This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

Tightly-coupled Visual/Inertial/Map Integration with Observability Analysis for Reliable Localization of Intelligent Vehicles

Xi Zheng, Weisong Wen*, Li-Ta Hsu

***Abstract*—Reliable and cost-effective localization is of great importance for the realization of intelligent vehicles (IV) in complex scenes. The visual-inertial odometry (VIO) can provide high-frequency position estimation over time but is subjected to drift over time. Recently developed map-aided VIO opens a new window for drift-free visual localization but the existing loosely coupled integration fails to fully explore the complementariness of all the raw measurements. Moreover, the observability of the existing map-aided VIO is still to be investigated, which is of great importance for the safety-critical IV. To fill these gaps, in this article, we propose a factor graph-based state estimator that tightly couples a 3D lightweight prior line map with a VIO system and rigorous observability analysis. In particular, for the cross-modality matching between 3D prior maps and 2D images, we first utilize the geometric line structure coexisting in the 3D map and 2D image to build the line feature association model. More importantly, an efficient line-tracking strategy is designed to reject the potential line feature-matching outliers. Meanwhile, a new line feature-based cost model is proposed as a constraint for factor graph optimization with proof of the rationality behind this model. Moreover, we also analyze the observability of the proposed prior line feature-aided VIO system for the first time, the result shows that x, y, and z three global translations are observable and the system only has one unobservable direction theoretically, i.e. the yaw angle around the gravity vector. The proposed system is evaluated on both simulation outdoor and real-world indoor environments, and the results demonstrate the effectiveness of our methods. To benefit the research community, we open-sourced the dataset with detailed line labeling by https://github.com/ZHENGXi-git/TC-VIML.**

***Index Terms*—Localization, visual inertial odometry (VIO), 3D prior map, line feature, observability analysis, intelligent vehicles.**

I. Introduction

Accurate and cost-effective localization is the fundamental function of intelligent vehicles to understand their ego-motion and is the key to implementing the following planning and control functions [1, 2, 3]. Global Navigation Satellite System (GNSS) is a widely used technology that provides accurate positioning to users worldwide [4]. However, GNSS signals can be blocked or reflected by tall buildings, dense tree canopies, and other obstacles, in which, the positioning accuracy could be significantly degraded [5, 6]. With the help of sensor integration, such as the light detection and ranging (LiDAR) and inertial measurement units (IMU) [7, 8, 9], the positioning accuracy can be largely improved. However, the high cost and weight of LiDAR limit its massive deployments. Different from the existing LiDAR-based solution, the camera is cost-effective and readily available. As a result, lots of efforts were devoted to exploring the potential of visual-based localization accuracy. The integration of the visual and IMU, namely VIO, is one of the most popular solutions to provide positioning for autonomous systems [10, 11, 12, 13]. However, Visual-based solutions also have limitations, such as susceptibility to low-light conditions, the need for sufficient visual features, and challenges in environments with featureless or dynamic surroundings [14].

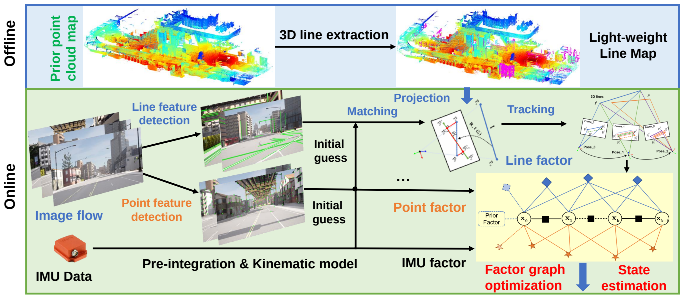


Fig. 1: The flowchart of the proposed framework. The 3D lines in the prior map are detected offline, which are matched with the online detected 2D lines. After feature matching and tracking, the line correspondences, IMU measurements, and point feature pairs are integrated as constraints for factor graph optimization in a tightly coupled form.

Besides, prior map-aided VIO opens a new window for localization. In particular, Prior map-aided VIO aligns 2D visual features with known 3D point cloud data for localization, leveraging the advantages of cameras and prior maps. The map is typically developed based on high-precision devices including geodetic level GNSS receivers, LiDAR, and high-accuracy inertial navigation systems, to create a dense point cloud map in advance [15, 16]. Through offline preprocessing, valuable map-related key features, such as lane lines are extracted as prior information for real-time state estimation.

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

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

During online localization, the prior map is loaded and low-cost cameras are utilized to obtain colorful image features, which are registered with the prior map for precise localization. By fusing data from online cameras with the known map, systems can mitigate errors and correct the drift of the existing VIO system [15, 17, 18]. Based on the advantages mentioned above, this approach has been gaining more and more attention.

Nevertheless, this method still presents unresolved challenges that require further exploration. For instance, the heavy prior map can consume a significant amount of storage. The accuracy and consistency of the prior map are also important for reliable localization. In response to this challenge, researchers have introduced High-Definition (HD) maps as a potential solution, which provide a higher level of precision, including information about the road geometry, lane markings, and traffic signs [19, 20]. Meanwhile, integrating online camera data with the prior point cloud map can also be challenging for localization because of the cross-modality alignment problem between different sensors and data sources [21].

Some papers focus on the mutual geometric line features existing in 2D images and 3D maps for cross-modality registration. In the study presented by Yu *et al.* [22], 2D lines from images and 3D lines from prior maps are detected and matched for line correspondences based on the initial guess provided by the VINS-Mono [10]. Our previous work developed a loose-coupled integration of the VIO and map-matching, where the state estimation from the VIO is integrated with the line-feature-based matching constraint using the factor graph optimization [23]. However, the loosely coupled integration in [23] fails to fully explore the complementariness of the raw measurements across multiple frames. The outliers in visual line matching can degrade the performance of the integration in complex scenes with repetitive textures. Moreover, the degeneration of the map-aided VIO commonly exists in highly complex urban scenes for IV due to insufficient line features. A rigorous formulation of the system observability analysis is expected to account for the potential degeneration of the map-aided VIO.

In this paper, we proposed a tightly coupled framework that associated the classic VIO optimization factors with the cross-modal line features-based constraints for pose estimation, which is shown in Fig. 1. In this framework, the 3D lines are extracted from the prior map offline and then the original point cloud map can be condensed to a lightweight line map containing building edges and road segment markings. The cross-modal line correspondences are obtained by line matching and a proposed line tracking strategy for outliers rejection. The novel line feature-based cost model proposed in our previous work [23] is tightly coupled with the IMU pre-integration factor and point feature factor for state estimation. Meanwhile, we justify the rationality of this novel model in this paper. Besides, we analyze the observability of the proposed system to better understand the internal relationships between state variables and system measurements. To the best of our knowledge, we are the first group that analyze the observability of the prior map-aided VIO system.

The contributions of this paper are listed as follows,

1) We introduce a novel line feature-based optimization constraint and justify it, in which the conventional line feature error function is transformed from minimizing the line distance into the point re-projection error. Moreover, We develop an optical flow-related fast line feature tracking method to remove correspondence outliers and improve the reliability of cross-modality matching.
2) The IMU pre-integration factors, visual point feature factor, and prior line feature factor are constructed into a tightly coupled optimization structure to improve the accuracy of state estimation.
3) We analyze for the first time the observability of the prior map-aided visual localization framework. The proposed system is constructed into a nonlinear affine model and the observability properties are conducted by finite Lie derivatives thought basis element factorization. The results show that the yaw angle around gravity is the only unobservable direction of the proposed system.
4) The proposed framework is evaluated in both simulation and real-world experiments. The results demonstrate the effectiveness of the proposed method.

II. Related Work

A. Prior Map-aided Localization

In terms of the HD-maps, Bétaille *et al.* [24] proposed a concept of enhanced maps in which road lanes are represented and topologically connected into a long lane by Kalmen filter-based estimation. In [19], road markings including lane lines and symbols are extracted from mobile laser scanning point clouds by edge detection and conditional Euclidean clustering algorithms to create HD maps. A deep learning architecture named RoadStarNet was presented to detect and segment road line markings and express them in a graph-based format [25]. However, these works solely focus on the construction of HD maps, overlooking their application in localization. It is difficult to assess their ultimate effectiveness in improving localization accuracy.

Regarding the cross-modality alignment problem, some methods extract and fuse consistent features from different data and then get corresponding constraints for state estimation. For example, The paper [26] manually labeled road markings within a sparse point cloud map, which were registered with the detected edges in online images by Chamfer matching to build epipolar geometry constraints for pose estimation. Ye *et al.* [17] rendered vertex and normal maps from prior 3D surfel maps by principal component analysis, which were projected into depth maps to provide the depth information for tracked direct point features in online images. The 6-degree of freedom (DoF) poses were estimated by optimizing the direct photometric error constraints. Huang *et al.* [18] modeled dense prior 3D maps by Gaussian mixture model, which were associated with online visual measurements to build hybrid structure factors for joint pose optimization.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

Others chose to use the point registration methods for pose estimation. Zuo *et al.* [27] utilized a stereo visual-inertial odometry (VIO) based on a multi-state constraint Kalmen filter to construct a semi-dense point cloud, which was registered with the prior LiDAR map by normal distribution transform-related method [28]. In [29], the BA method was used to obtain initial camera poses and sparse point clouds with unknown scales. The sparse cloud was aligned with a prior LiDAR map for pose estimation by a 7-DoF iterative closest point (ICP) scheme [30]. Qin *et al.* [15] applied a segmentation network to extract semantic road markings among prior LiDAR maps to build a lightweight HD map, which was matched with sparse point cloud generated from online visual features by ICP method.

B. Observability Analysis of Visual-related Localization

Observability plays a crucial role in state estimation, which refers to the ability to estimate the states of a dynamic system based on its measurements, ensuring that the system's state can be effectively understood [31, 32]. In other words, the state variable corresponding to an unobservable direction will have indistinguishable trajectories based on measurements of system [33]. Observability analysis can provide reference information for the performance of state estimation and infer degenerate motions that can lead to additional unobservable directions, significantly impacting the performance of state estimation [34, 35]. In addition, observability analysis is used for online self-calibration of camera-IMU modules to guarantee the calibration parameters are observable [36, 37, 38].

