A Tightly Coupled Visual-Inertial GNSS State Estimator Based on Point-Line Feature

Bo Dong and Kai Zhang

Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China; db19@mails.tsinghua.edu.cn
Research Institute of Tsinghua, Pearl River Delta, Guangzhou 510530, China; Correspondence: zhangkai@sz.tsinghua.edu.cn

Abstract: Visual-inertial odometry (VIO) is known to suffer from drifting and can only provide local coordinates. In this paper, we propose a tightly coupled GNSS-VIO system based on point-line features for robust and drift-free state estimation. Feature-based methods are not robust in complex areas such as weak or repeated textures. To deal with this problem, line features with more environmental structure information can be extracted. In addition, to eliminate the accumulated drift of VIO, we tightly fused the GNSS measurement with visual and inertial information. The GNSS pseudorange measurements are real-time and unambiguous but experience large errors. The GNSS carrier phase measurements can achieve centimeter-level positioning accuracy, but the solution to the whole-cycle ambiguity is complex and time-consuming, which degrades the real-time performance of a state estimator. To combine the advantages of the two measurements, we use the carrier phase smoothed pseudorange instead of pseudorange to perform state estimation. Furthermore, the existence of the GNSS receiver and IMU also makes the extrinsic parameter calibration crucial. Our proposed system can calibrate the extrinsic translation parameter between the GNSS receiver and IMU in real-time. Finally, we show that the states represented in the ECEF frame are fully observable, and the tightly coupled GNSS-VIO state estimator is consistent. We conducted experiments on public datasets. The experimental results demonstrate that the positioning precision of our system is improved and the system is robust and real-time.

Keywords: GNSS-VIO; line feature; carrier phase smoothed pseudorange; parameter calibration; observability

1. Introduction

Localization plays an important role in many applications such as robotics, unmanned driving, and unmanned aerial vehicles (UAVs). The fusion of information from multiple sensors for localization is the current mainstream method. The sensor fusion approaches have been widely studied for decades and can be divided into two streams: loosely coupled and tightly coupled approaches. Loosely coupled approaches [1,2] process the measurements of each sensor individually and then fuse all the results to obtain the final results. Tightly coupled approaches [3,4] combine measurements from all sensors and then use an estimator to process these measurements to obtain the final results. In general, the accuracy of tightly coupled approaches will be higher than that of loosely coupled approaches.

Visual odometry (VO) has received more attention due to the low price of cameras. Davison et al. [5] introduced MonoSLAM, which is the first monocular visual SLAM system that extracts the Shi-Tomasi corner [6]. MonoSLAM actively searches for matching point features in projected ellipses. However, because of the high computational complexity of MonoSLAM, it can only handle some small scenes. Klein et al. [7] proposed PTAM, which is the first optimization-based monocular visual SLAM system that executes feature tracking and mapping as two independent tasks in parallel in two threads.
Because PTAM is designed for small-scale AR scenes, it also has some disadvantages. For example, it can only handle small-scale scenes, and the tracked features are easy to lose. A milestone work after PTAM is ORB-SLAM [8] and ORB-SLAM2 [9]. ORB-SLAM improves PTAM by detecting and tracking ORB features and adding loop closure. These improvements greatly improve the positioning precision of VO. However, ORB-SLAM can only build sparse point cloud maps and cannot track keyframes during pure rotation, which also limits its application.

In addition, when encountering environments with repeated textures, point features cannot express environmental structure information well. On the contrary, line features contain more information related to the environmental structure and can overcome the interference caused by repeated textures. Therefore, point-line feature fusion can ensure that the VO system is more robust. He Yijia et al. [10] proposed PL-VIO, which is a VIO system based on point-line features. It uses the LSD line segment extraction algorithm [11] from OpenCV [12] to detect line features. However, LSD becomes the bottleneck for real-time performance due to its high computational cost [13], which causes the performance of PL-VIO to be seriously affected. To improve the speed and positioning precision of PL-VIO, Fu Qiang et al. [14] proposed PL-VINS, which is also a VIO framework based on point and line features. PL-VINS designs a modified LSD algorithm by studying hidden parameter tuning and length rejection strategy. The modified LSD runs at least three times faster than the LSD. PLF-VINS [15] is another VIO system based on point-line features, which introduces two methods for fusing point and line features. The first method is to use the positional similarity of point-line features to search for the relationship between point and line features. The second method is to fuse 3D parallel lines. The residuals formed by the two methods are then added to the VIO system. The results of PLF-VINS show that its positioning precision is greatly improved compared to some classic SLAM systems, such as OKVIS [4] and VINS-Mono [16].

