2023 IEEE 26th International Conference on
Intelligent Transportation Systems (ITSC)
24-28 September 2023. Bilbao, Bizkaia, Spain

Tightly-coupled Line Feature-aided Visual Inertial Localization within Lightweight 3D Prior Map for Intelligent Vehicles

Xi Zheng, Weisong Wen*, Li-Ta Hsu

***Abstract*— Visual-inertial navigation system (VINS) is widely used for autonomous platforms but suffers from drifting over a long time. To remedy this situation, a lightweight 3D prior map-aided visual-inertial navigation system is presented in this paper, which tightly couples the visual-inertial data stream with a lightweight prior map involving 3D line information. To fill the gaps between 3D maps and 2D images, the mutual geometric feature of line segments is utilized to connect these two types of information in different dimensions. By detecting and matching line features in two data sources, the line pairs are utilized as constraints in the nonlinear optimization model and added to the existing factor graph framework in a tightly coupled form. Meanwhile, a fast line feature tracking strategy is employed to monitor and remove extreme outliers, which will further improve the reliability of this structural characteristic during the cross-modality localization. The effectiveness of the proposed method is evaluated by public indoor unmanned aerial vehicles (UAV) datasets, and outdoor unmanned ground vehicles (UGV) datasets generated by the CARLA simulator.**

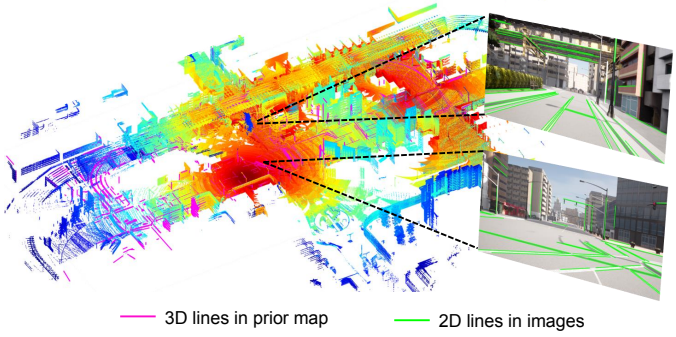

Fig. 1: The demonstration graph of 3D lines in the point cloud map and 2D lines in images. Lines from different sources are matched as line correspondences, which will be integrated as constraints for state estimation.

I. Introduction

Localization is a fundamental function of the autonomous navigation system, specifically, high-precision and low-cost characteristics are indispensable requirements for practical applications, such as self-driving vehicles [1] and unmanned aerial vehicles (UAV) [2]. The global navigation satellite system (GNSS) [3] could provide global and drift-free position measurements for localization and has been widely used for autonomous systems. However, GNSS signals can be severely affected by surrounding buildings or vehicles in urban environments because of signal occlusion and reflection. It could cause a decrease in the number of available satellites, non-light-of-sight receptions, and multipath effects [4], which lead to compromised localization performance. In particular, GNSS measurements are unavailable in indoor scenarios which limits its application in related environments. The 3D mapping aided GNSS positioning [5], [6], 3D LiDAR aided GNSS positioning methods [7], [8] are proposed to mitigate the impacts of the GNSS multipath and NLOS receptions. However, these methods relied on precise prior 3D building models or expensive 3D LiDAR sensors.

This research is supported by the Guangdong Basic and Applied Basic Research Foundation (2021A1515110771), and the University Grants Committee of Hong Kong under the scheme Research Impact Fund on the project R5009-21 "Reliable Multiagent Collaborative Global Navigation Satellite System Positioning for Intelligent Transportation Systems". The authors are with the Department of Aeronautical and Aviation Engineering, the Hong Kong Polytechnic University, Hong Kong, zheng-xi.zheng@connect.polyu.hk, lt.hsu@polyu.edu.hk

* Corresponding author: Weisong Wen welson.wen@polyu.edu.hk

Vision-based and light detection and ranging (LiDAR)-based simultaneous localization and mapping (SLAM) [9] technologies have been increasingly sophisticated and are used in a variety of applications. They are constantly fused with inertial measurement units (IMU) to improve localization accuracy [10]–[12]. LiDAR-based SLAM usually provides more accurate localization results. However, the high cost of the LiDAR sensor is one of the key factors limiting its massive deployment. Vision-based SLAM methods can supply rich image information and are also widely used due to their lightweight and low-cost nature. However, the accuracy of vision-based localization systems has a large offset in long distance scenes. Moreover, camera perception is disturbed in complex illumination and low-texture situations, and monocular cameras will face scale ambiguity problems.

Another option is to combine the advantages of LiDAR and camera, by constructing a dense point cloud map with high-precision LiDAR as prior information and then using the camera-based solution to solve the localization problem in this map [13]–[15]. This solution could utilize the accurate 3D information provided by LiDAR to correct the drift of visual localization. However, the dense map can cause a computational burden. Meanwhile, The cross-modality alignment between 3D point clouds and 2D pixel images has certain obstacles because of the pattern difference problem [16]. To address these issues, there are some algorithms proposed.

