the body coordinate to the world coordinate. The state vector used in our system is as follows:

$$\begin{aligned}
\mathcal{X} &= [\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{I-1},\\
&\quad \lambda_0, \lambda_1, \ldots, \lambda_{J-1},\\
&\quad \mathbf{o}_0, \mathbf{o}_1, \ldots, \mathbf{o}_{K-1}],\\
\mathbf{x}_i &= [\mathbf{p}_{b_i}^w, \mathbf{q}_{b_i}^w, \mathbf{v}_{b_i}^w, \mathbf{b}_a, \mathbf{b}_g],\ i \in [0, I-1],\\
\mathbf{o}_k &= [\boldsymbol{\psi}_k, \phi_k],\ k \in [0, K-1],
\end{aligned} \tag{3}$$

where $\mathcal{X}$ represents the entire state, and $\mathbf{x}_i$ represents the body state in the i-th sliding window, which is made up of the following parameters: position, quaternion, velocity, and biases of the accelerometer and gyroscope. In addition, the entire state includes the inverse depths of point features, which are represented as $\lambda_j,\ j \in [0, J-1]$. In this letter, lines expressed in the orthonormal representations are newly added as $\mathbf{o}$. I, J, and K are the numbers of sliding window, point features, and line features, respectively.

C. UV-Slam

Employing defined states in (3), the entire objective for optimization is as follows:

$$\begin{aligned}
\min_{\mathcal{X}} \Bigg\{ & \| \mathbf{r}_0 - \mathbf{J}_0 \mathcal{X} \|^2 + \sum_{i \in \mathcal{B}} \| \mathbf{r}_I(\mathbf{z}_{b_{i+1}}^{b_i}, \mathcal{X}) \|_{\Sigma_{b_{i+1}}^{b_i}}^2\\
& + \sum_{(i,j) \in \mathcal{P}} \rho_p \| \mathbf{r}_p(\mathbf{z}_{p_j}^{c_i}, \mathcal{X}) \|_{\Sigma_{p_j}^{c_i}}^2 + \sum_{(i,k) \in \mathcal{L}} \rho_l \| \mathbf{r}_l(\mathbf{z}_{l_k}^{c_i}, \mathcal{X}) \|_{\Sigma_{l_k}^{c_i}}^2\\
& + \sum_{(i,k) \in \mathcal{V}} \rho_v \| \mathbf{r}_v(\mathbf{z}_{v_k}^{c_i}, \mathcal{X}) \|_{\Sigma_{v_k}^{c_i}}^2 \Bigg\},
\end{aligned} \tag{4}$$

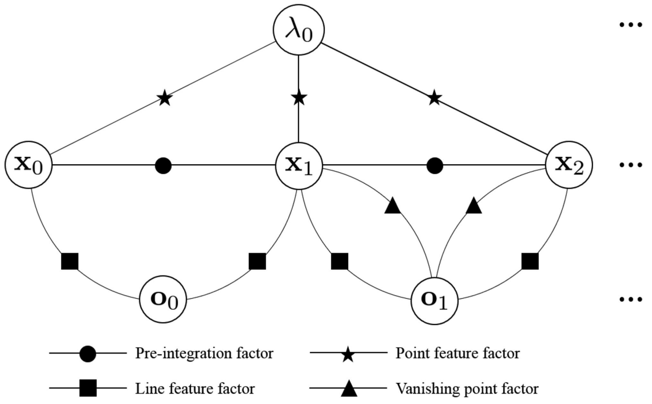


Fig. 3. An example of a factor graph for UV-SLAM. For $\mathbf{o}_0$, only the line feature factor is used as a nonstructural line. Therefore, only line feature factor is employed. On the other hand, $\mathbf{o}_1$ is a line with structural regularity and both line feature factor and the vanishing point factor are used.

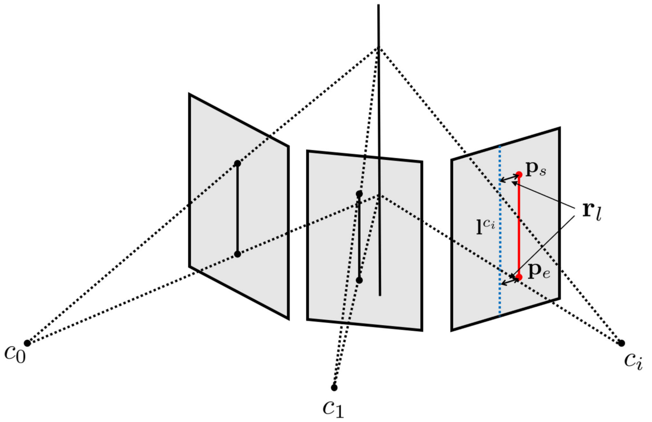


Fig. 4. Illustration of line residual. The red solid and blue dashed lines on the image represent observation and re-projected estimation, respectively. The 3D line obtained from triangulation is re-projected into a new frame. Afterward, the distance between both endpoints of the observed line and the re-projected line is defined as the residual $\mathbf{r}_l$ of the line.