Specifically, Martinelli [39] derived closed-form solutions of VINS that analytically expressed IMU biases, 3D velocity, and absolute roll and pitch angles are observable. Hesch *et al.* [40] introduced basis elements in observability analysis to factorize the observability matrix. The factorized matrix is used to determine the unobservable mode of nonlinear VIO systems by computing a finite number of Lie derivatives, which showed that the monocular VINS has four unobservable DoFs, corresponding to the three global translations and global yaw angle around the gravity vector. Guo *et al.* [41] studied the observability properties of an IMU-RGBD camera navigation system including point and plane features based on Hesch's method. Yang *et al.* [35] conducted a thorough observability analysis with different geometric features including points, lines, planes, and their combinations for linearized VINS.

In summary, observability analysis allows us to gain a deeper understanding of a system. By analyzing the unobservable directions, we can know the restriction of state estimation performance. It is highly expected to provide a rigorous formulation for the map-aided VIO for the safety-critical IV localization.

III. System Overview

The notations in this paper are defined as follows. The coordinate systems involved in this paper are the global world frame $(\cdot)^w$, the body frame $(\cdot)^b$ (i.e. IMU frame $(\cdot)^i$), and the camera frame $(\cdot)^c$. The rotation matrix $\mathbf{R}_b^a$ means the rotation from frame b to frame a. The states to be estimated in the localization problem are the transformation $\mathbf{T}_b^w$, including the translation $\mathbf{p}_b^w$ and quaternion rotation $\mathbf{q}_b^w$ (or the rotation matrix $\mathbf{R}_b^w$) from the body frame to the world frame. The other states are the velocity $\mathbf{v}_b^w$, IMU accelerometer bias $\mathbf{b}_a$, and gyroscope bias $\mathbf{b}_g$. Besides, the extrinsic parameter from the camera frame to the body frame is $\mathbf{p}_c^b$ and $\mathbf{q}_c^b$. The inverse depth of each point feature is λ. $\otimes$ is the symbol of the multiplication operation of quaternions.

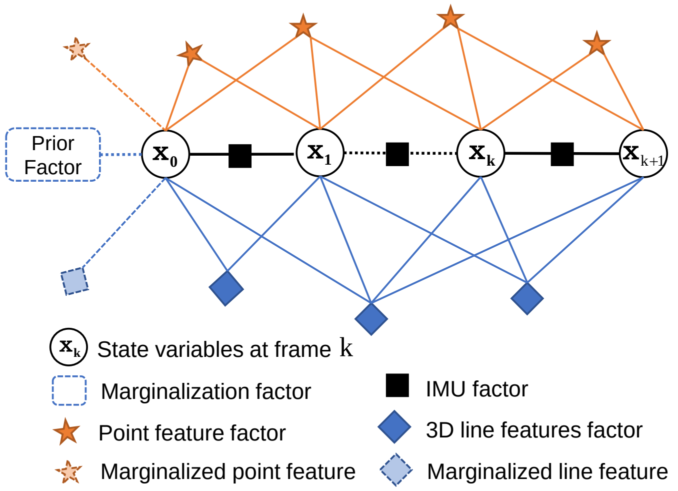


Fig. 2: Factor graph structure in the proposed system. The different symbols indicate the prior, IMU, point, and line constraints used in the optimization problem.

This paper utilizes IMU measurements, visual point features, and line features to construct constraints for nonlinear optimization-based state estimation. To improve stability and accuracy, the proposed method applies the sliding window structure, which can compose multi-frame constraints simultaneously and is the same as the classic VIO system [10]. Based on the sliding window pipeline, the states are defined as

$$\begin{aligned} \mathcal{X} &= \left[\mathbf{x}_k, \ldots, \mathbf{x}_{k+N}, \mathbf{x}_c^b, \lambda_m, \ldots, \lambda_{m+M}\right] \\ \mathbf{x}_i &= \left[\mathbf{p}_{b_i}^w, \mathbf{v}_{b_i}^w, \mathbf{q}_{b_i}^w, \mathbf{b}_a, \mathbf{b}_g\right], i \in [k, k+N], \\ \mathbf{x}_c^b &= \left[\mathbf{p}_c^b, \mathbf{q}_c^b\right] \end{aligned} \tag{1}$$

where, $N+1$ is the number of the frame in sliding window. $M+1$ is the point feature number in this window.

The factor graph structure of the proposed optimization model is displayed in Fig. 2. And the model can be expressed by

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

where, the four cost terms $\mathbf{r}_x$ are the prior factor, IMU measurement factor, point feature-based factor, and line feature-based factor. $\mathbf{z}_{(\cdot)}$ means the different measurements. It should

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

be mentioned that the prior factor comes from the marginalized constraints shown in Fig. 2 and $\mathbf{H}_p$ is the Jacobian matrix of the prior factor. The construction of the other factors are derived in the next section.

IV. State Estimation

A. IMU Pre-integration

For monocular visual localization system, the addition of IMU can provide scale information for vision, and the gyroscope and accelerometer equipped with IMU can deliver high-frequency angular velocity and acceleration data for state estimation. This has profound significance in improving the positioning accuracy of low-cost automatic robot systems. However, the measurement noise and random walk of IMU lead to gradually increasing drift errors in inertial measurements. Meanwhile, as the optimization progresses, the measurement bias of IMU will change, so that the position, velocity and angle increments needs to be recalculated. To avoid repeated integration computations, pre-integration provides an approximate correction for IMU measurements [42]. In VIO system, the low-frequency image keyframes are used as time nodes for state estimation. The pre-integration of IMU combines high-frequency inertial measurements between two keyframes into relative motion constraints in factor graph optimization in this paper [43].

1) IMU-driven System Kinematic Model: The raw measurements of IMU gyroscope $\widetilde{\boldsymbol{\omega}}$ and accelerometer $\widetilde{\mathbf{a}}$ are modeled as

$$\begin{aligned} \widetilde{\boldsymbol{\omega}}(t) &= \boldsymbol{\omega}(t) + \mathbf{b}_g(t) + \mathbf{n}_g(t) \\ \widetilde{\mathbf{a}}(t) &= \mathbf{R}_w^b(\mathbf{a}(t) - \mathbf{g}^w) + \mathbf{b}_a(t) + \mathbf{n}_a(t) \end{aligned}, \tag{3}$$

where, $\boldsymbol{\omega}$ and $\mathbf{a}$ are the acceleration and angular velocity of the body frame respect to the inertial frame. $\mathbf{g}^w = [0, 0, g]^T$ is the Gravity acceleration in the world frame, $\mathbf{b}_g$ and $\mathbf{b}_a$ are the gyroscope and acceleration bias, which are called random walk and are modeled as Gaussian distributions with zero mean and covariance $\boldsymbol{\sigma}_{b_g}^2$ and $\boldsymbol{\sigma}_{b_a}^2$, respectively. The $\mathbf{n}_g$ and $\mathbf{n}_a$ are measurement Gaussian white noise, shown as $\mathbf{n}_g \sim \mathcal{N}(0, \boldsymbol{\sigma}_g^2)$, $\mathbf{n}_a \sim \mathcal{N}(0, \boldsymbol{\sigma}_a^2)$ [10]. The kinematic model can be described as [40]

$$\begin{aligned} \dot{\mathbf{p}}_b^w(t) &= \mathbf{v}_b^w(t) \\ \dot{\mathbf{v}}_b^w(t) &= \mathbf{R}_b^w(t)\mathbf{a}(t) \\ \dot{\mathbf{q}}_b^w(t) &= \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}(t))\mathbf{q}_b^w(t), \\ \dot{\mathbf{b}}_g(t) &= \mathbf{0}_{3\times1} \\ \dot{\mathbf{b}}_a(t) &= \mathbf{0}_{3\times1} \end{aligned} \tag{4}$$

where,

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor\boldsymbol{\omega}\rfloor_\times & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^\top & 0 \end{bmatrix}, \lfloor\boldsymbol{\omega}\rfloor_\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{5}$$

2) IMU pre-intergration: If the IMU body states in frame k is known (position $\mathbf{p}_{b_k}^w$, velocity $\mathbf{v}_{b_k}^w$, and rotation $\mathbf{q}_{b_k}^w$), according to the (3) and (4), these states in frame $k+1$ can be calculated by the following continuous integral formula,

$$\begin{aligned} \mathbf{p}_{b_{k+1}}^w &= \mathbf{p}_{b_k}^w + \mathbf{v}_{b_k}^w \Delta t + \iint_{t\in[t_k,t_{k+1}]} \mathbf{R}_b^w(t)(\widetilde{\mathbf{a}}(t) - \mathbf{b}_a(t) - g^w)dt^2 \\ \mathbf{v}_{b_{k+1}}^w &= \mathbf{v}_{b_k}^w + \int_{t\in[t_k,t_{k+1}]} \mathbf{R}_b^w(t)(\widetilde{\mathbf{a}}(t) - \mathbf{b}_a(t) - g^w)dt \\ \mathbf{q}_{b_{k+1}}^w &= \mathbf{q}_{b_k}^w \otimes \int_{t\in[t_k,t_{k+1}]} \frac{1}{2}\boldsymbol{\Omega}(\widetilde{\boldsymbol{\omega}}(t) - \mathbf{b}_g(t))\mathbf{q}_b^{b_k}(t)dt \end{aligned}, \tag{6}$$

Where Δt is the time different between two frames, $\mathbf{R}_b^w(t)$ means the rotation from body to world frame at time t. It can be seen from (6), when IMU body states at frame k changes because of optimization updates, the states at frame $k+1$ require reintegration. To avoid this computational consumption, (6) is decomposed and simplified to

$$\begin{aligned} \mathbf{p}_{b_{k+1}}^w &= \mathbf{p}_{b_k}^w + \mathbf{v}_{b_k}^w \Delta t - \frac{1}{2}\mathbf{g}^w \Delta t^2 + \mathbf{p}_{b_k}^w \alpha_{b_{k+1}}^{b_k} \\ \mathbf{v}_{b_{k+1}}^w &= \mathbf{v}_{b_k}^w - \mathbf{g}^w \Delta t + \mathbf{p}_{b_k}^w \beta_{b_{k+1}}^{b_k} \\ \mathbf{q}_{b_{k+1}}^w &= \mathbf{q}_{b_k}^w \otimes \gamma_{b_{k+1}}^{b_k} \end{aligned}, \tag{7}$$

among that,

$$\begin{aligned} \alpha_{b_{k+1}}^{b_k} &= \iint_{t\in[t_k,t_{k+1}]} \mathbf{R}_b^{b_k}(t)(\widetilde{\mathbf{a}}(t) - \mathbf{b}_a(t))dt^2 \\ \beta_{b_{k+1}}^{b_k} &= \int_{t\in[t_k,t_{k+1}]} \mathbf{R}_b^{b_k}(t)(\widetilde{\mathbf{a}}(t) - \mathbf{b}_a(t))dt \\ \gamma_{b_{k+1}}^{b_k} &= \int_{t\in[t_k,t_{k+1}]} \frac{1}{2}\boldsymbol{\Omega}(\widetilde{\boldsymbol{\omega}}(t) - \mathbf{b}_g(t))\gamma_t^{b_k}dt \end{aligned}. \tag{8}$$