In terms of data representation, the existing methods are classified into two categories: image structure-based and geometric structure-based. These methods usually first

2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC) | 979-8-3503-9946-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/ITSC57777.2023.10422196

convert data from different modalities to the same dimension and then perform state estimation. Concerning the image structure-based methods, in [17], synthetic images are generated from numerous views of environments by the surface reflectivities of 3D LiDAR map, which are evaluated with the online camera images via maximizing normalized mutual information for visual localization. Lu et al. [18] utilized the manually labeled road markings in a 3D map represented as a set of sparse points to register the edges of the landmarks detected in images by Chamfer matching [19], which is built as the epipolar geometry constraints for the pose optimization problem. The above algorithms are restricted in accuracy and usage scenarios due to the dependence on known templates. The method in [14] built vertex and normal maps by rendering a prior surfel map. It used this global planar information to correspond with tracked direct sparse image features to construct constraints for pose estimation. However, this method relies on depth maps to build the 2D-3D constraints, and when image feature tracking fails, it reverts to the traditional visual localization method. Huang et al. [20] adopted a similar architecture as the previous paper [14], but the 3D prior map is modeled using the Gaussian mixture model to associate with image features. Meanwhile, this paper used the pose results from the SLAM front-end for initialization and only used 2D-3D constraints for back-end optimization, which improves initialization stability but must trigger back-end optimization.

Regarding the geometric structure-based methods, Caselitz et al [21] utilized local bundle adjustment to obtain initial camera poses and sparse point clouds with unknown scale, which is aligned with a 3D prior map to build a 7-degree of freedom (DoF) registration problem solved by iterative closest point (ICP) scheme. In [22], a stereo visual-inertial odometry (VIO) based on the multi-state constraint Kalmen filter was used to construct semi-dense clouds. And then, the clouds were registered with the prior map by a normal distribution transform related method [23]. However, large-scale BA and depth image operation is time-consuming. Qin et al. [13] used a segmentation network to extract semantic road markings and projected these features into the world frame to merge a light-weight point cloud map, and then similar local semantic points are generated and used to align with the prior map by ICP method for pose estimation. But the lack of diversity of road markings may affect positioning accuracy.

In this paper, we hope to utilize the structured properties of prior maps to achieve cross-modality fusion of 3D point clouds and 2D images, while maintaining sufficient convenience and computational speed. Inspired by [24], The line features mutually existing in the two data sources draw our attention (the line feature demonstration result is shown in Fig. 1). Paper [24] took the pose estimation results of VINS [9] as the initial guess and constructed an optimization model to refine the pose by 2D-3D line feature constraints in a loosely coupled manner. Our previous work also [25] employed a loosely coupled framework that is similar to the paper [24], but utilized a new line correspondence-based optimization model for state estimation. In this paper, the proposed method incorporates the new line constraints as a factor [26] into the VIO nonlinear optimization model to estimate pose in a tightly coupled manner. Compared to the loosely coupled form in [24], the tightly coupled optimization structure can obtain smoother localization trajectories by combining IMU, visual point features, and cross-modal line features as constraints simultaneously. Meanwhile, to remedy the mismatching phenomenon that may be caused by the reduced accuracy of the initial values due to the tightly coupled mode, a fast-line feature tracking strategy is used to monitor this risk.

In conclusion, the contributions of this paper are summarized as follows.

1) Using a new line feature-based constraint to build the optimization model by deconstructing the intrinsic association between the line feature and point feature, and then the original error function based on the minimum distance between lines is converted into a point-to-point reprojection optimization problem.
2) A fast line feature tracking strategy based on the direct optical flow method is applied to monitor and remove outliers.
3) A tightly coupled factor graph optimization structure is constructed to fuse the line feature in the prior map with visual-inertial data to suppress drift. The proposed framework is evaluated in both indoor and outdoor environments.

The rest of this paper is organized as follows. Section II describes the pipeline and notations of the proposed system. The methodology is presented in Section III. Experimental results and discussions are provided in Section IV. Section V concludes this article and outlines future works.

II. System Overview

The pipeline of the proposed system is illustrated in Fig. 2. The proposed system uses raw IMU measurements, images, and a lightweight prior map as input data. The prior information is integrated on the basis of the classic VIO framework. It is worth pointing out that the work in this paper is carried out on the framework proposed in [9]. Therefore, the point features and IMU preintegration modules involved in this paper are consistent with the reference paper and are not elaborated on here. In terms of the work in this paper, a dense point cloud map is generated by multiple sensors (such as LiDAR, IMU, and scanning radar), in which the 3D line features are detected and memorized offline in the form of endpoints so that the massive point cloud map can be significantly lightened.

During real-time localization, the system first loads the 3D lines obtained in advance as prior information. After the initialization by VIO is completed, the 3D lines that belong to the field of view (FoV) of the cameras in the lightweight prior map are reprojected into an image based on the initial guess of the camera pose to obtain the potential line predictions. Among that, the initial guess is provided by IMU preintegration and system extrinsic parameters. At

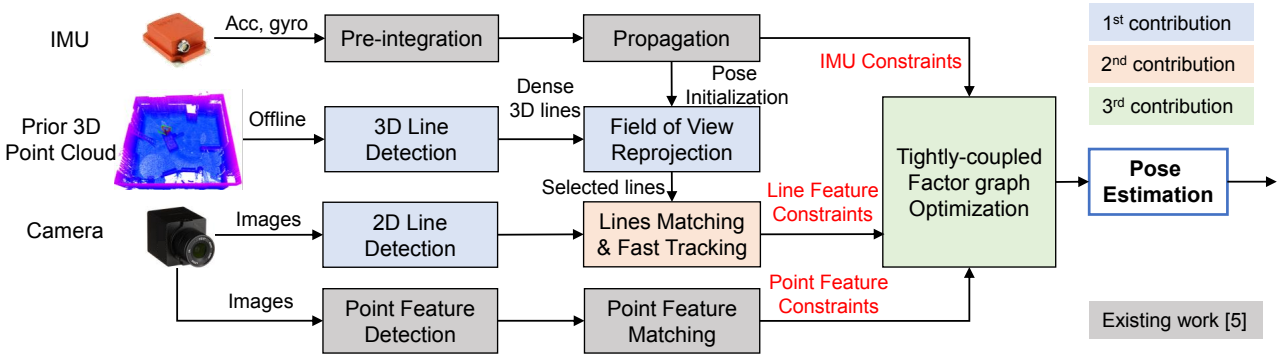

Fig. 2: The framework of the proposed method. The 3D lines in the prior map are detected offline, which are matched with the online detected 2D lines. The matched line correspondences, IMU measurements, and point feature pairs are integrated as constraints for factor graph optimization in a tightly coupled form.