where $\mathbf{r}_0$, $\mathbf{r}_I$, $\mathbf{r}_p$, $\mathbf{r}_l$, and $\mathbf{r}_v$ represent marginalization, IMU, point, line, and vanishing point measurement residuals, respectively. In addition, $\mathbf{z}_{b_{i+1}}^{b_i}$, $\mathbf{z}_{p_j}^{c_i}$, $\mathbf{z}_{l_k}^{c_i}$, and $\mathbf{z}_{v_k}^{c_i}$ stand for observations of IMU, point, line, and vanishing point, respectively; $\mathcal{B}$ is the set of all pre-integrated IMU measurements in a sliding window; $\mathcal{P}$, $\mathcal{L}$ and $\mathcal{V}$ are the sets of point, line, and vanishing point measurements in observed frames; and $\Sigma_{b_{i+1}}^{b_i}$, $\Sigma_{p_j}^{c_i}$, $\Sigma_{l_k}^{c_i}$ and $\Sigma_{v_k}^{c_i}$ represent IMU, point, line, and vanishing point measurement covariance matrices, respectively. ρ_p, ρ_l, and ρ_v mean loss functions of the point, line, and vanishing point measurements, respectively. ρ_p and ρ_l are set to the Huber norm function [40] and ρ_v is set to the inverse tangent function because of the vanishing point measurement model's unbound problem. An example of the factor graph for the defined cost function is shown in Fig. 3. If there is no vanishing point measurement for a specific line, only the line feature factor is used as in the case of $\mathbf{o}_0$. If a specific line has corresponding vanishing point measurement, the line feature and vanishing point factors are employed as in the case of $\mathbf{o}_1$. For the optimization process, Ceres Solver [41] is used.

D. Line Measurement Model

First, the re-projection of the 3D line $\mathbf{L}$ in the Plücker coordinate is as follows:

$$\mathbf{l}^c = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} = \mathbf{K}'\mathbf{n}^c = f_x f_y (\mathbf{K}^{-1})^\top \mathbf{n}^c$$

$$= \begin{bmatrix} f_y & 0 & 0 \\ 0 & f_x & 0 \\ -f_y c_x & -f_x c_y & f_x f_y \end{bmatrix} \mathbf{n}^c = \mathbf{n}^c, \tag{5}$$

where $\mathbf{l}$, $\mathbf{K}'$ and $\mathbf{K}$ represent the re-projected line, the projection matrix of a line feature, and the camera's intrinsic parameter, respectively. (f_x, f_y) and (c_x, c_y) denote image's focal lengths and principal points, respectively. Because the proposed algorithm applies to a normalized plane, $\mathbf{K}$ and $\mathbf{K}'$ are identity matrices. As a result, the re-projected line is equal to the normal vector in the proposed method.

As shown in Fig. 4, the residual of the line measurement model is defined as the following re-projection error:

$$\mathbf{r}_l = \begin{bmatrix} d(\mathbf{p}_s, \mathbf{l}^c) \\ d(\mathbf{p}_e, \mathbf{l}^c) \end{bmatrix}, \tag{6}$$

where

$$d(\mathbf{p}, \mathbf{l}^c) = \frac{\mathbf{p}^\top \mathbf{l}^c}{l_d},\ l_d = \sqrt{l_1^2 + l_2^2},$$

$$\mathbf{p}_s = (u_s, v_s, 1),\ \mathbf{p}_e = (u_e, v_e, 1), \tag{7}$$

and $\mathbf{r}_l$ denotes the line residual and d denotes the distance between both endpoints of the observed line and the re-projected line. $\mathbf{p}_s$ and $\mathbf{p}_e$ are the endpoints of the observed line in the image. The corresponding Jacobian matrix with respect to the 3D line can be represented by the body state change, $\delta\mathbf{x}$, and the orthonormal representation change, $\delta\mathbf{o}$, as follows:

$$\mathbf{J}_l = \frac{\partial \mathbf{r}_l}{\partial \mathbf{l}^c} \frac{\partial \mathbf{l}^c}{\partial \mathbf{L}^c} \begin{bmatrix} \frac{\partial \mathbf{L}^c}{\partial \delta\mathbf{x}} & \frac{\partial \mathbf{L}^c}{\partial \mathbf{L}^w} \frac{\partial \mathbf{L}^w}{\partial \delta\mathbf{o}} \end{bmatrix}, \tag{8}$$

with

$$\frac{\partial \mathbf{r}_l}{\partial \mathbf{l}^c} = \begin{bmatrix} \frac{-l_1(\mathbf{p}_s^\top \mathbf{l}^c)}{l_d^3} + \frac{u_s}{l_d} & \frac{-l_2(\mathbf{p}_s^\top \mathbf{l}^c)}{l_d^3} + \frac{v_s}{l_d} & \frac{1}{l_d} \\ \frac{-l_1(\mathbf{p}_e^\top \mathbf{l}^c)}{l_d^3} + \frac{u_e}{l_d} & \frac{-l_2(\mathbf{p}_e^\top \mathbf{l}^c)}{l_d^3} + \frac{v_e}{l_d} & \frac{1}{l_d} \end{bmatrix}_{2\times3},$$

$$\frac{\partial \mathbf{l}^c}{\partial \mathbf{L}^c} = \begin{bmatrix} \mathbf{K}' & \mathbf{0}_{3\times3} \end{bmatrix}_{3\times6},$$