Where, $\alpha_{b_{k+1}}^{b_k}, \beta_{b_{k+1}}^{b_k}, \gamma_{b_{k+1}}^{b_k}$ are called as IMU pre-integration measurements and can be solved without knowing the IMU body states in frame k [44]. The three new measurements will be used for IMU constraints during state optimizatoin. However, (8) include the state variables $\mathbf{b}_a$ and $\mathbf{b}_g$. To avoid repeat integration operation when $\mathbf{b}_a$ and $\mathbf{b}_g$ are changed during optimization, the solution of (8) is approximated by a first-order expansion [10],

$$\begin{aligned} \alpha_{b_{k+1}}^{b_k} &= \hat{\alpha}_{b_{k+1}}^{b_k} + \mathbf{J}_{b_a}^{\alpha}\delta\mathbf{b}_{a_k} + \mathbf{J}_{b_g}^{\alpha}\delta\mathbf{b}_{g_k} \\ \beta_{b_{k+1}}^{b_k} &= \hat{\beta}_{b_{k+1}}^{b_k} + \mathbf{J}_{b_a}^{\beta}\delta\mathbf{b}_{a_k} + \mathbf{J}_{b_g}^{\beta}\delta\mathbf{b}_{g_k} \\ \gamma_{b_{k+1}}^{b_k} &= \hat{\gamma}_{b_{k+1}}^{b_k} + \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\mathbf{J}_{b_g}^{\gamma}\delta\mathbf{b}_{g_k} \end{bmatrix} \end{aligned}, \tag{9}$$

where $\mathbf{J}_b^a$ is the Jacobian matrices of pre-integrated measurements with respect to bias at time k [44]. So far, the IMU pre-integration is converted to a linear operation, which greatly reduces the complexity of the computation.

Combining the (7) and (9), the measurement residual of IMU preintegration between two consecutive frames b_k and

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

b_{k+1} can be defined as

$$\mathbf{r}_b(\mathbf{z}^{b_k}_{b_{k+1}}, \mathcal{X}) = \begin{bmatrix} \delta\alpha^{b_k}_{b_{k+1}} \\ \delta\beta^{b_k}_{b_{k+1}} \\ \delta\gamma^{b_k}_{b_{k+1}} \\ \delta b_a \\ \delta b_g \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{R}^{b_k}_w(\mathbf{p}^w_{b_{k+1}} - \mathbf{p}^w_{b_k} - \mathbf{v}^w_{b_k}\Delta t + \frac{1}{2}\mathbf{g}^w\Delta t^2) - \hat{\alpha}^{b_k}_{b_{k+1}} \\ \mathbf{R}^{b_k}_w(\mathbf{v}^w_{b_{k+1}} - \mathbf{v}^w_{b_k} + \mathbf{g}^w\Delta t) - \hat{\beta}^{b_k}_{b_{k+1}} \\ 2[(\mathbf{q}^w_{b_k})^{-1} \otimes \mathbf{q}^w_{b_{k+1}} \otimes (\hat{\gamma}^{b_k}_{b_{k+1}})^{-1}] \\ b_{ab_{k+1}} - b_{ab_k} \\ b_{gb_{k+1}} - b_{gb_k} \end{bmatrix} \tag{10}$$

B. Point Feature Measurements Model

This paper retains the point feature as an optimization factor in the VIO system. The factor graph optimization is illustrated in Fig. 2. In terms of point features, the measurement residual is built by the re-projection error. This error means the pixel distance in an image between the projected feature p and the observed feature $\hat{p}$. Assuming the k-th point feature is co-measured in the i-th and j-th frame as $\hat{p}^{c_i}_k(\mu^{c_i}_k, \nu^{c_i}_k)$ and $\hat{p}^{c_j}_k(\mu^{c_j}_k, \nu^{c_j}_k)$, respectively. Its measurement residual model can be written as

$$\begin{aligned} \mathbf{r}_c(\mathbf{z}_c, \mathcal{X}) &= p^{c_j}_k - \hat{p}^{c_j}_k \\ p^{c_j}_k &= \pi\Big(\mathbf{R}^c_b\left(\mathbf{R}^{b_j}_w\left(\mathbf{P}^w_k - \mathbf{p}^w_{b_j}\right) - \mathbf{p}^b_c\right)\Big)' \end{aligned} \tag{11}$$

$$\mathbf{P}^w_k = \begin{bmatrix} \mathbf{X}^w_k \\ \mathbf{Y}^w_k \\ \mathbf{Z}^w_k \end{bmatrix} = \mathbf{R}^w_{b_i}\left(\mathbf{R}^b_c \frac{1}{\lambda_i}\pi^{-1}\left(\begin{bmatrix} \hat{\mu}^{c_i}_k \\ \hat{\nu}^{c_i}_k \end{bmatrix}\right) + \mathbf{p}^b_c\right) + \mathbf{p}^w_{b_i}, \tag{12}$$

where, $\mathbf{P}^w_k$ is the position of the k-th point feature in the world frame, and $\pi(\cdot)$ means the projection function and $\pi^{-1}(\cdot)$ is the back-projection.

C. Line Feature Measurements Model

1) Line Representation: In the visual-based state estimation, the line segments in space are usually represented by the *Plücker* coordinates and orthonormal representation [35, 45, 46]. The former is used for the line initialization and projection, the latter is applied for optimization. In Fig. 3 (a), the *Plücker* coordinate of a 3D line $\boldsymbol{l}^w$ can be expressed by a 6-vector $\mathcal{L} = (\mathbf{n}^\top, \mathbf{v}^\top)^\top$. The $\mathbf{v}$ is the line direction vector, the $\mathbf{n}$ is the normal vector of the plane π including the $\boldsymbol{l}^w$ and origin. If there are two known points $\mathbf{P_s}$, $\mathbf{P_e}$ on $\boldsymbol{l}^w$, the *Plücker* coordinate can be expressed by

$$\begin{bmatrix} \mathbf{n} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{P_s} \times \mathbf{P_e} \\ \mathbf{P_e} - \mathbf{P_s} \end{bmatrix}. \tag{13}$$

However, infinite lines in 3D space have four degree of freedoms (DoFs), so the *Plücker* coordinate has a additional constraint,

$$\mathbf{n}^\top \mathbf{v} = 0 \tag{14}$$

Due to the constraint of the *Plücker* coordinate [45], the orthonormal representation $(\mathbf{U}, \mathbf{W})$ is introduced to process the partial derivative and nonlinear optimization, which can be obtained by the QR decomposition of the *Plücker* coordinate [45],

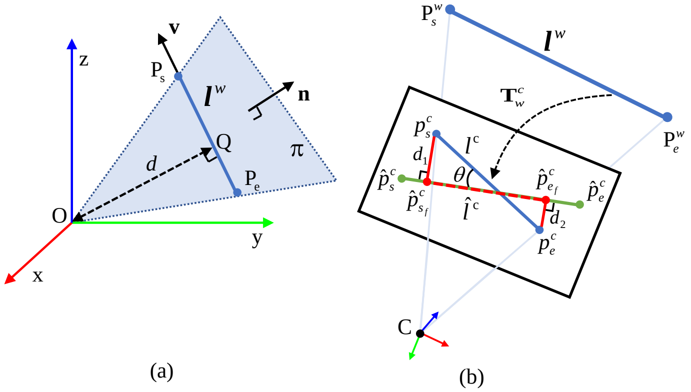
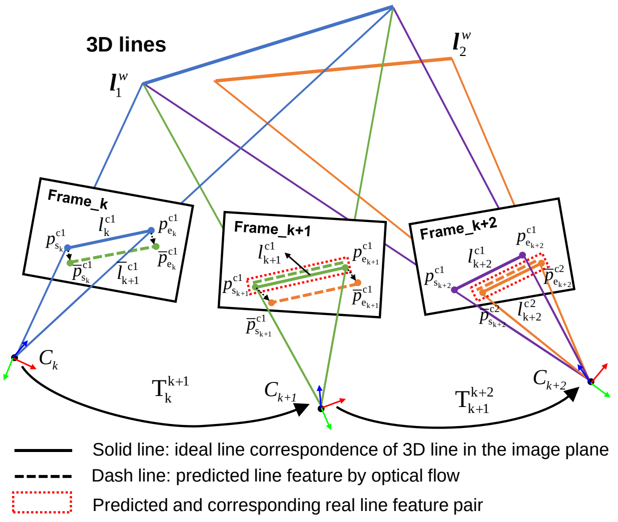


Fig. 3: (a) The line representation methods. (b) The line matching criteria and reprojection error.



3D lines
l_1^w
l_2^w
Frame_k
Frame_k+1
Frame_k+2
C_k
C_k+1
C_k+2
T_k^k+1
T_k+1^k+2
Solid line: ideal line correspondence of 3D line in the image plane
Dash line: predicted line feature by optical flow
Predicted and corresponding real line feature pair

Fig. 4: The line tracking strategy.

$$[\mathbf{n} \mid \mathbf{v}] = \mathbf{U}\begin{bmatrix} \omega_1 & 0 \\ 0 & \omega_2 \\ 0 & 0 \end{bmatrix}, \mathbf{W} = \begin{bmatrix} \omega_1 & -\omega_2 \\ \omega_2 & \omega_1 \end{bmatrix}. \tag{15}$$

Accordingly, the *Plücker* coordinate can be transformed by orthonormal representation,

$$\mathcal{L} = (\omega_1 \mathbf{u}_1^\top, \omega_2 \mathbf{u}_2^\top)^\top, \tag{16}$$

where, $\mathbf{u}_1, \mathbf{u}_2$ are the i-th column of matrix $\mathbf{U}$.