the same time, the system detects 2D lines in real-time images as measurements and matches them with the line predictions from the previous step, followed by a fast sliding window tracking method to monitor and remove mismatched outliers. Eventually, the obtained line pair constraints will be combined with the original IMU preintegration and point feature constraints as factors to construct a new nonlinear optimization model in a tightly coupled form for state estimation.

The notations in this paper are defined as follows. The frames used in this paper are the world frame $(\cdot)^w$, the body frame $(\cdot)^b$ (the origin is defined on the IMU frame $(\cdot)^i$), and the camera frame $(\cdot)^c$. The state variables that need to be estimated in the system contain the $\mathbf{R}_b^w, \mathbf{t}_b^w$, which means the rotation matrix and translation vector from the body frame to the world frame. Moreover, $\mathbf{R}_b^a$ and $\mathbf{t}_b^a$ express the rotation matrix and translation vector from the b frame to the a frame. The velocity $\mathbf{v}_b^w$ of the body frame, accelerometer bias $\mathbf{b}_a$, gyroscope bias $\mathbf{b}_g$ and the inverse depth ρ for each point features.

III. State Estimation

In this section, we formulate the state estimation problem in the form of a factor graph and describe the constraint construction process based on line features, including line representation, matching and tracking mechanisms, and a complete tightly coupled optimization model.

A. Line Representation

There are various ways to represent a line in space, among which the *Plücker* coordinates and orthonormal representation are commonly used in visual motion estimation [27]–[29]. Different representations have their characteristics, and the most appropriate approach depends on the definition of the problem. Meanwhile, these approaches are interoperable and can be interchanged.

For example, Fig. 3 (a) illustrates the representation of a line in space. Given a 3D line $\boldsymbol{l}$ with two endpoints $\mathbf{P_s}$, $\mathbf{P_e}$, the *Plücker* coordinates can be expressed by $\boldsymbol{l} = (\mathbf{n}^\top, \mathbf{v}^\top)^\top$, and

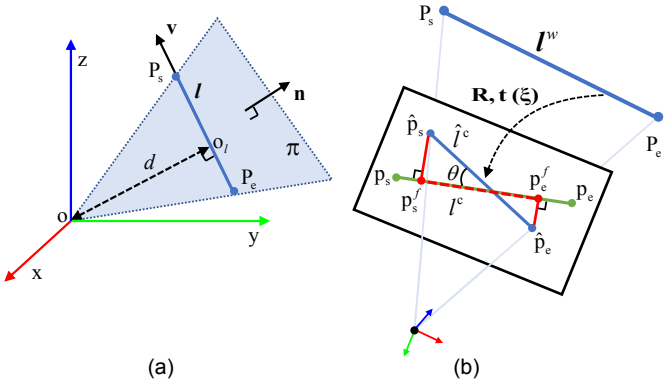

Fig. 3: (a) The line representation methods. (b) The line matching criteria and reprojection error.

$$\begin{bmatrix} \mathbf{n} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{P_s} \times \mathbf{P_e} \\ \mathbf{P_e} - \mathbf{P_s} \end{bmatrix}, \tag{1}$$

where $\mathbf{n}$ is the normal vector of the plane determined by the two endpoints and origin, $\mathbf{v}$ represents the line direction vector. The orthonormal representation $(\mathbf{U}, \mathbf{W})$ is introduced in [28] for calculating the partial derivative during nonlinear optimization, which can be expressed by

$$\mathbf{U} = \mathbf{R}(\boldsymbol{\phi}) = \begin{bmatrix} \frac{\mathbf{n}}{\|\mathbf{n}\|} & \frac{\mathbf{v}}{\|\mathbf{v}\|} & \frac{\mathbf{n}\times\mathbf{v}}{\|\mathbf{n}\times\mathbf{v}\|} \end{bmatrix}, \tag{2}$$

where, $\boldsymbol{\phi} = [\phi_1, \phi_2, \phi_3]^\top$ is the rotation angle around x-, y-, z-axes, respectively. And

$$\mathbf{W} = \frac{1}{\sqrt{(\|\mathbf{n}\|^2 + \|\mathbf{v}\|^2)}} \begin{bmatrix} \|\mathbf{n}\| & -\|\mathbf{v}\| \\ \|\mathbf{v}\| & \|\mathbf{n}\| \end{bmatrix}. \tag{3}$$

As the distance d from the origin to the line in Fig. 3 (a) can be calculated as $d = \frac{\|\mathbf{n}\|}{\|\mathbf{v}\|}$, the $\mathbf{W}$ in (3) is related with the distance. Meanwhile, the orthonormal representation is similar to the quaternion form introduced in [29].