$$\frac{\partial \mathbf{L}^c}{\partial \delta\mathbf{x}} =$$

$$(\mathcal{T}_c^b)^{-1} \begin{bmatrix} (\mathbf{R}_b^w)^\top [\mathbf{d}^w]_\times & [(\mathbf{R}_b^w)^\top (\mathbf{n}^w + [\mathbf{d}^w]_\times \mathbf{p}_b^w)]_\times \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix.$$

$$\mathbf{0}_{6\times3}\ \mathbf{0}_{6\times3}\ \mathbf{0}_{6\times3} \Big]_{6\times15},$$

$$\frac{\partial \mathbf{L}^c}{\partial \mathbf{L}^w} \frac{\partial \mathbf{L}^w}{\partial \delta\mathbf{o}} =$$

$$(\mathcal{T}_c^b)^{-1} \begin{bmatrix} \mathbf{0}_{3\times1} & -w_1\mathbf{u}_3 & w_1\mathbf{u}_2 & -w_2\mathbf{u}_1 \\ w_2\mathbf{u}_3 & \mathbf{0}_{3\times1} & -w_2\mathbf{u}_1 & w_1\mathbf{u}_2 \end{bmatrix}_{6\times4}, \tag{9}$$

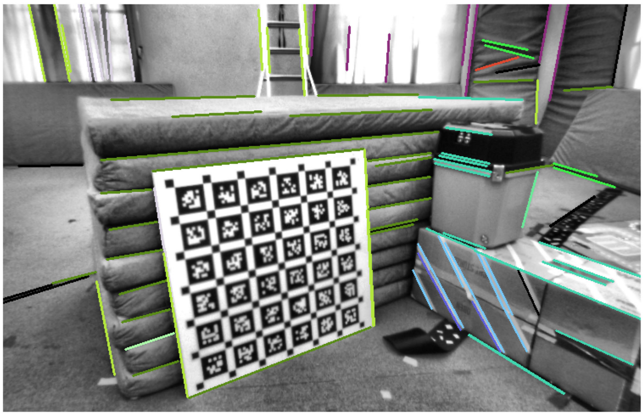


Fig. 5. Image of clustered lines using vanishing points. The lines with the same vanishing point are expressed in the same color. Usually, three or more vanishing points can be extracted using J-linkage.

where

$$\mathbf{U} = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3\end{bmatrix} = \begin{bmatrix}\frac{\mathbf{n}}{\parallel \mathbf{n} \parallel} & \frac{\mathbf{d}}{\parallel \mathbf{d} \parallel} & \frac{\mathbf{n}\times\mathbf{d}}{\parallel \mathbf{n}\times\mathbf{d} \parallel}\end{bmatrix},$$

$$\mathbf{w} = \begin{bmatrix}w_1\\ w_2\end{bmatrix} = \frac{1}{\sqrt{\parallel \mathbf{n} \parallel^2 + \parallel \mathbf{d} \parallel^2}}\begin{bmatrix}\parallel \mathbf{n} \parallel\\ \parallel \mathbf{d} \parallel\end{bmatrix}, \tag{10}$$

and $\mathcal{T}_c^b$ is a transformation matrix from the camera coordinate to the body coordinate in the Plücker coordinate.

E. Vanishing Point Measurement Model

After the vanishing points are calculated, the observed line features use the corresponding vanishing points as new observations. An example of clustering lines through vanishing points is shown in Fig. 5. The lines with the same vanishing point are expressed in the same color.

To estimate the vanishing points, the point on the 3D line is expressed in a homogeneous coordinate as follows [18]:

$$\mathbf{V}(t) = \mathbf{V}_0 + t\mathbf{D} = \begin{bmatrix}x_0 + td_1\\ y_0 + td_2\\ z_0 + td_3\\ 1\end{bmatrix}, t \in (0, \infty), \tag{11}$$

where

$$\mathbf{V}_0 = \begin{bmatrix}x_0 & y_0 & z_0 & 1\end{bmatrix}^\top,$$

$$\mathbf{D} = \begin{bmatrix}\mathbf{d}^c\top, 0\end{bmatrix}^\top = \begin{bmatrix}d_1, d_2, d_3, 0\end{bmatrix}^\top, \tag{12}$$

and $\mathbf{V}_0$ represents a point on the 3D line. Then, the vanishing point equals the projection of a point at infinity on the 3D line as follows:

$$\mathbf{v}^c = \begin{bmatrix}v_1\\ v_2\\ v_3\end{bmatrix} = \lim_{t\to\infty}\mathbf{P}(\mathbf{V}_0 + t\mathbf{D}) = \mathbf{K}\mathbf{d}^c = \mathbf{d}^c, \tag{13}$$

where $\mathbf{P} = \mathbf{K}[\mathbf{I}|\mathbf{0}]$ is a camera projection matrix. In the UV-SLAM, the vanishing point from the line is equal to the direction vector of the line. The vanishing point estimation is calculated by the intersection of the $\mathbf{v}^c$ and the image plane, as shown in Fig. 6. Finally, the vanishing point residual is as follows:

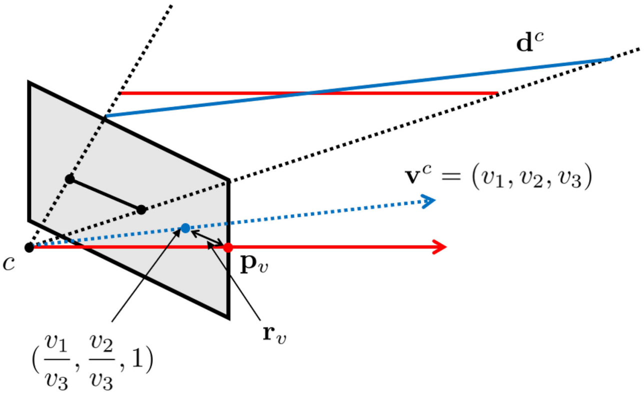

$\mathbf{d}^c$
$\mathbf{v}^c = (v_1, v_2, v_3)$
c
$\mathbf{p}_v$
$\mathbf{r}_v$
$(\frac{v_1}{v_3}, \frac{v_2}{v_3}, 1)$

Fig. 6. Illustration of the vanishing point residual. The points where the red solid and blue dashed lines intersect with the image represent observation and estimation, respectively. In vanishing point estimation, the direction vector of the line is used. The vanishing point difference between the observation from the observed line and the estimation from the 3D line is defined as the residual.

$$\mathbf{r}_v = \mathbf{p}_v - \frac{1}{v_3}\begin{bmatrix}v_1\\ v_2\end{bmatrix}, \tag{14}$$

where $\mathbf{r}_v$ and $\mathbf{p}_v$ represent the vanishing point residual and the vanishing point observation, respectively. The corresponding Jacobian matrix with respect to the vanishing point can be obtained in terms of $\delta\mathbf{x}$ and $\delta\mathbf{o}$ as follows:

$$\mathbf{J}_v = \frac{\partial\mathbf{r}_v}{\partial\mathbf{v}^c}\frac{\partial\mathbf{v}^c}{\partial\mathbf{L}^c}\begin{bmatrix}\frac{\partial\mathbf{L}^c}{\partial\delta\mathbf{x}} & \frac{\partial\mathbf{L}^c}{\partial\mathbf{L}^w}\frac{\partial\mathbf{L}^w}{\partial\delta\mathbf{o}}\end{bmatrix} \tag{15}$$

where

$$\frac{\partial\mathbf{r}_v}{\partial\mathbf{v}^c} = \begin{bmatrix}-\frac{1}{v_3} & 0 & \frac{v_1}{v_3^2}\\ 0 & -\frac{1}{v_3} & \frac{v_2}{v_3^2}\end{bmatrix}_{2\times3},$$

$$\frac{\partial\mathbf{v}^c}{\partial\mathbf{L}^c} = \begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{K}\end{bmatrix}_{3\times6}. \tag{16}$$

IV. Fisher Information Matrix Rank Analysis

We rigorously analyze the observability of line features through Fisher information matrix (FIM) rank analysis. If the FIM is singular, the system is unobservable [42]. Some approaches employed the fact the Jacobian matrix used in the FIM calculation must satisfy the full column rank condition for the FIM to satisfy nonsingularity [43], [44]. We also use this approach to analyze the observability of the proposed method.

First, the FIM of the line measurement in the orthonormal representation is as follows:

$$\mathbf{H}_l^{\delta\mathbf{o}} = \mathbf{J}_l^{\delta\mathbf{o}\top}\mathbf{\Omega}_l^{\delta\mathbf{o}}\mathbf{J}_l^{\delta\mathbf{o}}, \tag{17}$$

where

$$\mathbf{J}_l^{\delta\mathbf{o}} = \frac{\partial\mathbf{r}_l}{\partial\mathbf{l}^c}\frac{\partial\mathbf{l}^c}{\partial\mathbf{L}^c}\frac{\partial\mathbf{L}^c}{\partial\mathbf{L}^w}\frac{\partial\mathbf{L}^w}{\partial\delta\mathbf{o}}$$

$$= \begin{bmatrix}0 & j_{l_{12}} & j_{l_{13}} & j_{l_{14}}\\ 0 & j_{l_{22}} & j_{l_{23}} & j_{l_{24}}\end{bmatrix}, \tag{18}$$

and $j_{l_{pq}}$ is the non-zero element in the p-th row and the q-th column of $\mathbf{J}_l^{\delta\mathbf{o}}$, which is obtained by substituting (9) into (18).