The difference between the line segments used in this article and the above is that the lines in this paper are extracted from prior maps. We can obtain the line segments in 3D space directly without calculating their positions by multi-view geometry principles, which means that the two endpoints of the lines have explicit 3D coordinates rather than infinite. Therefore, this paper uses the two explicit endpoints $\mathbf{P}_s(x_s, y_s, z_s), \mathbf{P}_e(x_e, y_e, z_e)$ to represent the spatial line and performs subsequent operations based on these two endpoints.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

2) Line Matching & Tracking: Before constructing optimization constraints using line segment features, it is necessary to detect and match lines in 2D images and 3D prior maps. First, the line segments in prior map are extracted by a segment-based 3D line detection method [47] and the lines in 2D images are detected by fast line detection (FLD) method [48]. For line matching, 3D lines in prior map are projected onto image plane based on the field of view (FoV) of camera first. Then these 3D lines falling into FoV will be matched with the line features detected in 2D images. Fig. 3 (b) shows the matching criteria. A 3D line $\boldsymbol{l}^w$ is projected into an image plane by the initial pose $\mathbf{T}_w^c$, which is the line l^c containing two endpoints p_s^c, p_e^c. The line l^c is matched with a detected line $\hat{l}^c$ by the line distance: $D = d_1 + d_2$, the angle θ between the two lines, and the overlap shown by the red dash line $(\hat{p}_{s_f}^c, \hat{p}_{e_f}^c)$. As mentioned above, a sliding window structure is used to improve the robustness and accuracy of state estimation. The visual features within the field of view (FoV) of the sliding window are tracked and filtered by the Lucas-Kanade (LK) optical flow method [49] to improve feature reliability.

For the line feature tracking strategy in this paper, we still focus on the two endpoints of lines. By drawing on the point tracking method, the line features introduced in this paper adopt a comparable optical flow-based tracking approach. As shown in Fig. 4, when a detected 2D line l_k^{c1} on frame k is matched with a 3D line $\boldsymbol{l}_1^w$, the two endpoints on l_k^{c1} are tracked by LK optical flow method to predict its position $\bar{l}_{k+1}^{c1}$ on the next keyframe $k+1$. The detected line l_{k+1}^{c1}, which is closest to the predicted line $\bar{l}_{k+1}^{c1}$ on the next frame $k+1$, is taken as the corresponding tracked line feature of l_k^{c1}. Next, if the line l_{k+1}^{c1} has the same matching 3D line $\boldsymbol{l}_1^w$ as l_k^{c1}, then this set of matching pairs is considered reliable. Otherwise, this set of features is removed from observations. Fig. 4 demonstrates the two cases: frame k to $k+1$ shows the case of successful tracking, and frame $k+1$ to $k+2$ is the failed case.

3) Line Feature-based Optimization Model: As shown in Fig. 3, a 3D line $\boldsymbol{l}^w$ with two endpoints $\mathbf{P}_s^w, \mathbf{P}_e^w$ is projected into l^c (called as the predicted line) on the image plane, its endpoints are p_s^c, p_e^c. This projected line l^c is matched with the detected line $\hat{l}^c$ on this image and with endpoints p_s^c, p_e^c have two corresponding foot points $\hat{p}_{s_f}^c, \hat{p}_{e_f}^c$ on $\hat{l}^c$. For the line pair, the error function can be built by the distance between the endpoint and its foot point (taking the point pair $(p_s^c, \hat{p}_{s_f}^c)$ as example),

$$\begin{aligned} \mathbf{r}_l(\mathbf{z}_l, \mathcal{X}) &= p_s^c - \hat{p}_{s_f}^c \\ p_l^c &= \pi\big(\mathbf{R}_b^c\left(\mathbf{R}_w^b\left(\mathbf{P}_s^w - \mathbf{p}_b^w\right) - \mathbf{p}_c^b\right)\big). \\ \hat{p}_{s_f}^c &= f(p_s^c) \end{aligned} \tag{17}$$

where, $f(\cdot)$ is to solve the foot point $\hat{p}_{s_f}^c$ on the matched line $\hat{l}^c$, which is related to the states $\mathbf{R}_b^c$, $\mathbf{p}_b^c$, $\mathbf{R}_w^b$, $\mathbf{p}_w^b$. Firstly, the general equation of line $\hat{l}^c$ on the image is

$$\mathbf{A}\hat{\mu} + \mathbf{B}\hat{\nu} + \mathbf{C} = 0. \tag{18}$$

$(\hat{\mu}, \hat{\nu})$ is the pixel coordinate on the image plane. If the pixel location of the endpoint p_s^c on image is (μ, ν), its foot point $\hat{p}_{s_f}^c$ on $\hat{l}^c$ can be computed by

$$\hat{\mu}_f = \frac{\mathbf{B}^2\mu - \mathbf{A}\mathbf{B}\nu - \mathbf{A}\mathbf{C}}{\mathbf{A}^2 + \mathbf{B}^2}, \hat{\nu}_f = \frac{\mathbf{A}^2\nu - \mathbf{A}\mathbf{B}\mu - \mathbf{B}\mathbf{C}}{\mathbf{A}^2 + \mathbf{B}^2}. \tag{19}$$

So, the function $f(\cdot)$ can be written by

$$\begin{aligned} \hat{p}_{s_f}^c &= \begin{bmatrix} \hat{\mu}_f \\ \hat{\nu}_f \\ 1 \end{bmatrix} = f(p_s^c) = \mathbf{F} \begin{bmatrix} \mu \\ \nu \\ 1 \end{bmatrix} \\ \mathbf{F} &= \frac{1}{\mathbf{A}^2 + \mathbf{B}^2} \begin{bmatrix} \mathbf{B}^2 & -\mathbf{A}\mathbf{B} & -\mathbf{A}\mathbf{C} \\ -\mathbf{A}\mathbf{B} & \mathbf{A}^2 & -\mathbf{B}\mathbf{C} \\ 0 & 0 & \mathbf{A}^2 + \mathbf{B}^2 \end{bmatrix} \end{aligned} \tag{20}$$

4) The Proof of The New Cost Function: By combining formulas (17) and (20), the line feature-based residual can be expressed as

$$\begin{aligned} \mathbf{r}_l(\mathbf{z}_l, \mathcal{X}) &= p_s^c - f(p_s^c) = (\mathbf{I} - \mathbf{F})p_s^c \\ &= \frac{1}{\mathbf{A}^2 + \mathbf{B}^2} \begin{bmatrix} \mathbf{A}^2 & \mathbf{A}\mathbf{B} & \mathbf{A}\mathbf{C} \\ \mathbf{A}\mathbf{B} & \mathbf{B}^2 & \mathbf{B}\mathbf{C} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mu \\ \nu \\ 1 \end{bmatrix}. \end{aligned} \tag{21}$$

In an ideal situation, when the model converges to the global optimal, the residual $\mathbf{r}_l = 0$. Based on the (21), we have

$$\mathbf{A}\mu + \mathbf{B}\nu + \mathbf{C} = 0, \tag{22}$$

which means that the two endpoints on the predicted line l^c satisfy the line equation of the matched line $\hat{l}^c$. This implies that the two lines coincide exactly, which verifies the global optimal convergence condition when the residual is zero.

V. Observability Analysis

During the state estimation process, observability analysis is regularly used to describe whether a system is observable over different state spaces [40, 50, 51]. Unobservability may lead to the non-convergence of the direction of the state to be estimated. Therefore, the properties of the system state estimation can be profoundly understood through observability analysis.

A. Observability Analysis with Lie Derivatives

We first need to construct the proposed system into a nonlinear affine expression as follows [40],

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \displaystyle\sum_{i=1}^{n} \mathbf{f}_i(\mathbf{x})\mathbf{u}_i \\ \mathbf{z}_p = \mathbf{h}_1(\mathbf{x}) \\ \mathbf{z}_l = \mathbf{h}_2(\mathbf{x}) \end{cases} \tag{23}$$

where, $\mathbf{x}$ is the states, $\mathbf{u} = [\widetilde{\boldsymbol{\omega}}, \widetilde{\mathbf{a}}]$ is the IMU input. $\dot{\mathbf{x}}$ is from IMU system kinematic model in (4), the derivative of the extrinsic parameters $\dot{\mathbf{q}}_c^b = \mathbf{0}_{3\times1}, \dot{\mathbf{p}}_c^b = \mathbf{0}_{3\times1}$, and the derivative of the point feature 3D position $\dot{\mathbf{P}}_k^w = \mathbf{0}_{3\times1}$. $\mathbf{z}_p$ is the point feature-based measurement model in (11). $\mathbf{z}_l$ is

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

the line feature-based measurement model in (17). Therefore, the complete model in (23) can be expressed by

$$\begin{aligned} &\dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \mathbf{f}_1(\mathbf{x})\widetilde{\boldsymbol{\omega}} + \mathbf{f}_2(\mathbf{x})\widetilde{\mathbf{a}} \\ \Longrightarrow &\begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{v}} \\ \dot{\mathbf{s}}_1 \\ \dot{\mathbf{b}}_g \\ \dot{\mathbf{b}}_a \\ \dot{\mathbf{p}}_c^b \\ \dot{\mathbf{s}}_2 \\ \dot{\mathbf{P}}_k^w \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{g}^w - \mathbf{R}_b^w \mathbf{b}_a \\ -\mathbf{D}_1 \mathbf{b}_g \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{D}_1 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \end{bmatrix} \widetilde{\boldsymbol{\omega}} + \begin{bmatrix} \mathbf{0}_3 \\ \mathbf{R}_b^w \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \end{bmatrix} \widetilde{\mathbf{a}} \\ &\mathbf{z}_p = \pi\left(\mathbf{R}_b^c\left(\mathbf{R}_w^b\left(\mathbf{P}_k^w - \mathbf{p}_b^w\right) - \mathbf{p}_c^b\right)\right) \\ &\mathbf{z}_l = \pi\left(\mathbf{R}_b^c\left(\mathbf{R}_w^b\left(\mathbf{P}_l^w - \mathbf{p}_b^w\right) - \mathbf{p}_c^b\right)\right) \end{aligned} \tag{24}$$

where, $\mathbf{P}_l^w$ means the endpoint of a line segment, $\mathbf{s}_1$ corresponds to the $\mathbf{q}_b^w$, $\mathbf{s}_2$ means the $\mathbf{q}_c^b$. $\mathbf{s}$ is the Cayley-Gibbs-Rodrigues (CGR) parameterization representation of the orientation to facilitate the derivation calculation, and $\mathbf{s} = \mathbf{n}\tan\frac{\theta}{2}$ with $\mathbf{n}$ is the rotation axis and θ is the axial angle [52][53]. The partial derivative of $\mathbf{q}_b^w(t)$ respect to time is

$$\begin{aligned} \dot{\mathbf{s}}_b^w(t) &= \mathbf{D}(\widetilde{\omega} - \mathbf{b}_g) \\ \mathbf{D} &\triangleq \frac{\partial \mathbf{s}}{\partial \theta} = \frac{1}{2}(\mathbf{I} + \lfloor \mathbf{s} \rfloor_\times + \mathbf{s}\mathbf{s}^\top) \end{aligned}. \tag{25}$$