The endpoints of the 3D line in the above articles are uncertain because these spatial positions of 3D lines need to be solved by image features through the principle of

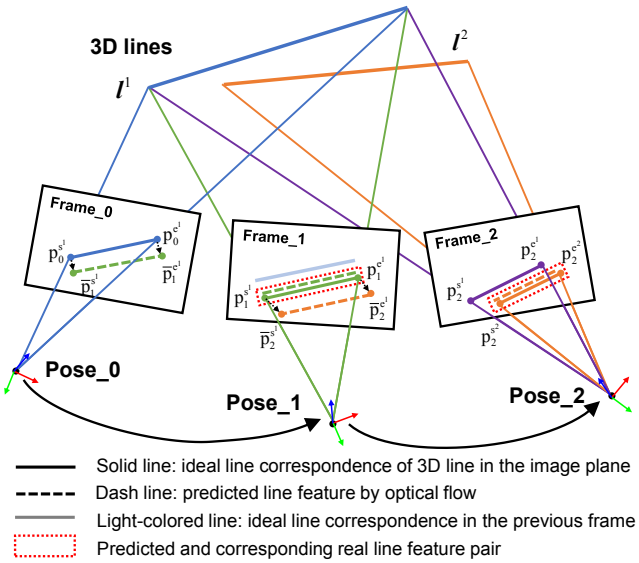

Fig. 4: The line tracking strategy. The Frame_0 to Frame_1 represents the line l^1 in space being successfully tracked through the optical flow of the two endpoints. Frame_1 to Frame_2 demonstrates the failure of line tracing.

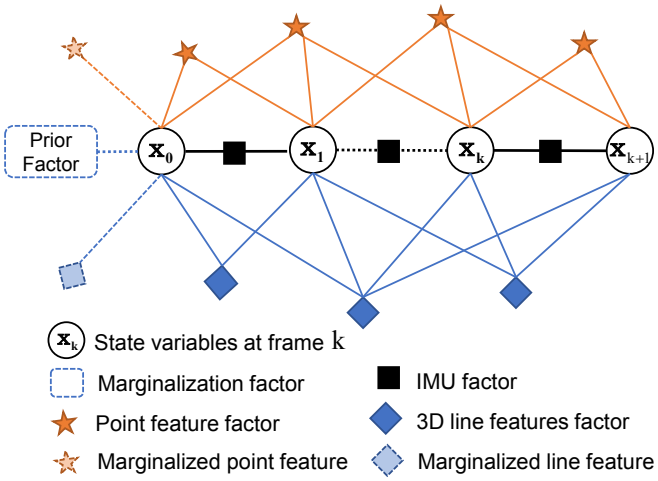

Fig. 5: Factor graph structure in the proposed system. The different symbols indicate the prior, IMU, point, and line constraints used in the optimization problem.

multiple-view geometry. However, 3D lines may not fully project into images, which means that the endpoints of the same 3D line are not uniform in different images, and thus no definite 3D endpoints can be obtained. Therefore, the above scenario cannot use endpoints to represent spatial lines. In contrast, 3D lines in this paper are prior information and their endpoints have explicit locations, so we could simply and directly express 3D lines using the two endpoints.

B. Line Matching & Tracking Strategy

In terms of line matching, a 3D line $\boldsymbol{l}^w$ with two endpoints $\mathbf{P_s}, \mathbf{P_e}$ is projected into the camera plane based on the initial guess of the camera pose (cropping the effective part of the line belonging to the camera FoV if necessary) to get a 2D projected line $\hat{l}^c$ with endpoints $\hat{\mathbf{p}}_\mathbf{s}$, $\hat{\mathbf{p}}_\mathbf{e}$, as shown in Fig. 3 (b). The line $\hat{l}^c$ needs to be matched with the detected line l^c in the image based on the defined three similarity criteria. The details of the criteria are displayed in Fig. 3 (b). The first is the distance between the two lines as shown by the solid red line in Fig. 3 (b). The second is the angle θ between two lines. The third is the ratio of the overlap of the two lines, which is demonstrated by the red dashed line in Fig. 3 (b).

Due to the aforementioned cross-modality differences among a point cloud and images, it is difficult to have consistent feature descriptors between 3D spatial lines and 2D image lines. However, relying only on the matching criteria mentioned above without descriptors implies a high probability of outliers in the matching between projected and detected lines. To address this issue, this paper proposes a fast line feature tracking strategy based on the Lucas-Kanade (LK) optical flow method [30] that does not require consistent descriptors while satisfying the real-time performance of operations. The detailed process can be illustrated in Fig. 4.

Assuming a line $\boldsymbol{l}^1$ in space is captured by a camera, and its ideal matching pair on Frame_0 is represented by $\overrightarrow{\mathbf{p}_0^{s^1}\mathbf{p}_0^{e^1}}$, the $\mathbf{p}_0^{s^1}$ and $\mathbf{p}_0^{e^1}$ are two endpoints. Rather than using the conventional line band descriptor (LBD) [31] to track the line feature in images, the proposed method expresses the line feature in terms of its endpoints, which are directly tracked by computing the optical flow instead of processing the whole line segment. For example, the endpoints $\overline{\mathbf{p}}_1^{s^1}$ and $\overline{\mathbf{p}}_1^{e^1}$ on Frame_0 are the predicted position of the points $\mathbf{p}_0^{s^1}$ and $\mathbf{p}_0^{e^1}$ in the next frame by optical flow tracking, which is represented by the same green dashed line in Frame_1. However, there may be errors between the predicted line and the real detected line, so the detected line $\overrightarrow{\mathbf{p}_1^{s^1}\mathbf{p}_1^{e^1}}$ nearest to the predicted green dashed line needs to be considered as the final tracked line segment, which is denoted by the green solid line in Fig. 4. At this point, we also match the line $\overrightarrow{\mathbf{p}_1^{s^1}\mathbf{p}_1^{e^1}}$ with 3D lines in Frame_1, and if it corresponds to the same 3D line as line $\overrightarrow{\mathbf{p}_0^{s^1}\mathbf{p}_0^{e^1}}$ on Frame_0, then this is a set of credible matching pair.