$\boldsymbol{\Omega}_l^{\delta\mathbf{o}}$ represents the inverse of covariance matrix of the line observation. Because $\frac{\partial \mathbf{l}^c}{\partial \mathbf{L}^c}$, $\frac{\partial \mathbf{L}^c}{\partial \mathbf{L}^w}$, and $\frac{\partial \mathbf{L}^w}{\partial \delta\mathbf{o}}$ are full rank matrices from (9), the rank of $\mathbf{J}_l^{\delta\mathbf{o}}$ is determined by $\frac{\partial \mathbf{r}_l}{\partial \mathbf{l}^c}$ in (18). From (9), the maximum rank of $\frac{\partial \mathbf{r}_l}{\partial \mathbf{l}^c}$ is 2, and the case in which the rank becomes 1 is as follows:

$$\frac{l_2}{l_1} = \frac{v_s - v_e}{u_s - u_e}. \tag{19}$$

However, because a line in the orthonormal representation has four parameters, the observability of the line cannot be guaranteed with the line measurement model alone. To solve this problem, a new observation on the line other than both endpoints is introduced as follows:

$$\mathbf{p}_l = \alpha \mathbf{p}_s + (1-\alpha)\mathbf{p}_e, \alpha \in (0,1), \tag{20}$$

where $\mathbf{p}_l$ represents the new observation. However, despite adding a new observation, the rank of $\mathbf{J}_l^{\delta\mathbf{o}}$ is up to 2. Therefore, the line features using only the line measurement model are still not observable. From a new perspective, we propose to calculate the FIM with the vanishing point measurements. The FIM of the vanishing point measurement in the orthonormal representation is as follows:

$$\mathbf{H}_v^{\delta\mathbf{o}} = \mathbf{J}_v^{\delta\mathbf{o}\top}\boldsymbol{\Omega}_v^{\delta\mathbf{o}}\mathbf{J}_v^{\delta\mathbf{o}}, \tag{21}$$

where

$$\begin{aligned}\mathbf{J}_v^{\delta\mathbf{o}} &= \frac{\partial \mathbf{r}_v}{\partial \mathbf{v}^c}\frac{\partial \mathbf{v}^c}{\partial \mathbf{L}^c}\frac{\partial \mathbf{L}^c}{\partial \mathbf{L}^w}\frac{\partial \mathbf{L}^w}{\partial \delta\mathbf{o}} \\ &= \begin{bmatrix} j_{v_{11}} & 0 & j_{v_{13}} & j_{v_{14}} \\ j_{v_{21}} & 0 & j_{v_{23}} & j_{v_{24}} \end{bmatrix},\end{aligned} \tag{22}$$

and $j_{v_{pq}}$ is the non-zero element in the p-th row and the q-th column of $\mathbf{J}_v^{\delta\mathbf{o}}$, which is obtained by substituting (16) into (22). $\boldsymbol{\Omega}_v^{\delta\mathbf{o}}$ represents the inverse of covariance matrix of the vanishing point observation. Similarly, all matrices are full rank except $\frac{\partial \mathbf{r}_v}{\partial \mathbf{v}^c}$. Therefore, the rank of $\mathbf{J}_v^{\delta\mathbf{o}}$ is 2, which can be obtained from the rank of $\frac{\partial \mathbf{r}_v}{\partial \mathbf{v}^c}$ in (16).

By eigenvalue decomposition, the FIM considering both the line measurement and the vanishing point measurement can be obtained as follows:

$$\begin{aligned}\mathbf{H}^{\delta\mathbf{o}} &= \mathbf{H}_l^{\delta\mathbf{o}} + \mathbf{H}_v^{\delta\mathbf{o}} \\ &= \mathbf{J}_l^{\delta\mathbf{o}\top}\boldsymbol{\Omega}_l^{\delta\mathbf{o}}\mathbf{J}_l^{\delta\mathbf{o}} + \mathbf{J}_v^{\delta\mathbf{o}\top}\boldsymbol{\Omega}_v^{\delta\mathbf{o}}\mathbf{J}_v^{\delta\mathbf{o}} \\ &= \mathbf{J}^{\delta\mathbf{o}\top}\boldsymbol{\Omega}^{\delta\mathbf{o}}\mathbf{J}^{\delta\mathbf{o}},\end{aligned} \tag{23}$$

where

$$\begin{aligned}\mathbf{J}^{\delta\mathbf{o}} &= \begin{bmatrix} 0 & j_{l_{12}} & j_{l_{13}} & j_{l_{14}} \\ 0 & j_{l_{22}} & j_{l_{23}} & j_{l_{24}} \\ j_{v_{11}} & 0 & j_{v_{13}} & j_{v_{14}} \\ j_{v_{21}} & 0 & j_{v_{23}} & j_{v_{24}} \end{bmatrix}, \\ \boldsymbol{\Omega}^{\delta\mathbf{o}} &= \begin{bmatrix} \boldsymbol{\Omega}_l^{\delta\mathbf{o}} & 0 \\ 0 & \boldsymbol{\Omega}_v^{\delta\mathbf{o}} \end{bmatrix}.\end{aligned} \tag{24}$$

At this time, the rows of the line measurement and the vanishing point measurement are independent in $\mathbf{J}^{\delta\mathbf{o}}$. Therefore, the rank of $\mathbf{J}^{\delta\mathbf{o}}$ is 4, except for the case of (19). We can confirm that the line features become fully observable by additionally using the vanishing point meausrement model.