Specifically, the observability analysis of a nonlinear system can be executed by taking the Lie derivatives of the measurement function $\mathbf{h}(\mathbf{x})$ to build the observability matrix $\mathcal{O}$ [40, 50], i.e.

$$\mathcal{O} = \begin{bmatrix} \nabla_{\mathbf{x}}\mathcal{L}^0\mathbf{h}_k \\ \nabla_{\mathbf{x}}\mathcal{L}^1_{\mathbf{f}_i}\mathbf{h}_k \\ \nabla_{\mathbf{x}}\mathcal{L}^2_{\mathbf{f}_i\mathbf{f}_j}\mathbf{h}_k \\ \vdots \end{bmatrix}, \begin{cases} k = 1,2 \\ i,j = 1,2,3 \end{cases} \tag{26}$$

$$\begin{aligned} \mathcal{L}^0\mathbf{h}_k &= \mathbf{h}_k(x) \\ \mathcal{L}^{i+1}_{\mathbf{f}_j}\mathbf{h}_k &= \nabla_{\mathbf{x}}\mathcal{L}^i\mathbf{h}_k \cdot \mathbf{f}_j \\ \nabla_{\mathbf{x}}\mathcal{L}^i\mathbf{h}_k &= \begin{bmatrix} \frac{\partial\mathcal{L}^i\mathbf{h}_k}{\partial x_1} & \frac{\partial\mathcal{L}^i\mathbf{h}_k}{\partial x_2} & \cdots & \frac{\partial\mathcal{L}^i\mathbf{h}_k}{\partial x_m} \end{bmatrix} \end{aligned}. \tag{27}$$

The right nullspace of matrix $\mathcal{O}$ determines the unobservable directions of the system. However, the full expression of $\mathcal{O}$ is unwieldy, this paper refers to the simplification in literature [40], which utilized a set of basis elements of state variables, as $\boldsymbol{\beta}(\mathbf{x}) = [\beta_1(\mathbf{x})^\top, \beta_2(\mathbf{x})^\top, \cdots, \beta_t(\mathbf{x})^\top]$, to decompose the observability matrix into a product of two matrices as,

$$\mathcal{O} = \Xi \cdot \mathbf{B}. \tag{28}$$

Where, Ξ is a full-rank matrix and $\mathbf{B}$ has the same null space with $\mathcal{O}$, i.e. $null(\mathcal{O}) = null(\mathbf{B})$.

In the proposed prior line map-based VIO system, the basis functions are defined as,

$$\begin{aligned} &\beta_1 \triangleq \mathbf{z}_p, \beta_2 \triangleq \frac{1}{p_z}, \beta_3 \triangleq \mathbf{z}_l, \beta_4 \triangleq \frac{1}{p_{z_l}}, \beta_5 \triangleq \mathbf{C}_1\mathbf{v}, \\ &\beta_6 \triangleq \mathbf{C}_2, \beta_7 \triangleq \mathbf{P}_c^b, \beta_8 \triangleq \mathbf{b}_g, \beta_9 \triangleq \mathbf{C}_1\mathbf{g}, \beta_{10} \triangleq \mathbf{b}_a \end{aligned}, \tag{29}$$

where, $\mathbf{C}_1 = \mathbf{R}_w^b$, $\mathbf{C}_2 = \mathbf{R}_b^c$. Next, the observability matrix can be expressed by the basis functions,

$$\mathcal{O} = \begin{bmatrix} \mathcal{L}^0\mathbf{h}_k \\ \mathcal{L}^1_{\mathbf{f}_i}\mathbf{h}_k \\ \mathcal{L}^2_{\mathbf{f}_i\mathbf{f}_j}\mathbf{h}_k \\ \vdots \end{bmatrix} = \begin{bmatrix} \frac{\partial\mathcal{L}^0\mathbf{h}_k}{\partial\beta} \\ \frac{\partial\mathcal{L}^1_{\mathbf{f}_i}\mathbf{h}_k}{\partial\beta} \\ \frac{\partial\mathcal{L}^2_{\mathbf{f}_i\mathbf{f}_j}\mathbf{h}_k}{\partial\beta} \\ \vdots \end{bmatrix} \frac{\partial\boldsymbol{\beta}}{\partial\mathbf{x}} = \Xi \cdot \mathbf{B}. \tag{30}$$

The Ξ is a full-rank matrix, $\mathbf{B}$ matrix can be written by

$$\begin{aligned} \mathbf{B} &= \begin{bmatrix} \frac{\partial\beta_1}{\partial\mathbf{x}} & \frac{\partial\beta_2}{\partial\mathbf{x}} & \cdots & \frac{\partial\beta_8}{\partial\mathbf{x}} \end{bmatrix}^\top \\ \frac{\partial\beta_i}{\partial\mathbf{x}} &= \begin{bmatrix} \frac{\partial\beta_i}{\partial x_1} & \frac{\partial\beta_i}{\partial x_2} & \cdots & \frac{\partial\beta_i}{\partial x_m} \end{bmatrix}, \end{aligned} \tag{31}$$

where, the m is the number of the states. Take the $\frac{\partial\beta_1}{\partial\mathbf{x}}$ as an example, firstly,

$$\beta_1 = \mathbf{z}_p = \frac{1}{p_z}\begin{bmatrix} p_x \\ p_y \end{bmatrix}, \tag{32}$$

$\mathbf{z}_p$ is the pixel coordinate of a feature point under the normalized camera model, the span of $\partial\beta_1$ with respect to $\mathbf{x}$, i.e.

$$\begin{aligned} \frac{\partial\beta_1}{\partial\mathbf{x}} &= \begin{bmatrix} \frac{\partial\beta_1}{\partial\mathbf{p}} & \frac{\partial\beta_1}{\partial\mathbf{v}} & \frac{\partial\beta_1}{\partial\theta_1}\frac{\partial\theta_1}{\partial\mathbf{s}_1} & \frac{\partial\beta_1}{\partial\mathbf{b}_g} & \frac{\partial\beta_1}{\partial\mathbf{b}_a} & \frac{\partial\beta_1}{\partial\mathbf{p}_c^b} & \frac{\partial\beta_1}{\partial\theta_2}\frac{\partial\theta_2}{\partial\mathbf{s}_2} & \frac{\partial\beta_1}{\partial\mathbf{p}_k^w} \end{bmatrix} \\ &= \frac{\partial\mathbf{z}_p}{\partial\mathbf{P}_k^c} \cdot \frac{\partial\mathbf{P}_k^c}{\partial\mathbf{x}} \end{aligned} \tag{33}$$

with,

$$\begin{aligned} \frac{\partial\mathbf{z}_p}{\partial\mathbf{P}_k^c} &= \begin{bmatrix} \frac{1}{p_z} & 0 & -\frac{p_x}{p_z^2} \\ 0 & \frac{1}{p_z} & -\frac{p_y}{p_z^2} \end{bmatrix} \\ \frac{\partial\mathbf{P}_k^c}{\partial\mathbf{x}} &= \begin{bmatrix} \mathbf{C}_2\mathbf{C}_1 & \mathbf{0}_3 & \mathbf{C}_2\lfloor\mathbf{P}_k^b\rfloor_\times\frac{\partial\theta_1}{\partial\mathbf{s}_1} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{C}_2 & \lfloor\mathbf{P}_k^c\rfloor_\times\frac{\partial\theta_2}{\partial\mathbf{s}_2} & -\mathbf{C}_2\mathbf{C}_1 \end{bmatrix} \end{aligned} \tag{34}$$

$\frac{\partial\mathbf{z}_p}{\partial\mathbf{P}_k^c}$ comes from the normalized camera intrinsic projection model. $\mathbf{P}_k^c$ is the point feature position on the camera frame. Similarly, the matrix $\mathbf{B}$ can be written as,

$$\begin{aligned} \mathbf{B} &\triangleq \mathbf{B}_1\mathbf{B}_2 \\ &= \begin{bmatrix} \zeta & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \eta & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \end{aligned} \tag{35}$$

where,

$$\zeta = \begin{bmatrix} \frac{1}{p_z} & 0 & -\frac{p_x}{p_z^2} \\ 0 & \frac{1}{p_z} & -\frac{p_y}{p_z^2} \\ 0 & 0 & -\frac{1}{p_z^2} \end{bmatrix}, \quad \eta = \begin{bmatrix} \frac{1}{p_{z_l}} & 0 & -\frac{p_{x_l}}{p_{z_l}^2} \\ 0 & \frac{1}{p_{z_l}} & -\frac{p_{y_l}}{p_{z_l}^2} \\ 0 & 0 & -\frac{1}{p_{z_l}^2} \end{bmatrix}. \tag{36}$$

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

$$\mathrm{B}_2 = \begin{bmatrix} \mathbf{C}_2\mathbf{C}_1 & \mathbf{0}_3 & \mathbf{C}_2\lfloor \mathbf{P}_k^b \rfloor_\times \frac{\partial\theta_1}{\partial\mathbf{s}_1} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{C}_2 & \lfloor \mathbf{P}_k^c \rfloor_\times \frac{\partial\theta_2}{\partial\mathbf{s}_2} & -\mathbf{C}_2\mathbf{C}_1 \\ \mathbf{C}_2\mathbf{C}_1 & \mathbf{0}_3 & \mathbf{C}_2\lfloor \mathbf{P}_l^b \rfloor_\times \frac{\partial\theta_1}{\partial\mathbf{s}_1} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{C}_2 & \lfloor \mathbf{P}_l^c \rfloor_\times \frac{\partial\theta_2}{\partial\mathbf{s}_2} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{C}_1 & \lfloor \mathbf{C}_1\mathbf{v} \rfloor_\times \frac{\partial\theta_1}{\partial\mathbf{s}_1} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \lfloor \mathbf{C}_1\mathbf{g}^w \rfloor_\times \frac{\partial\theta_1}{\partial\mathbf{s}_1} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}. \tag{37}$$

It is clear that the $\mathbf{B}_1$ matrix is full-rank, so the unobservable modes can be analyzed by the right nullspace of the matrix $\mathbf{B}_2$. Assuming the left null space of $\mathbf{B}_2$ is $\mathbf{N}_L = [\mathbf{N}_1, \mathbf{N}_2, \mathbf{N}_3, \mathbf{N}_4, \mathbf{N}_5, \mathbf{N}_6, \mathbf{N}_7, \mathbf{N}_8]$we have

$$\mathbf{0} = [\mathbf{N}_1, \mathbf{N}_2, \mathbf{N}_3, \mathbf{N}_4, \mathbf{N}_5, \mathbf{N}_6, \mathbf{N}_7, \mathbf{N}_8] \cdot \mathbf{B}_2 \tag{38}$$