On the contrary, Frame_1 to Frame_2 in Fig. 4 demonstrates an unreliable matching case. Similarly, the endpoints $\overline{\mathbf{p}}_2^{s^1}$ and $\overline{\mathbf{p}}_2^{e^1}$ in Frame_1 are the predicted positions on the next frame by optical flow tracking, and the nearest detected line on Frame_2 is $\overrightarrow{\mathbf{p}_2^{s^2}\mathbf{p}_2^{e^2}}$. However, the 3D line in space that matches with line $\overrightarrow{\mathbf{p}_2^{s^2}\mathbf{p}_2^{e^2}}$ is $\boldsymbol{l}^2$ rather than $\boldsymbol{l}^1$. In a word, the line features that are continuously tracked in images have different line correspondences in 3D space, in which case we consider this set of line correspondences unreliable and will remove this set of line matching pairs from observations.

C. Optimization Model

Drawing on the conventional VIO architecture, we utilize IMU data, point feature, line feature, and marginalized prior information as factors to construct an optimization model and solve the state estimation problem by the least square

method. Fig. 5 displays the factor graph of the proposed system. Inside, the prior factor refers to the information marginalized from old frames. Assuming that the different measurements are independent of each other and the observation noise obeys a Gaussian distribution with zero mean and covariance is $\mathbf{Q}$. By minimizing the square sum of all observations residuals and prior factors, the optimization model can be written as

$$\mathcal{X}^* = \min_{\mathcal{X}} \left\{ \|\mathbf{r}_p - \mathbf{H}_p\mathcal{X}\|^2 + \sum_{k\in\mathcal{B}} \|\mathbf{r}_b(\mathbf{z}_b,\mathcal{X})\|^2_{\mathbf{Q}_b} + \sum_{k\in\mathcal{C}} \|\mathbf{r}_c(\mathbf{z}_c,\mathcal{X})\|^2_{\mathbf{Q}_c} + \sum_{k\in\mathcal{L}} \|\mathbf{r}_l(\mathbf{z}_l,\mathcal{X})\|^2_{\mathbf{Q}_l} \right\}. \tag{4}$$

Where, four residual terms $\mathbf{r}_x$ correspond to the prior, IMU, point feature, and line feature factor, respectively (The prior, IMU, and point feature factor can be referred in paper [9]). $\mathbf{H}_p$ is the Jacobian matrix of the prior factor. $\mathcal{X}$ is the state variables, $\mathbf{z}$ is the observations. Next, we will focus on the construction of the cost function based on line features.

The cost function of line features is generally expressed by minimizing the distance from endpoints to the matched line. In this paper, the error relationship between lines is further advanced by transforming the point-to-line distance into a point-to-point reprojection error. As shown in Fig. 3 (b), a 3D line $\boldsymbol{l}^w$ with known positions of endpoints is projected into an image as $\hat{l}^c$, and the pixel location of endpoints on images can be calculated by

$$s\hat{\mathbf{p}} = s\begin{bmatrix}\hat{\mu}\\ \hat{\nu}\\ 1\end{bmatrix} = \mathbf{K}\cdot\mathbf{T}\cdot\mathbf{P}, \tag{5}$$

where, s is the scale, $\hat{\mathbf{p}}(\hat{\mu},\hat{\nu})$ is the predicted pixel position in a image, $\mathbf{P}(X,Y,Z)$ is the endpoint position on space, $\mathbf{K}$ is the intrinsic matrix of camera, $\mathbf{T} = [\mathbf{R}^c_w|\mathbf{t}^c_w]$ means the transformation from the world frame to the image plane, and can be decomposed as

$$\begin{aligned}\mathbf{R}^c_w &= \mathbf{R}^c_b\mathbf{R}^b_w\\ \mathbf{t}^c_w &= \mathbf{R}^c_b\mathbf{t}^b_w + \mathbf{t}^c_b.\end{aligned} \tag{6}$$

Then, the distance between $\hat{l}^c$ and its matching line l^c on image can be converted to the distance between its endpoints $\hat{\mathbf{p}}_s$, $\hat{\mathbf{p}}_e$ and its vertical foot points $\mathbf{p}^f_s$, $\mathbf{p}^f_e$ on line l^c, as the red lines shown in Fig. 3. And the error distance function can be written as

$$\mathbf{e} = \mathbf{p}^f - \hat{\mathbf{p}} = (\mu^f - \hat{\mu})^2 + (\nu^f - \hat{\nu})^2. \tag{7}$$

Where, the vertical foot points can be computed by

$$\mu^f = \frac{\mathbf{B}^2\hat{\mu} - \mathbf{A}\mathbf{B}\hat{\nu} - \mathbf{A}\mathbf{C}}{\mathbf{A}^2+\mathbf{B}^2}, \nu^f = \frac{\mathbf{A}^2\hat{\nu} - \mathbf{A}\mathbf{B}\hat{\mu} - \mathbf{B}\mathbf{C}}{\mathbf{A}^2+\mathbf{B}^2}. \tag{8}$$

The parameters $\mathbf{A},\mathbf{B},\mathbf{C}$ in (8) are obtained from the general form of the line l^c on 2D image:

$$\mathbf{A}x + \mathbf{B}y + \mathbf{C} = 0, \tag{9}$$

TABLE I: RMSE [35] of ATE (m) on EuRoC dataset

Sequence	VINS-Mono [9]	LCL-C [24]	TCL
V1_01_esay	0.078	0.164	**0.068**
V1_02_medium	0.110	0.152	**0.084**
V1_03_difficult	0.189	0.217	0.182
V2_01_esay	0.096	0.192	**0.068**
V2_02_medium	0.167	0.281	**0.108**
V2_03_difficult	0.253	0.341	0.254

which can be calculated by the two endpoints $\mathbf{p}_s$, $\mathbf{p}_e$ on l^c.