TABLE I
TRANSLATIONAL RMSE WITHOUT LOOP CLOSING FOR THE EUROC DATASETS (UNIT: M)

Translation RMSE	VINS-Mono	PL-VINS	ALVIO	Our method in [4]	UV-SLAM
MH_01_easy	0.159	0.164	0.148	0.142	**0.139**
MH_02_easy	0.140	0.174	0.136	0.126	**0.094**
MH_03_medium	0.225	**0.187**	0.209	0.198	0.189
MH_04_difficult	0.408	0.335	0.389	0.301	**0.261**
MH_05_difficult	0.312	0.347	0.317	0.293	**0.188**
V1_01_easy	0.094	0.071	0.085	0.087	**0.067**
V1_02_medium	0.115	0.086	0.075	0.072	**0.070**
V1_03_difficult	0.203	0.152	0.200	0.156	**0.109**
V2_01_easy	0.099	0.090	0.094	0.098	**0.085**
V2_02_medium	0.161	0.120	0.133	**0.103**	0.112
V2_03_difficult	0.341	0.278	0.288	0.277	**0.213**

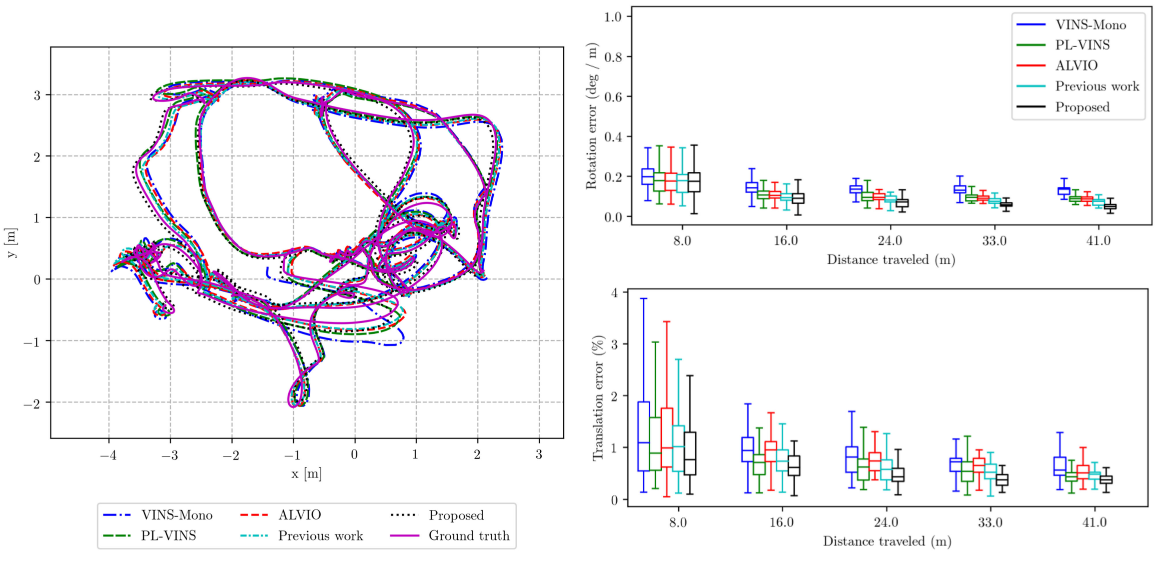
V. EXPERIMENTAL RESULTS

The experiments were carried out on an Intel Core i7-9700 K processor with 32 GB of RAM. Using the EuRoC micro aerial vehicle (MAV) datasets [45], we tested the state-of-the-art algortihms and the UV-SLAM. Each dataset offers a varied level of complexity depending on factors like lighting, texture, and MAV speed. Therefore, the datasets were appropriate to validate the performance of the proposed method.

We compared the localization accuracy of the proposed method with that of VINS-Mono which is our base algorithm. In addition, we also compared PL-VINS [16], ALVIO, and our previous work which use line features on top of VINS-Mono. The parameters of compared algorithms are set to the default values in the open-source codes. We employed the rpg trajectory evaluation tool [46]. Table I shows the translational root mean square error (RMSE) for the EuRoC datasets. The proposed method has better performance than state-of-the-art algorithms in all datasets. In particular, the proposed algorithm shows 32.3%, 23.8%, 26.4%, and 17.6% smaller average error than VINS-Mono, PL-VINS, ALVIO, and our previous work, respectively. More accurate results could be obtained in the proposed algorithm because the line features become fully observable using the vanishing point measurements. The results for trajectory, rotation error, and translation error of `V2_02_medium` in the EuRoC datasets are shown in Fig. 7.