It can be seen from (38) that $\mathbf{N}_3\mathbf{C}_1 = \mathbf{0}$, $\mathbf{N}_6\mathbf{I}_1 = \mathbf{0}$, $\mathbf{N}_8\mathbf{I}_1 = \mathbf{0}$, and $-\mathbf{N}_1\mathbf{C}_2\mathbf{C}_1 = \mathbf{0}$, so $\mathbf{N}_1 = 0$, $\mathbf{N}_3 = 0$, $\mathbf{N}_6 = 0$, and $\mathbf{N}_8 = 0$. Then $\mathbf{N}_2\mathbf{C}_2\mathbf{C}_1 = \mathbf{0} \rightarrow \mathbf{N}_2 = \mathbf{0}$, so $\mathbf{N}_5 = \mathbf{0}$ and $\mathbf{N}_4 = \mathbf{0}$ based on the sixth and seventh columns of matrix $\mathbf{B}_2$. Therefore, only the $\mathbf{N}_7$ is non-zero vector, and based on the third column of $\mathbf{B}_2$, we have

$$\mathbf{N}_7 \lfloor \mathbf{C}_1\mathbf{g}^w \rfloor_\times \frac{\partial\theta_1}{\partial\mathbf{s}_1} = \mathbf{0}. \tag{39}$$

Since the matrix $\frac{\partial\theta_1}{\partial\mathbf{s}_1}$ is full rank, we have

$$\mathbf{N}_7 = \pm(\mathbf{C}_1\mathbf{g}^w)^\top. \tag{40}$$

In summary, the left null space of $\mathbf{B}_2$ is

$$\mathbf{N}_L = [\mathbf{0}_{1\times3}, \mathbf{0}_{1\times3}, \mathbf{0}_{1\times3}, \mathbf{0}_{1\times3}, \mathbf{0}_{1\times3}, \mathbf{0}_{1\times3}, (\mathbf{C}_1\mathbf{g}^w)^\top, \mathbf{0}_{1\times3}]_{1\times24} \tag{41}$$

Its dimension is 1. Therefore, the dimension of the right null space of 24×24 matrix $\mathbf{B}_2$ is also 1. The right null space of $\mathbf{B}_2$ is the following vector,

$$\mathbf{N}_R = \begin{bmatrix} \lfloor \mathbf{C}_1^\top \mathbf{P}_c^b \rfloor_\times \mathbf{g}^w \\ -\lfloor \mathbf{v} \rfloor_\times \mathbf{g}^w \\ \frac{\partial\mathbf{s}_1}{\partial\theta_1}\mathbf{C}_1\mathbf{g}^w \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ -\frac{\partial\mathbf{s}_2}{\partial\theta_2}\mathbf{C}_2\mathbf{C}_1\mathbf{g}^w \\ \mathbf{0}_3 \end{bmatrix}_{24\times1}. \tag{42}$$

It can be seen from $\mathbf{N}_R$ that the only 1 dimension right null space is related to the gravity $\mathbf{g}^w$. the observability analysis indicates that the global x, y, z translation directions of the proposed system after fusing prior line map are observable. The only unobservable direction is corresponds to global rotation about the gravity vector (yaw angle). The detailed derivation process can be found in supplement material.

VI. Experiment Results

In this section, we evaluate the performance of the proposed method through experiments in both simulation and real-world environments. The simulation environment is conducted in the CARLA simulator [54] and the real-world experiments are based on the public EuRoC micro aerial vehicle (MAV) dataset [55]. The comparison methods we executed include

1) **VINS**: the VINS-Mono method without loop-closure module that is developed by Qin [10].
2) **Benchmark**: a loosely coupled structure between prior line feature factor and VIO proposed by Yu [22].
3) **LCL**: our previous proposed **l**oosely **c**oupled **l**ine feature-aided framework with outlier rejection module [23].
4) **TCL**: the proposed **t**ightly **c**oupled **l**ine-aided framework in this paper.

The criteria of localization accuracy used in this section are the root mean squared error (RMSE), absolute trajectory error (ATE) [56], and Euclidean distance error. All methods are implemented on Ubuntu 20.04 with robot operating system (ROS) and conducted on a PC with Inter-Core i9-12900K and 32 GB memory.

TABLE I: RMSE of ATE (m) on CARLA dataset

Sequence	VINS	Benchmark	LCL	TCL
CARLA_01	2.070	2.726	1.931	**1.712**
CARLA_02	4.613	4.312	3.621	**3.205**

A. Simulation Experiments

1) Setup: CARLA simulator is widely used in autonomous driving and SLAM domains. It supports flexible sensor combinations, such as cameras, LiDAR, IMU, and radar et al. We can synthesize diverse scenarios to simulate real-world environments. In this section, we apply an unmanned ground vehicle (UGV) running in urban environments. A LiDAR (64 channels, 100 Hz), IMU (100 Hz), and camera (960x600, 30 Hz) are used to collect datasets. The UGV runs two different trajectories with lengths of 1km and 2km in a town, which are shown in Fig. 6 and called as CARLA_01 and CARLA_02. Some selected city and tunnel scenes in this map and feature detection results are shown in Fig. 5. It can be seen that lines can provide more robust features than points in challenging scenarios with low texture and high similarity.

2) Results: We have named the simulation trajectories CARLA_01 and CARLA_02 according to their lengths. Fig. 7 shows the ATE comparison results of VINS-Mono, Benchmark, LCL, and TCL based on dataset CARLA_01. We can see that our proposed TCL method outperforms others. Our methods greatly improve the computational stability compared to Benchmark and have a better absolute positioning accuracy than VINS-Mono. The RMSE of ATE results about two trajectories are displayed in Table I. On three sets of simulation datasets, our proposed TCL method has an average improvement of 11.44% in positioning accuracy compared to LCL, 26.42% compared to VINS-Mono, and 30.15% compared to Benchmark.

As an example, Fig. 8 and Fig. 9 depict the positioning error and rotation error of CARLA_01 compared with VINS-Mono

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

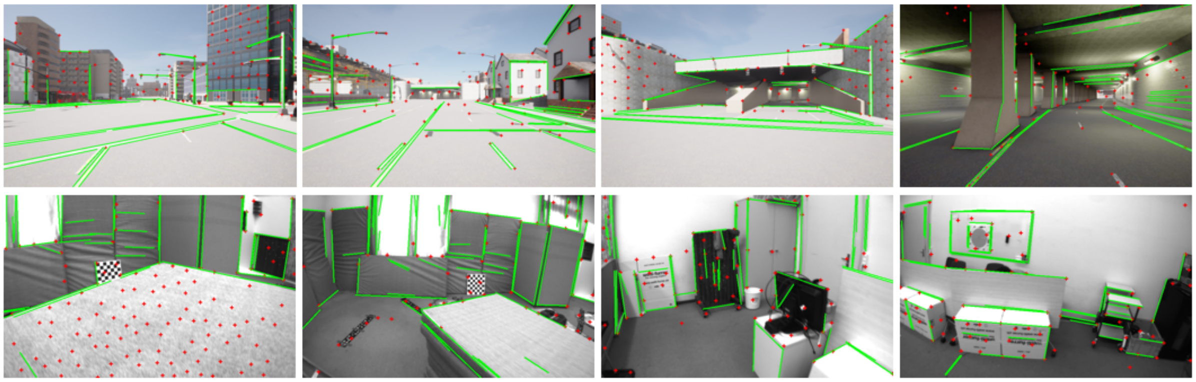


Fig. 5: Images in the top row show simulated urban environments on CARLA and images in the bottom row display indoor scenarios of the EuRoC dataset. Red marks are the point features and green lines denote the line features.

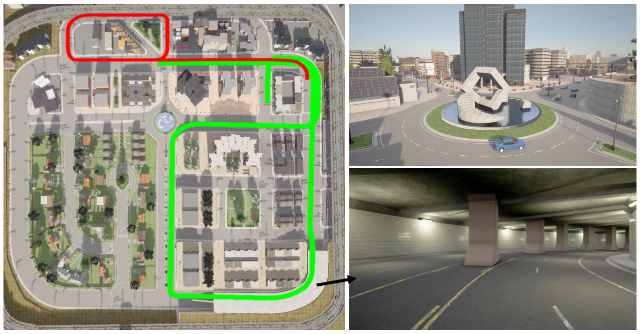


Fig. 6: Two driving trajectories and scenarios in CARLA simulator. The red and green curves represent the trajectories of 1km and 2km, respectively.

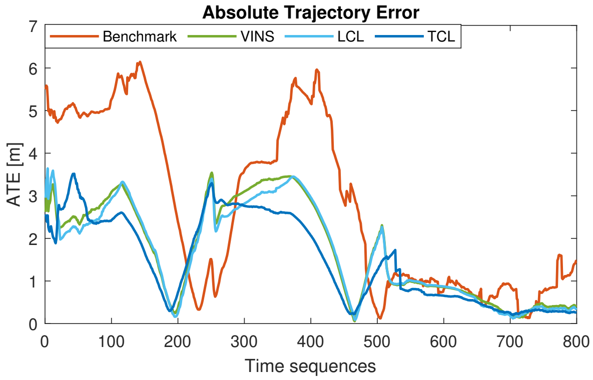


Fig. 7: The ATE results of the CARLA_01 dataset.

and Benchmark, respectively. It can be observed that the proposed method has better accuracy and robustness performances than Benchmark in six directions and its error distribution does not exhibit as many abrupt changes as Benchmark. This is one of the advantages of tightly coupled structures, which have smoother trajectories than loosely coupled structures and are similar to VINS-Mono. Compared to VINS-Mono, TCL has obviously lower errors in x-direction positioning and yaw-angle results, and it has shown some improvement in accuracy

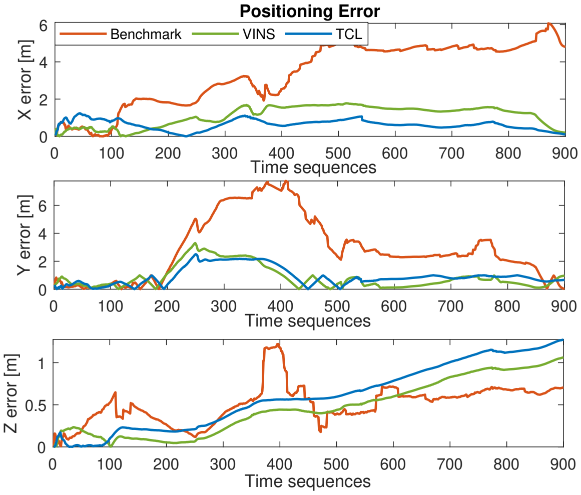


Fig. 8: The positioning error of CARLA_01 dataset.