It can be seen from (8), the calculation of foot point $\mathbf{p}^f$ is related to the prediction $\hat{\mathbf{p}}$, while $\hat{\mathbf{p}}$ is obtained based on state variables (see from equation (5)). Therefore, if a line feature l is projected into frame k, the residual based on this line feature on (4) can be expressed as

$$\begin{aligned}\mathbf{r}_l(\mathbf{z}_l,\mathcal{X}) &= \mathbf{p}^f_l - \hat{\mathbf{p}}_l\\ &= \mathbf{p}^f_l(\mu^f(\mathbf{T}^k_w),\nu^f(\mathbf{T}^k_w)) - \frac{1}{s_l}(\mathbf{K}\left(\mathbf{T}^k_w\right)\mathbf{P}_l)^{\cdot}\end{aligned} \tag{10}$$

It is known that the measurements and the predictions in (10) are associated with the state variables. After the optimization model is constructed, the nonlinear least squares problem can be solved by Ceres Solver [32].

IV. Experiment Results

The proposed system is evaluated on the public EuRoC MAV (Micro Aerial Vehicle) [33] and CARLA Simulator dataset [34]. We compare the proposed Tightly Coupled prior Line feature-based VIO (TCL) method against VINS-Mono without loop closure [9] (since the real-time estimation requirement) and a Loosely-Coupled prior Line feature aided VIO based on a Conventional optimization model (LCL-C) [24]. These comparison algorithms are all based on the monocular and IMU localization framework. All experiments are conducted on a PC with Inter-Core i9-12900K and 32 GB memory.

A. UAV Indoor Dataset Experiment

The UAV indoor experiment is conducted on the EuRoC MAV dataset. The trajectory datasets and prior 3D point cloud maps are collected by stereo cameras (Aptina MT9V034, WVGA monochrome), IMU (ADIS16448, 200 Hz), Vicon, and Leica MS50, which includes two indoor scenes with three different fight trajectories. The root mean squared error (RMSE) of absolute trajectory error (ATE) [35] is used to validate the performance of localization accuracy. The results of the proposed method and comparison methods are shown in Table I. It can be seen that the proposed method outperforms others in the easy and medium scenarios, but does not significantly improve accuracy in difficult scenarios. Firstly, the improvement in accuracy to some extent indicates the effectiveness of the proposed scheme. Nevertheless, it is rarely useful when facing complex UAV large maneuvers. This is mainly because the tightly coupled form has difficulty

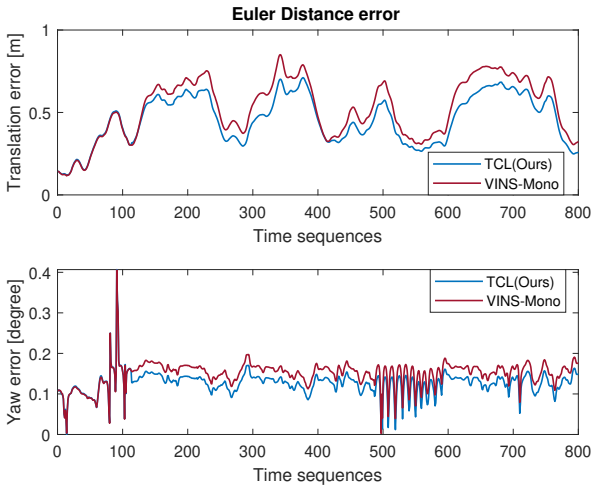

Fig. 6: The Euler distance error on translation and yaw (the four unobservable directions: x, y, z, and yaw) based on the V2_02_medium dataset.

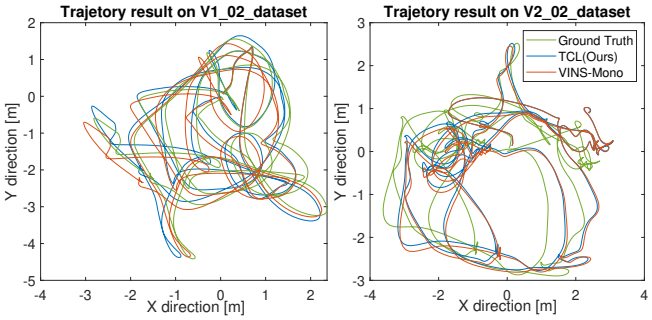

Fig. 7: The trajectory results on the EuRoC V1_02_medium and V2_02_medium dataset.

in providing suitable initial guesses for cross-modality line matching in large maneuver scenarios, thus making the constraint ineffective. Meanwhile, this issue is an important part of our follow-up work.

More comparison details about positioning accuracy are presented in Fig. 6. The top figure shows the translation error and the bottom figure is the yaw rotation error, which we use the direct Euler distance to represent. The four directions x, y, z, and yaw are the unobservable directions of the VIO-based localization system, which are also the main sources of positioning errors [29], [36]. In addition, the trajectories of EuRoC dataset V1_02_medium and V2_02_medium are illustrated in Fig. 7. It can be seen that the proposed method has better performance than VINS-Mono.