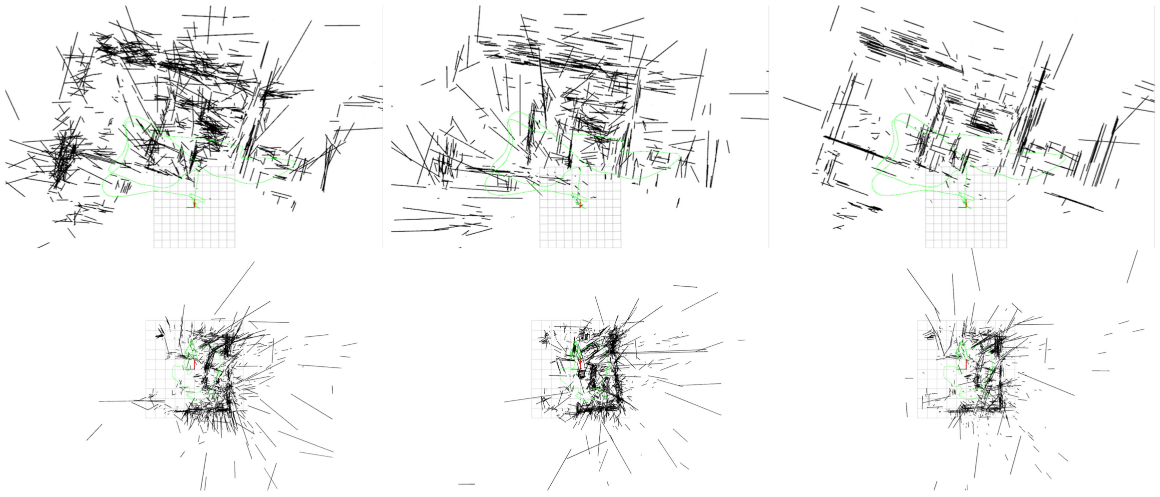
In addition, the mapping results for `MH_05_difficult` and `V2_01_easy` in the EuRoC datasets are shown in Fig. 8. In the case of ALVIO, the quality of mapping is low due to degenerate lines. In our previous work, degenerate lines were corrected only in pure translational camera motion. Noteworthily, lines' direction vectors are aligned thanks to the vanishing point measurements in UV-SLAM. All top views of the line mapping results for the EuRoC datasets are available at: https://github.com/url-kaist/UV-SLAM/blob/main/mapping_result.pdf.

The average runtime is about 53.528 ms for the frontend and 47.086 ms for the backend for the EuRoC datasets. UV-SLAM has only about 3 ms longer frontend runtime than other algorithms because it extracts vanishing points. Moreover, the runtime of the backend corresponding to optimization is similar to those of other algorithms. This is because the proposed vanishing point measurement model does not use new parameters.



(a) (b)

Fig. 7. The top views of (a) trajectory and (b) boxplot of RMSE according to distance traveled for VINS-Mono, PL-VINS, ALVIO, our previous work [4], and UV-SLAM for `V2_02_medium` in the EuRoC datasets.



(a) (b) (c)

Fig. 8. The top views of line mapping results of (a) ALVIO, (b) our previous work [4], and (c) UV-SLAM for `MH_05_difficult` (top) and `V2_01_easy` (bottom) in the EuRoC datasets.

VI. Conclusion

In summary, we proposed UV-SLAM, which is the unconstrained line-based SLAM using the vanishing point measurements. The proposed method can be used without any assumptions such as the Manhattan or Atlanta world. We calculated the residual and Jacobian matrices of the vanishing point measurements. Through FIM rank analysis, we verified that line's observability is guaranteed by introducing the vanishing point measurements into the existing method. In addition, we showed that localization accuracy and mapping quality have increased through quantitative and qualitative comparisons with state-of-the-art algorithms. For future work, we will implement mesh or pixel-wise mapping through sparse line mapping from the proposed algorithm.

References

[1] J. Jeon, S. Jung, E. Lee, D. Choi, and H. Myung, "Run your visual-inertial odometry on NVIDIA jetson: Benchmark tests on a micro aerial vehicle," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 5332–5339, Jul. 2021.

[2] 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*, vol. 18, no. 4, 2018, Art. no. 1159.

[3] K. Jung, Y. Kim, H. Lim, and H. Myung, "ALVIO: Adaptive line and point feature-based visual inertial odometry for robust localization in indoor environments," in *Proc. Int. Conf. Robot Intell. Technol. Appl.*, 2020, pp. 171–184, https://arxiv.org/abs/2012.15008

[4] H. Lim, Y. Kim, K. Jung, S. Hu, and H. Myung, "Avoiding degeneracy for monocular visual SLAM with point and line features," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 11675–11681.

[5] D. G. Kottas and S. I. Roumeliotis, "Efficient and consistent vision-aided inertial navigation using line observations," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2013, pp. 1540–1547.

[6] X. Kong, W. Wu, L. Zhang, and Y. Wang, "Tightly-coupled stereo visual-inertial navigation using point and line features," *Sensors*, vol. 15, no. 6, pp. 12 816–12833, 2015.

[7] G. Zhang, J. H. Lee, J. Lim, and I. H. Suh, "Building a 3-D line-based map using stereo SLAM," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1364–1377, Dec. 2015.

[8] A. Bartoli and P. Sturm, "Structure-from-motion using lines: Representation, triangulation, and bundle adjustment," *Comput. Vis. Image Understanding*, vol. 100, no. 3, pp. 416–441, 2005.

[9] 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," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 3686–3693.

[10] Y. Yang, P. Geneva, K. Eckenhoff, and G. Huang, "Visual-inertial odometry with point and line features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 2447–2454.

[11] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. Robot. Automat*, 2007, pp. 3565–3572.

[12] A. Pumarola, A. Vakhitov, A. Agudo, A. Sanfeliu, and F. Moreno-Noguer, "PL-SLAM: Real-time monocular visual SLAM with points and lines," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 4503–4508.

[13] X. Zuo, X. Xie, Y. Liu, and G. Huang, "Robust visual SLAM with point and line features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 1775–1782.

[14] S. J. Lee and S. S. Hwang, "Elaborate monocular point and line SLAM with robust initialization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1121–1129.