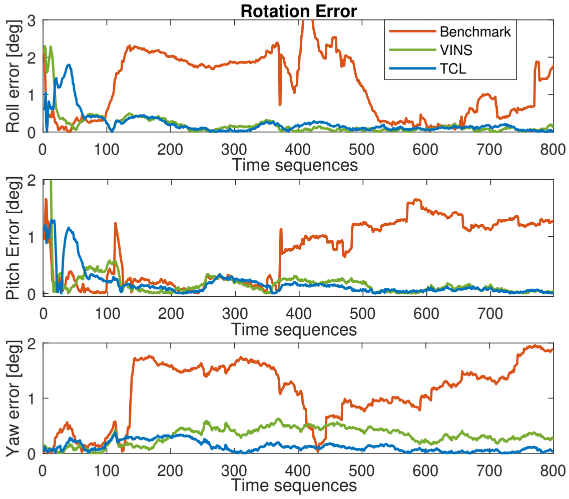


Fig. 9: The Rotation error of CARLA_01 dataset

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

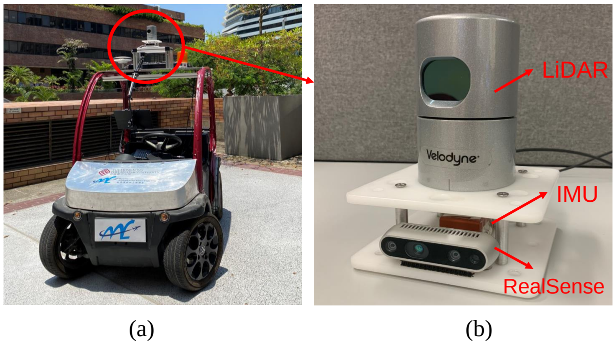


Fig. 10: (a) The vehicle platform. (b) Sensor unit

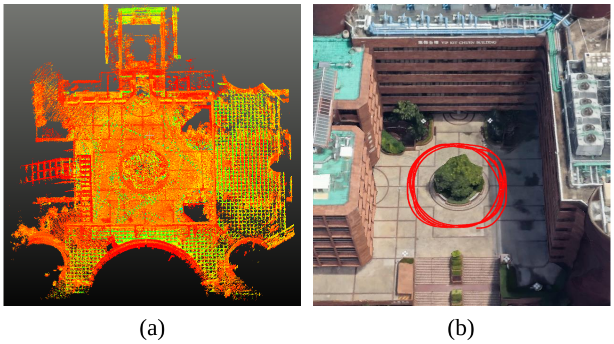


Fig. 11: (a) The 3D point cloud of the PolyU square. (b) Motion trajectory of vehicle on Google 3D map.

in the y-direction, roll, and pitch rotation angles. However, the positioning result in the z-direction is somewhat inferior to that of VINS-Mono. Combined with the observability analysis in the previous section, this result is reasonable. The proposed method can alleviate cumulative drifts but cannot completely solve this problem.

B. Real-world Experiments

1) setup: We have also assessed the performance of the proposed method based on real-world environments. The first dataset we utilized is part of the public EuRoC dataset, which consists of six sequences recorded by an MAV flying through two distinct rooms with three different challenges. These challenges are related to the motion and speed of the MAV, illumination, and texture of environments. Besides, the prior point cloud maps of rooms and the ground truth are collected by a Leica MS50 scanner, Vicon motion capture platform, IMU (ADIS16448, 200 Hz), and cameras (640×480, 30 Hz) [55]. The environments of indoor rooms can be seen in Fig. 5 and the corresponding line maps are shown in Fig. 13.

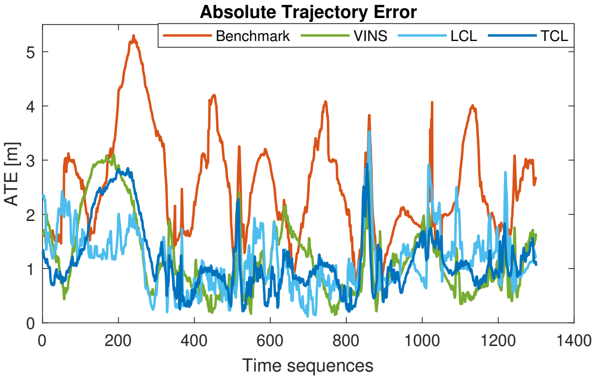


Fig. 12: The ATE results of the PolyU dataset.

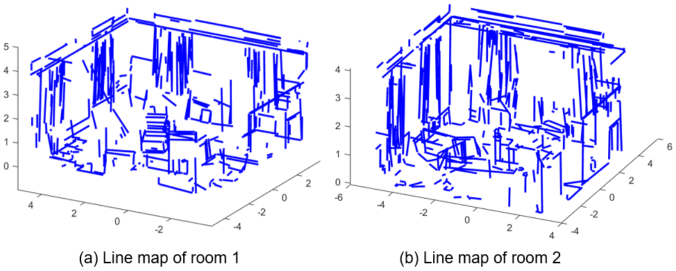


Fig. 13: The indoor line maps of two rooms on EuRoC dataset

Besides, we conduct an unmanned ground vehicle (UGV) real-world experiment in the Hong Kong Polytechnic University (PolyU) campus. Fig. 10 displays the vehicle platform and the sensor unit, which consists of a LiDAR (HDL 32E Velodyne, 10 Hz), an IMU (Xsens Mti 10, 200 Hz), and a RealSense camera (D435i, 640×480, 30 Hz). Fig. 11 shows the collected 3D point cloud of the PolyU square and the motion trajectory of the vehicle on Google Maps. The ground-truth are calculated by LIO_SAM implementation. [9]

2) Localization Results: The RMSE of ATE is used to validate the performance of localization accuracy. The results of the proposed method and comparison methods are shown in Table II. It can be seen that the proposed method is superior to the others on PolyU dataset. Specifically, the accuracy of our algorithm is 17.7% 40.3%, and 13.3% better than VINS, benchmark, and LCL, respectively. The ATE curve results are shown in Fig. 12. For EuRoC dataset, the first three and last three datasets in this table correspond to three trajectories in the same room. The same room means the same point cloud prior map and shares a prior lightweight line map. As can be seen in this table, the results in the same room show the effectiveness of the proposed method in performing different trajectories based on the same prior map. It also can be seen that the method proposed in this paper outperforms others in the easy and medium scenarios and barely increases the accuracy in the difficult scenarios. The reason is that the performance of the prior map-aided localization method relies on the initial guess of the cross-modality matching. However, in difficult scenarios, it is unstable to provide a good enough initial guess for the tightly coupled method.

As an illustrative example, the positioning error and rotation error of the V1_02_medium dataset are shown in Fig. 14 and Fig. 15, respectively. It can be seen that our method performs better in the x-direction and yaw rotation angle than other methods, which is consistent with the simulation experiments. In addition, the trajectories result of ground truth, VINS-Mono, and ours based on the EuRoC dataset V1_02_medium

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

TABLE II: RMSE of ATE (m) statistics on real-world dataset

Sequence	VINS	Benchmark	LCL	TCL
PolyU	1.588	2.191	1.502	**1.307**
V1_01_esay	0.078	0.164	0.152	**0.068**
V1_02_medium	0.110	0.152	0.092	**0.084**
V1_03_difficult	0.189	0.217	**0.161**	0.182
V2_01_esay	0.096	0.192	0.166	**0.068**
V2_02_medium	0.167	0.281	0.136	**0.108**
V2_03_difficult	0.253	0.341	**0.218**	0.254

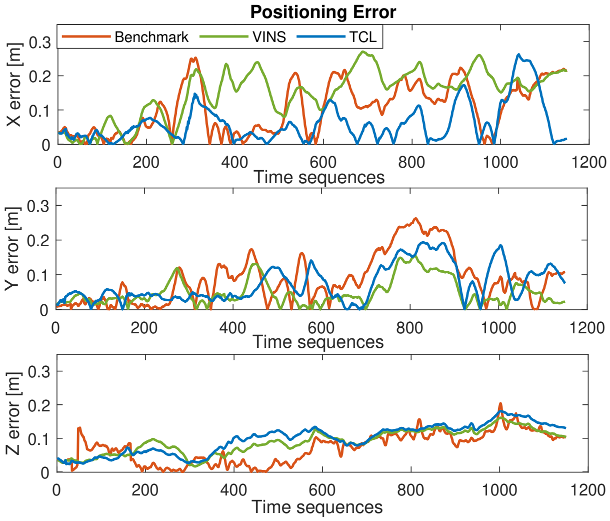


Fig. 14: The positioning error of V1_02_medium

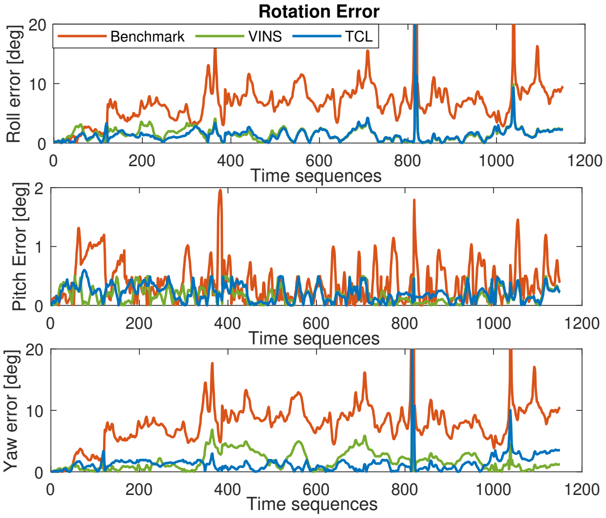


Fig. 15: The rotation error of V1_02_medium

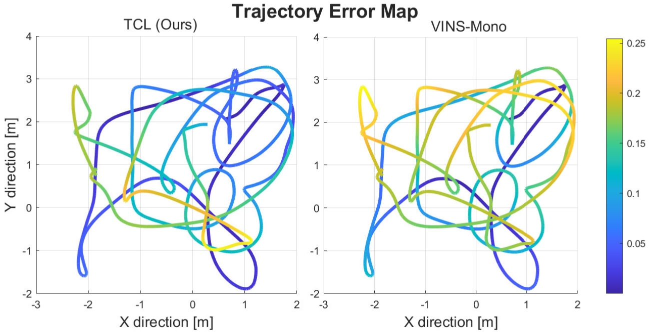


Fig. 16: The trajectory error map of the EuRoC V1_02_medium dataset. The color of trajectory corresponds to localization error.

is illustrated in Fig. 16. It can be seen that the proposed method has better performance than VINS-Mono.