B. UGV Outdoor Simulation Experiment

The UGV (Unmanned Ground Vehicles) outdoor dataset is collected on the CARLA simulator, which is an open-source simulator for autonomous driving research. The simulation platform supports flexible specification of sensor suites, environmental conditions, full control of all static and dynamic actors, maps generation, and much more [34]. Based on this powerful simulator, we simulated a vehicle equipped with

Fig. 8: The simulated urban environment on CARLA.

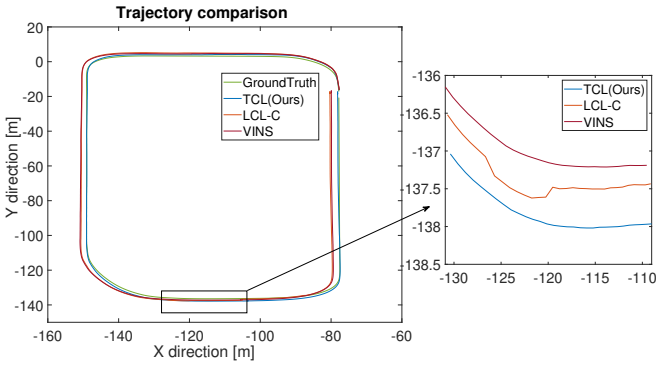

Fig. 9: Trajectory comparison results on the CARLA simulator and a close-up section of the trajectory.

high-precision LiDAR (64 channels), IMU (100 Hz), and cameras (1382x512) to collect data in an urban environment (one of the scenes is shown in Fig. 8) and construct prior map according to the ground truth. By extracting 3D lines from this point cloud map, we apply these prior features to the proposed system to improve the localization accuracy of the conventional VIO (see Fig. 1).

Figure 9 demonstrates the trajectory results of different algorithms on the CARLA simulation dataset, and it can be seen that our trajectory result is closest to the ground truth. Meanwhile, the right subplot on Fig. 9 shows the local details of the trajectory, from which it can be seen that the position estimation results in the loosely coupled mode have more visible fluctuations. In contrast, the trajectory results in the tightly coupled mode are smoother, which verifies the opinion mentioned above and is one of the advantages of the tightly coupled state estimator.

Table II gives the RMSE of ATE for different methods

TABLE II: RMSE [35] of ATE (m) on CARLA dataset

Length	VINS-Mono [9]	LCL-C [24]	TCL
80m	0.603	0.596	**0.235**
160m	0.622	0.675	**0.353**
240m	0.988	1.219	**0.905**
320m	1.032	1.229	**0.997**
400m	1.102	1.269	**1.015**
480m	1.500	1.548	**1.379**

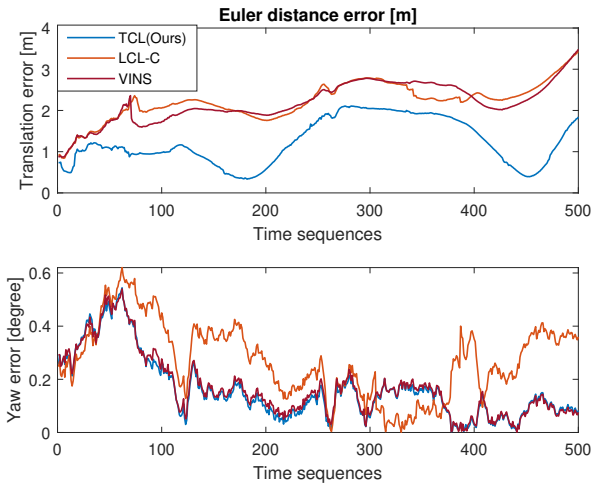

Fig. 10: The Euler distance error on translation and yaw (the four unobservable directions: x, y, z, and yaw) based on the CARLA dataset.

with gradually increasing driving length. In autonomous driving application scenarios, the proposed method outperforms comparative methods in terms of localization accuracy at various driving lengths. The translation error and yaw rotation error are shown in Fig. 10. It also can be seen that our method has a greater improvement in positioning accuracy. Compared with UAV flight, the proposed method is more effective in the self-driving scenario. On the one hand, the motion of self-driving cars is more stable and the large maneuvering motion is less than that of UAVs. In general, the proposed method is more suitable for autonomous driving scenarios because it can provide better initial guesses for the cross-modality state estimation in the tightly coupled form.

V. Conclusion

This paper presents a prior line feature-aided visual-inertial localization system, which integrates observations from the prior point cloud map, monocular, and IMU by a tightly coupled nonlinear optimization module. We represent the line segments by two endpoints, and by using this simple and straightforward expression, a new line constraint-based optimization model is proposed. Meanwhile, this paper enhances the reliability of cross-modality matching by a simple and fast line feature tracking strategy to monitor line correspondences. The indoor UAV and outdoor self-driving comparisons show the effectiveness of the proposed method, especially in the autonomous driving scenario.

In the future, we will analyze the degeneration phenomena of line feature constraint and make full use of prior map information to improve the effectiveness of cross-modality feature matching under complex motions. In addition, the prior map-based state estimation framework always faces the problem of initialization for the system as the VINS and the prior map are from different coordinate systems. The utilization of the GNSS signals and vehicle-road cooperative systems can be a promising solution which will be the future work of this paper. It can also be interesting to explore the possibility of estimating the initialization parameters, for example, the extrinsic parameters between the VINS local frame and the global map frame, as additional unknown variables during the state estimation.