Line features can improve the positioning precision of the VO system. However, in weak textures or dark scenes, the system still cannot extract a sufficient number of point and line features, which can cause large positioning errors. To overcome the shortcoming, fusing IMU measurements with images is a very effective method. MSCKF [17,18] is a VIO system based on EKF, which adds camera poses to the states of the system. When cameras observe landmarks, constraints can be formed between camera poses. The system is then updated by an observation model derived from geometric constraints. Since the number of camera poses is much smaller than the number of landmarks, this greatly reduces the time complexity of the system. VINS-Mono [16] is a tightly coupled, nonlinear optimization-based method that can obtain high-precision positioning results. Due to the existence of the loop closure and 4-DoF pose graph optimization, even if the system runs in large-scale scenes, it can still obtain accurate positioning results. In addition, to improve the performance of ORB-SLAM and eliminate the errors of ORB-SLAM, ORB-SLAM3 [19] adds IMU measurements based on ORB-SLAM. ORB-SLAM3 is more robust and has higher positioning precision compared with ORB-SLAM.

The VIO system is more robust than the VO system. Nevertheless, the VIO system has four unobservable directions, namely $x$, $y$, $z$, and $yaw$, which lead to the accumulated drift of VIO. To eliminate the accumulated drift, an effective approach is to combine VIO with GNSS measurements. Lee et al. [20] demonstrated that the GPS-aided VIO system is fully observable in the ENU frame. GVINS [21] is a tightly coupled GNSS-VIO state estimator that combines VIO with GNSS pseudorange and Doppler frequency shift measurements, which achieves high positioning accuracy and is a state-of-the-art method. Li Xingxing et al. [22] introduced a semi-tightly coupled framework based on GNSS Precise Point Positioning (PPP) and stereo VINS. The system can use S-VINS to return accurately predicted positions in GNSS-unfriendly areas. Liu Jinxu et al. [23] also proposed a tightly coupled GNSS-VIO state estimator that fuses GNSS raw measurements with VIO. It drops all GNSS measurements in GNSS-degradation scenes, which limits its positioning precision.
However, there are many challenges in fusing the information from multiple sensors. First, the VIO system cannot extract enough point features in areas with repeated textures. Second, GNSS Single Point Positioning (SPP) uses pseudorange measurement; it can only achieve meter-level positioning accuracy. Therefore, if the pseudorange measurement and VIO are directly fused, the positioning accuracy is not greatly improved. Finally, since the GNSS receiver and IMU are fixed in different spatial positions, it is necessary to estimate the extrinsic parameter between them.

In response to the above problems, this paper proposes a new tightly coupled GNSS-VIO system. Our proposed system is drift-free and can provide global coordinates. The main contributions of the text are as follows:

- To obtain environmental structure information and deal with environments with repeated textures, we extracted line features based on point features.
- To combine the merits of the pseudorange and carrier phase measurements, we used the carrier phase smoothed pseudorange instead of the pseudorange measurement, which can make the GNSS-VIO system run in real-time and improve the positioning accuracy.
- We demonstrate that the states represented in the ECEF frame are fully observable, and the tightly coupled GNSS-VIO state estimator is consistent.

The rest of this paper is organized as follows: Section 2 introduces the implementation method of GNSS-VIO in detail, including line features, GNSS raw measurements processing, and observability analysis. Section 3 summarizes the detailed structure of our system. Section 4 conducts experiments on public datasets. Finally, conclusions are given in Section 5.