3) Time Statistics: The time consumption statistics of VINS, benchmark, LCL and the proposed TCL method on three datasets are displayed in Table III, where, the feature processing includes the image feature detection and tracking module, the optimization involves sliding window state estimation and marginalization. As seen in the table, after fusing a priori maps in a tightly coupled manner, our algorithm takes more time than VINS, mainly in feature processing. However, compared to benchmark and LCL in a loosely coupled structure, our method has an advantage in overall time consumption.

TABLE III: Time statistics of Feature Processing / Optimization Treads (ms)

Datasets	VINS	Benchmark	LCL	TCL
EuRoC	5.8/17.4	11.3/7.5	9.9/1.5	8.9/25.5
PolyU	2.9/23.3	5.6/4.8	4.8/1.3	4.5/28.1
CARLA_01	8.2/20.1	48.9/52.1	46.5/1.8	24.2/32.2

VII. Conclusion

In this article, we proposed a tightly coupled nonlinear optimization-based state estimation method that integrates the classic VIO and a lightweight prior line map containing IMU pre-integration, point feature, and cross-modality line feature factors. For the line factor, we utilize the simple two endpoints to represent a line feature and present a fast line tracking strategy to monitor and reject outliers for improving the stability of the cross-model line matching. Moreover, a novel line feature-based cost model is proposed and proved in this paper, and we also analyze the observability of the proposed framework for the first time. The result shows that global translation directions x, y, and z are observable in this method and only the global yaw rotation is unobservable. Finally, we conducted experiments in both simulation outdoor and real-world indoor environments to verify the performance of our system, and the results show the effectiveness of our method. Our subsequent work will discuss the impact of the line feature degeneration on state observability. After all, in the observability analysis of this article, we assume that the spatial coordinates of the line endpoints are known. However, in the actual optimization

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

process, we can only restrict endpoint constraints to the same line segment, not strictly correct point-to-point constraints. Besides, the initial coordinate transformation between the prior map and the origin pose of VIO is still a problem that needs to be considered in this system. It can also be interesting to explore the possibility of estimating the initial transformation parameters as additional unknown variables during the state estimation.

REFERENCES

[1] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous localization and mapping: A survey of current trends in autonomous driving," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 3, pp. 194–220, 2017.

[2] L. Chen, Y. Li, C. Huang, B. Li, Y. Xing, D. Tian, L. Li, Z. Hu, X. Na, Z. Li *et al.*, "Milestones in autonomous driving and intelligent vehicles: Survey of surveys," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1046–1056, 2022.

[3] 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.

[4] W. W. Wen, G. Zhang, and L.-T. Hsu, "Gnss nlos exclusion based on dynamic object detection using lidar point cloud," *IEEE transactions on intelligent transportation systems*, vol. 22, no. 2, pp. 853–862, 2019.

[5] W. Wen, Y. Zhou, G. Zhang, S. Fahandezh-Saadi, X. Bai, W. Zhan, M. Tomizuka, and L.-T. Hsu, "Urbanloco: A full sensor suite dataset for mapping and localization in urban scenes," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2310–2316.

[6] W. Wen, X. Bai, Y. C. Kan, and L.-T. Hsu, "Tightly coupled gnss/ins integration via factor graph and aided by fish-eye camera," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 10 651–10 662, 2019.

[7] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and Systems*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "Openvins: A research platform for visual-inertial estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4666–4672.

[13] M. Ramezani and K. Khoshelham, "Vehicle positioning in gnss-deprived urban areas by stereo visual-inertial odometry," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 2, pp. 208–217, 2018.

[14] X. Gao, T. Zhang, Y. Liu, and Q. Yan, "14 lectures on visual slam: from theory to practice," *Publishing House of Electronics Industry, Beijing*, 2017.

[15] 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.

[16] L. Li, M. Yang, H. Li, C. Wang, and B. Wang, "Robust localization for intelligent vehicles based on compressed road scene map in urban environments," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 250–262, 2022.

[17] 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.

[18] 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.

[19] C. Ye, H. Zhao, L. Ma, H. Jiang, H. Li, R. Wang, M. A. Chapman, J. M. Junior, and J. Li, "Robust lane extraction from mls point clouds towards hd maps especially in curve road," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1505–1518, 2020.

[20] X. Xia, N. P. Bhatt, A. Khajepour, and E. Hashemi, "Integrated inertial-lidar-based map matching localization for varying environments," *IEEE Transactions on Intelligent Vehicles*, 2023.

[21] 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.

[22] 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.

[23] 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.

[24] D. Bétaille and R. Toledo-Moreo, "Creating enhanced maps for lane-level vehicle navigation," *IEEE Trans-*

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

actions on Intelligent Transportation Systems, vol. 11, no. 4, pp. 786–798, 2010.

[25] M. Bellusci, P. Cudrano, S. Mentasti, R. E. F. Cortelazzo, and M. Matteucci, "Semantic interpretation of raw survey vehicle sensory data for lane-level hd map generation," *Robotics and Autonomous Systems*, p. 104513, 2023.

[26] 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.

[27] 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.

[28] 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.

[29] 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.

[30] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. Spie, 1992, pp. 586–606.

[31] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Transactions on automatic control*, vol. 22, no. 5, pp. 728–740, 1977.

[32] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2001.

[33] J. Hernandez, K. Tsotsos, and S. Soatto, "Observability, identifiability and sensitivity of vision-aided inertial navigation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 2319–2325.

[34] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based rules for designing consistent ekf slam estimators," *The International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, 2010.

[35] 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.

[36] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *The International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, 2011.

[37] C. X. Guo and S. I. Roumeliotis, "Imu-rgbd camera 3d pose estimation and extrinsic calibration: Observability analysis and consistency improvement," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 2935–2942.

[38] Y. Yang, P. Geneva, X. Zuo, and G. Huang, "Online self-calibration for visual-inertial navigation: Models, analysis, and degeneracy," *IEEE Transactions on Robotics*, 2023.

[39] A. Martinelli, "Vision and imu data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 44–60, 2011.

[40] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Camera-imu-based localization: Observability analysis and consistency improvement," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 182–201, 2014.

[41] C. X. Guo and S. I. Roumeliotis, "Imu-rgbd camera navigation using point and plane features," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 3164–3171.

[42] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 61–76, 2011.

[43] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation." Georgia Institute of Technology, 2015.

[44] 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.

[45] 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.

[46] G. Zhang, J. H. Lee, J. Lim, and I. H. Suh, "Building a 3-d line-based map using stereo slam," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1364–1377, 2015.

[47] X. Lu, Y. Liu, and K. Li, "Fast 3d line segment detection from unorganized point cloud," *arXiv preprint arXiv:1901.02532*, 2019.

[48] J. H. Lee, S. Lee, G. Zhang, J. Lim, W. K. Chung, and I. H. Suh, "Outdoor place recognition in urban environments using straight lines," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 5550–5557.

[49] 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.

[50] J. Svacha, J. Paulos, G. Loianno, and V. Kumar, "Imu-based inertia estimation for a quadrotor using newton-euler dynamics," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 3861–3867, 2020.

[51] K. Zhang, C. Jiang, J. Li, S. Yang, T. Ma, C. Xu, and F. Gao, "Dido: Deep inertial quadrotor dynamical odometry," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9083–9090, 2022.

[52] P. Furgale, T. D. Barfoot, and G. Sibley, "Continuous-time batch estimation using temporal basis functions,"

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3419101

in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 2088–2095.

[53] P. C. Hughes, *Spacecraft attitude dynamics*. Courier Corporation, 2012.

[54] 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.

[55] 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.

[56] 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.

Xi Zheng received the B.Eng. and M.Eng degrees in astronautics from Northwestern Polytechnical University, Xi’an, China, in 2015 and 2018, respectively. She is currently working toward the Ph.D degree with the Department of Aeronautical and Aviation Engineering, the Hong Kong Polytechnic University. Her research interests include visual SLAM and safety quantifiable localization.

Weisong Wen (Member, IEEE) received a BEng degree in Mechanical Engineering from Beijing Information Science and Technology University (BISTU), Beijing, China, in 2015, and an MEng degree in Mechanical Engineering from the China Agricultural University, in 2017. After that, he received a PhD degree in Mechanical Engineering from The Hong Kong Polytechnic University (PolyU), in 2020. He was also a visiting PhD student with the Faculty of Engineering, University of California, Berkeley (UC Berkeley) in 2018. Before joining PolyU as an Assistant Professor in 2023, he was a Research Assistant Professor at AAE of PolyU since 2021. He has published 30 SCI papers and 40 conference papers in the field of GNSS (ION GNSS+) and navigation for Robotic systems (IEEE ICRA, IEEE ITSC), such as autonomous driving vehicles. He won the innovation award from TechConnect 2021, the Best Presentation Award from the Institute of Navigation (ION) in 2020, and the First Prize in Hong Kong Section in Qianhai-Guangdong-Macao Youth Innovation and Entrepreneurship Competition in 2019 based on his research achievements in 3D LiDAR aided GNSS positioning for robotics navigation in urban canyons. The developed 3D LiDAR-aided GNSS positioning method has been reported by top magazines such as Inside GNSS and has attracted industry recognition with remarkable knowledge transfer.

Li-Ta Hsu received the B.S. and Ph.D. degrees in aeronautics and astronautics from National Cheng Kung University, Taiwan, in 2007 and 2013, respectively. He is currently an associate professor with the Department of Aeronautical and Aviation Engineering. The Hong Kong Polytechnic University, before he served as a post-doctoral researcher in the Institute of Industrial Science at the University of Tokyo, Japan. In 2012, he was a visiting scholar at University College London, the U.K. His research interests include GNSS positioning in challenging environments and localization for pedestrian, autonomous driving vehicle, and unmanned aerial vehicle.

Authorized licensed use limited to: Hong Kong Polytechnic University. Downloaded on December 09,2024 at 14:08:50 UTC from IEEE Xplore. Restrictions apply.
© 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.