References

[1] N. Stannartz, J.-L. Liang, M. Waldner, and T. Bertram, "Semantic landmark-based hd map localization using sliding window max-mixture factor graphs," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 106–113.

[2] Y. Wang, X. Wen, L. Yin, C. Xu, Y. Cao, and F. Gao, "Certifiably optimal mutual localization with anonymous bearing measurements," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9374–9381, 2022.

[3] E. D. Kaplan and C. Hegarty, *Understanding GPS/GNSS: principles and applications*. Artech house, 2017.

[4] X. Bai, W. Wen, G. Zhang, H.-F. Ng, and L.-T. Hsu, "Gnss outliers mitigation in urban areas using sparse estimation based on factor graph optimization," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 197–202.

[5] H.-F. Ng, G. Zhang, Y. Luo, and L.-T. Hsu, "Urban positioning: 3d mapping-aided gnss using dual-frequency pseudorange measurements from smartphones," *Navigation*, vol. 68, no. 4, pp. 727–749, 2021.

[6] P. D. Groves and M. Adjrad, "Performance assessment of 3d-mapping–aided gnss part 1: Algorithms, user equipment, and review," *Navigation*, vol. 66, no. 2, pp. 341–362, 2019.

[7] W. Wen, G. Zhang, and L.-T. Hsu, "Object-detection-aided gnss and its integration with lidar in highly urbanized areas," *IEEE Intelligent Transportation Systems Magazine*, vol. 12, no. 3, pp. 53–69, 2020.

[8] W. W. Wen and L.-T. Hsu, "3d lidar aided gnss nlos mitigation in urban canyons," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18 224–18 236, 2022.

[9] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[10] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[11] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[12] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.

[13] T. Qin, Y. Zheng, T. Chen, Y. Chen, and Q. Su, "A light-weight semantic map for visual localization towards autonomous driving," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 11 248–11 254.

[14] H. Ye, H. Huang, and M. Liu, "Monocular direct sparse localization in a prior 3d surfel map," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 8892–8898.

[15] K. Yabuuchi, D. R. Wong, T. Ishita, Y. Kitsukawa, and S. Kato, "Visual localization for autonomous driving using pre-built point cloud maps," in *2021 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2021, pp. 913–919.

[16] M. Brown, D. Windridge, and J.-Y. Guillemaut, "A family of globally optimal branch-and-bound algorithms for 2d–3d correspondence-free registration," *Pattern Recognition*, vol. 93, pp. 36–54, 2019.

[17] R. W. Wolcott and R. M. Eustice, "Visual localization within lidar maps for automated urban driving," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 176–183.

[18] Y. Lu, J. Huang, Y.-T. Chen, and B. Heisele, "Monocular localization in urban environments using road markings," in *2017 IEEE intelligent vehicles symposium (IV)*. IEEE, 2017, pp. 468–474.

[19] G. Borgefors, "Hierarchical chamfer matching: A parametric edge matching algorithm," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 10, no. 6, pp. 849–865, 1988.

[20] H. Huang, H. Ye, Y. Sun, and M. Liu, "Gmmloc: Structure consistent visual localization with gaussian mixture models," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5043–5050, 2020.

[21] T. Caselitz, B. Steder, M. Ruhnke, and W. Burgard, "Monocular camera localization in 3d lidar maps," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1926–1931.

[22] X. Zuo, P. Geneva, Y. Yang, W. Ye, Y. Liu, and G. Huang, "Visual-inertial localization with prior lidar map constraints," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3394–3401, 2019.

[23] B. Huhle, M. Magnusson, W. Straßer, and A. J. Lilienthal, "Registration of colored 3d point clouds with a kernel-based extension to the normal distributions transform," in *2008 IEEE international conference on robotics and automation*. IEEE, 2008, pp. 4025–4030.

[24] H. Yu, W. Zhen, W. Yang, J. Zhang, and S. Scherer, "Monocular camera localization in prior lidar maps with 2d-3d line correspondences," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4588–4594.

[25] X. Zheng, W. Wen, and L.-T. Hsu, "Safety-quantifiable line feature-based monocular visual localization with 3d prior map," *arXiv preprint arXiv:2211.15127*, 2022.

[26] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on information theory*, vol. 47, no. 2, pp. 498–519, 2001.

[27] 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, p. 1159, 2018.

[28] A. Bartoli and P. Sturm, "Structure-from-motion using lines: Representation, triangulation, and bundle adjustment," *Computer vision and image understanding*, vol. 100, no. 3, pp. 416–441, 2005.

[29] Y. Yang and G. Huang, "Observability analysis of aided ins with heterogeneous features of points, lines, and planes," *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 1399–1418, 2019.

[30] J.-Y. Bouguet *et al.*, "Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm," *Intel corporation*, vol. 5, no. 1-10, p. 4, 2001.

[31] L. Zhang and R. Koch, "An efficient and robust line segment matching approach based on lbd descriptor and pairwise geometric consistency," *Journal of visual communication and image representation*, vol. 24, no. 7, pp. 794–805, 2013.

[32] S. Agarwal and K. Mierle, "Ceres solver: Tutorial & reference," *Google Inc*, vol. 2, no. 72, p. 8, 2012.

[33] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[34] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[35] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2012, pp. 573–580.

[36] S. Cao, X. Lu, and S. Shen, "Gvins: Tightly coupled gnss–visual–inertial fusion for smooth and consistent state estimation," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2004–2021, 2022.