2. Methods

2.1. Frames and Notations

The frames involved in our system consist of:

- Sensor Frame: In our system, $(\cdot)^c$, $(\cdot)^b$, and $(\cdot)^r$ denote the camera frame, the body frame, and the GNSS receiver frame respectively.
- Local World Frame: The origin of the local world frame $(\cdot)^w$ is the position where the VIO system starts running, and the $z$-axis is gravity aligned as illustrated in Figure 1.
- ENU Frame: The ENU frame is also called the East–North–Up frame. The $x$-, $y$-, and $z$-axis of the ENU frame point to the east, north, and up directions, respectively. In our system, the ENU frame is at the same origin as the local world, and the $z$-axis of the two frames is aligned (Figure 1). We use $(\cdot)^N$ to represent the ENU frame.
- Geodetic Frame: As shown in Figure 1, geodetic coordinate $(\cdot)^G$ of a point $p$ is represented by geodetic longitude $L$, geodetic latitude $B$, and geodetic height $H$. The geodetic longitude $L$ of $p$ is the angle between the reference meridian and another meridian that passes through $p$. The geodetic latitude $B$ of $p$ is the angle between the ellipsoid normal vector that passes through $p$ and the projection of the ellipsoid normal vector into the equatorial plane. The geodetic height $H$ of $p$ is the minimum distance between $p$ and the reference ellipsoid.
- ECEF Frame: The Earth-Centered, Earth-Fixed (ECEF) frame $(\cdot)^E$ is fixed to Earth. As depicted in Figure 1, the origin of the ECEF frame is at the center of mass of Earth. The $x$-axis points to the intersection of the Equator and the Prime Meridian. The $z$-axis is perpendicular to the equatorial plane in the direction of the North Pole. The $y$-axis is chosen to form a right-handed coordinate system with the $x$- and $z$-axis.
In this paper, $R_b^w$ and $p_b^w$ represent the rotation and translation of the body frame with respect to the local world frame, and $q_b^w$ is the corresponding quaternion form of the rotation $R_b^w$. $v_b^w$, $b_a$, and $b_g$ denote the velocity of the origin of the body frame measured in the local world frame, accelerometer bias, and gyroscope bias, respectively. $p_c^b$ and $q_c^b$ stand for the extrinsic parameters between the camera and IMU. $\delta_r$ and $\delta_t$ represent the receiver clock error and receiver clock drifting rate, respectively. $\phi$ is the yaw between the local world frame and the ENU frame. $R_N^w$ is the rotation matrix of the local world frame with respect to the ENU frame. $p_N^c$ denotes the translation of the ENU frame with respect to the ECEF frame. $p_c^b$ is the extrinsic translation parameter between the GNSS receiver and IMU. $| \cdot |_\times$ represents the skew-symmetric matrix of a 3D vector.

2.2. Line Feature

2.2.1. Plücker Coordinates

In our system, we described a spatial line with the Plücker Coordinates. Given a spatial line $L_w \in \pi_w$, its Plücker Coordinates are represented by $L_w = (n_w^T, d_w^T)^T$, where $n_w \in \mathbb{R}^3$ is the normal vector of the plane determined by $L_w$, and the origin of $\pi_w$ and $d_w \in \mathbb{R}^3$ is the line direction vector. We can transform $L_w$ in the local world frame to $L_c$ in the camera frame by:

$$L_c = \begin{bmatrix} R_c^w & [p_c^w] \times R_c^w \\ 0 & R_c^w \end{bmatrix} L_w = T_{cw} L_w, \quad (1)$$

2.2.2. Line Feature Triangulation

Assuming that $L_w$ is observed by camera $c_i$ and camera $c_j$ in the normalized image plane as $z_{L_w}^{c_i}$ and $z_{L_w}^{c_j}$, the two line segments can be denoted by two endpoints as shown in Figure 2. The two endpoints of $z_{L_w}^{c_i}$ are $s_{L_w}^{c_i} = [u_{s_i}^{c_i}, v_{s_i}^{c_i}, 1]^T$ and $e_{L_w}^{c_i} = [u_{e_i}^{c_i}, v_{e_i}^{c_i}, 1]^T$, and the two endpoints of $z_{L_w}^{c_j}$ are $s_{L_w}^{c_j} = [u_{s_j}^{c_j}, v_{s_j}^{c_j}, 1]^T$ and $e_{L_w}^{c_j} = [u_{e_j}^{c_j}, v_{e_j}^{c_j}, 1]^T$. 