[15] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[16] Q. Fu, J. Wang, H. Yu, I. Ali, F. Guo, and H. Zhang, "PL-VINS: Real-time monocular visual-inertial SLAM with point and line," 2020, *arXiv:2009.07462*.

[17] T. Qin, P. Li, and S. Shen, "VINS-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[18] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[19] T. Sugiura, A. Torii, and M. Okutomi, "3D surface reconstruction from point-and-line cloud," in *Proc. Int. Conf. 3D Vis.*, 2015, pp. 264–272.

[20] H. Zhou, D. Zhou, K. Peng, W. Fan, and Y. Liu, "SLAM-based 3D line reconstruction," in *Proc. World Congr. Intell. Control Automat.*, 2018, pp. 1148–1153.

[21] Y. Yang and G. Huang, "Observability analysis of aided INS with heterogeneous features of points, lines, and planes," *IEEE Trans. Robot.*, vol. 35, no. 6, pp. 1399–1418, Dec. 2019.

[22] A. ö. Ok, J. D. Wegner, C. Heipke, F. Rottensteiner, U. Sörgel, and V. Toprak, "Accurate reconstruction of near-epipolar line segments from stereo aerial images," *Photogrammetrie-Fernerkundung-Geoinformation*, no. 4, pp. 345–358, 2012.

[23] P. Kim, B. Coltin, and H. J. Kim, "Low-drift visual odometry in structured environments by decoupling rotational and translational motion," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 7247–7253.

[24] "Indoor RGB-D compass from a single line and plane," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4673–4680.

[25] Y. Li, N. Brasch, Y. Wang, N. Navab, and F. Tombari, "Structure-SLAM: Low-drift monocular SLAM in indoor environments," *IEEE Robot. Automat. Lett.*, vol. 5, no. 4, pp. 6583–6590, Oct. 2020.

[26] Y. Li, R. Yunus, N. Brasch, N. Navab, and F. Tombari, "RGB-D SLAM with structural regularities," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 11 581–11 587.

[27] R. Yunus, Y. Li, and F. Tombari, "ManhattanSLAM: Robust planar tracking and mapping leveraging mixture of manhattan frames," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 6687–6693.

[28] H. Zhou, D. Zou, L. Pei, R. Ying, P. Liu, and W. Yu, "StructSLAM: Visual SLAM with building structure lines," *IEEE Trans. Veh. Technol.*, vol. 64, no. 4, pp. 1364–1375, Apr. 2015.

[29] 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.*, vol. 35, no. 4, pp. 999–1013, Aug. 2019.

[30] B. Xu, P. Wang, Y. He, Y. Chen, Y. Chen, and M. Zhou, "Leveraging structural information to improve point line visual-inertial odometry," *arXiv preprint arXiv:2105.04064*, 2021.

[31] J. Lee and S.-Y. Park, "PLF-VINS: Real-time monocular visual-inertial SLAM with point-line fusion and parallel-line fusion," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 7033–7040, Oct. 2021.

[32] J. Ma, X. Wang, Y. He, X. Mei, and J. Zhao, "Line-based stereo SLAM by junction matching and vanishing point alignment," *IEEE Access*, vol. 7, pp. 181800–181811, 2019.

[33] J. Shi *et al.*, "Good features to track," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1994, pp. 593–600.

[34] C. Tomasi and T. Kanade, "Detection and tracking of point," *Int. J. Comput. Vis.*, vol. 9, pp. 137–154, 1991.

[35] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual-inertial odometry," *IEEE Trans. Robot.*, vol. 33, no. 1, pp. 1–21, Feb. 2017.

[36] G. Sibley, L. Matthies, and G. Sukhatme, "Sliding window filter with application to planetary landing," *J. Field Robot.*, vol. 27, no. 5, pp. 587–608, 2010.

[37] 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.*, vol. 32, no. 4, pp. 722–732, Apr. 2010.

[38] 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 Representation*, vol. 24, no. 7, pp. 794–805, 2013.

[39] R. Toldo and A. Fusiello, "Robust multiple structures estimation with J-linkage," in *Proc. Euro. Conf. Comput. Vision*, Springer, 2008, pp. 537–547.

[40] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in Statistics*. Berlin, Germany: Springer, 1992, pp. 492–518.

[41] S. Agarwal, K. Mierle, and Others, "Ceres solver," Accessed on: Sep. 09, 2021. [Online], Available: http://ceres-solver.org

[42] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation With Applications to Tracking and Navigation: Theory, Algorithms, and Software*. Hoboken, NJ, USA: Wiley, 2004.

[43] Z. Wang and G. Dissanayake, "Observability analysis of SLAM using fisher information matrix," in *Proc. IEEE Int. Conf. Control, Automation, Robot. Vis.*, 2008, pp. 1242–1247.

[44] S.-M. Lee, J. Jung, S. Kim, I.-J. Kim, and H. Myung, "DV-SLAM (dual-sensor-based vector-field SLAM) and observability analysis," *IEEE Trans. Ind. Electron.*, vol. 62, no. 2, pp. 1101–1112, Feb. 2015.

[45] M. Burri *et al.*, "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016.

[46] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 7244–7251.