Figure 1. An illustration of the ECEF, Geodetic, ENU, and local world frames.
A 3D plane $\pi$ can be modeled as:

$$
\pi x + \pi y + \pi z + \pi w = 0,
$$

where $n_x = [\pi x, \pi y, \pi z]^{\top}$ is the normal vector of $\pi$. For a point $p_0 = [x_0, y_0, z_0]^{\top}$ on the plane, we can obtain:

$$
\pi w = - (\pi x x_0 + \pi y y_0 + \pi z z_0) = -n_x^{\top} p_0.
$$

According to Equations (2) and (3), we can obtain the plane $\pi_i = (n_{\pi_i}, \pi_{w_i})$ and $\pi_j = (n_{\pi_j}, \pi_{w_j})$. As shown in Figure 2, the normal vector of $\pi_i$ is $n_{\pi_i} = [s_{L_w}^{\top}]_x e_{L_w}^{\top}$, and $\pi_{w_i}$ can be computed by the optical center $c_i = [0, 0, 0]^{\top}$ as $\pi_{w_i} = -n_{\pi_i}^{\top} c_0 = 0$. To obtain $\pi_j$, we need to transform the two endpoints $s_{L_w}^{\top}$ and $e_{L_w}^{\top}$ to the camera frame $c_j$. Therefore, the corresponding reprojected endpoints are $s_{L_w}^{\top} = R_{c_i}^{\top} s_{L_w}^{\top} + p_{c_i}^{\top}$ and $e_{L_w}^{\top} = R_{c_i}^{\top} e_{L_w}^{\top} + p_{c_i}^{\top}$, where $R_{c_i}$ and $p_{c_i}$ can be obtained by visual-inertial alignment. Similarly, the normal vector of $\pi_j$ is $n_{\pi_j} = [s_{L_w}^{\top}]_x e_{L_w}^{\top}$ and $\pi_{w_j}$ can be calculated by the translation $p_{c_i}^{\top}$ from the camera frame $c_j$ to the camera frame $c_i$ as $\pi_{w_j} = -n_{\pi_j}^{\top} p_{c_i}^{\top}$.

After $\pi_i$ and $\pi_j$ are computed, we can obtain the Plücker Coordinates of $L_w$ according to the dual Plücker matrix $L_{w_{\pi}}^{\pi}$:

$$
L_{w_{\pi}}^{\pi} = \begin{bmatrix} [d_w]_x & n_w & 0 \\ -n_w & 0 & 0 \end{bmatrix} = \pi_{\pi i}^{\top} \pi_{\pi j}^{\top},
$$

2.2.3. Orthonormal Representation

Since spatial lines only have 4-DoF, there is a problem of overparameterization using the Plücker Coordinates to represent spatial lines. In contrast, the orthonormal representation $(U, W) \in SO(3) \times SO(2)$ is more suitable for nonlinear optimization. Let $U = R(\psi)$ and $W = R(\phi)$ denote 3D and 2D rotation matrix, respectively, then we have:

$$
U = R(\psi) = [u_1, u_2, u_3] = \begin{bmatrix} n_w & d_w & n_w \times d_w \\ ||n_w|| & ||d_w|| & ||n_w \times d_w|| \end{bmatrix},
$$

$$
W = R(\phi) = \begin{bmatrix} w_1 & -w_2 & w_3 \\ w_2 & w_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} ||n_w|| & -||d_w|| \\ ||d_w|| & ||n_w|| \end{bmatrix}.
$$
where $\psi = [\psi_1, \psi_2, \psi_3]^T$ and $\phi$ are the 3D rotation angles around the $x$, $y$, and $z$-axis of the camera frame and the 2D rotation angle.

Therefore, we can define the orthonormal representation of a spatial line by a four-dimensional vector:

$$o = (\psi, \phi)^T, \quad (7)$$

In addition, given an orthonormal representation $(U, W)$, the corresponding Plücker Coordinates can be obtained by:

$$L'_w = \left[ w_1 u_1^T, w_2 u_2^T \right]^T = \frac{1}{\sqrt{\|n\|^2 + \|d\|^2}} L_w, \quad (8)$$

where $w_1 = \cos(\phi)$, $w_2 = \sin(\phi)$, and $u_i$ is the $i$th column of $U$. $L_w$ and $L'_w$ have a scale factor, but they denote the same spatial line.

2.2.4. Line Feature Reprojection Residual

The line feature reprojection residual is defined in terms of point-to-line distance. Given a spatial line $L_c = (n_c^T, d_c^T)^T$, it can be projected to the image plane by [24]:

$$l = \left[ l_1, l_2, l_3 \right]^T = K L_n, \quad (9)$$

where $K_L$ is the line projection matrix.

Finally, assume that $L_{i j}$ represents the $j$th spatial line $L_j$, which is observed by the $i$th camera frame $c_i$. Then the spatial line reprojection residual can be modeled as:

$$r_L(z_{c_i j}, \chi) = \begin{bmatrix} d(s_{c_i j}^e, l_{c_i j}^e) \\ d(e_{c_i j}^e, l_{c_i j}^e) \end{bmatrix}, \quad (10)$$

where $s_{c_i j}^e = \left[ u_{c_i j}^e, v_{c_i j}^e, 1 \right]^T$ and $e_{c_i j}^e = \left[ u_{c_i j}^e, v_{c_i j}^e, 1 \right]^T$ denote the two endpoints of the line segment projected on the normalized image plane. $d(s, l)$ is the point-to-line distance:

$$d(s, l) = \frac{s^T l}{\sqrt{l_1^2 + l_2^2}}, \quad (11)$$

For the corresponding Jacobian matrix, we followed the routine of [10].

2.3. GNSS Measurements

GNSS measurements include pseudorange, carrier phase, and Doppler frequency shift.

2.3.1. Pseudorange Measurement

The pseudorange is defined as the measured distance obtained by multiplying the travel time of the satellite signal by the speed of light. Due to the influence of satellite clock error, receiver clock error, and ionospheric and tropospheric delays, the pseudorange is prefixed with “pseudo” to distinguish it from the true distance from the satellite to the GNSS receiver. Generally, although the pseudorange measurement only has meter-level positioning precision (the positioning precision of P code is about 10 m, and the positioning precision of C/A code is 20 m to 30 m), it is real-time and has no ambiguity. Therefore, the pseudorange measurement is still very important for GNSS positioning technology. For a certain satellite $s_j$ and a GNSS receiver $r_k$ at time $k$, the pseudorange $\tilde{P}_{r_k}^{s_j}$ can be modeled as:

$$\tilde{P}_{r_k}^{s_j} = \left\| P_{s_j}^F - P_{r_k}^F \right\| + c(\delta t_k - \delta t_{s_j}) + r_{s_j} + \epsilon_{r_k}, \quad (12)$$
where \( \mathbf{p}_E^j \) and \( \mathbf{p}_E^r \) represent the position of satellite \( s_j \) and the GNSS receiver \( r_k \) at time \( k \) in the ECEF frame, respectively. \( c \) is the speed of light. \( \delta t_{s_j} \), \( T_{s_j}^r \), and \( I_{s_j}^r \) are the satellite clock error and tropospheric and ionospheric delays, which can be computed according to the satellite ephemeris. \( \epsilon_{p_j^k} \sim \left( 0, \sigma_{p_j^k}^2 \right) \) denotes the multipath and random error of the pseudorange measurement, which is subject to the Zero-mean Gaussian distribution.

### 2.3.2. Carrier Phase Measurement

Although pseudorange positioning is an important method for GNSS, its error is too large for some applications that require high-precision positioning. In contrast, due to the short wavelength of the carrier phase (\( \lambda_{L_1} = 19 \text{ cm} \) and \( \lambda_{L_2} = 24 \text{ cm} \)), if the carrier phase measurement is used for positioning, it can achieve centimeter-level positioning accuracy. However, since the carrier phase is a periodic sinusoidal signal, and the GNSS receiver can only measure the part of less than one wavelength, there is the problem of the whole-cycle ambiguity, which makes the positioning process time-consuming. The carrier phase is defined as the phase difference between the phase transmitted by the satellite and the reference phase generated by the GNSS receiver. Similar to the pseudorange measurement, the carrier phase measurement is also related to the position of satellite and GNSS receiver. The carrier phase measurement can be modeled as:

\[
\tilde{\Phi}_{s_j^r} = \| \mathbf{p}_E^s - \mathbf{p}_E^r \| + c \left( \delta t_{r_k} - \delta t_{s_j} \right) + T_{s_j}^r - I_{s_j}^r + \lambda N + \epsilon_{\Phi_j^k},
\]

where \( \lambda \) is the carrier wavelength, and \( N \) is the whole-cycle ambiguity. \( \epsilon_{\Phi_j^k} \sim \left( 0, \sigma_{\Phi_j^k}^2 \right) \) represents the multipath and random error of the carrier phase measurement, which is subject to the Zero-mean Gaussian distribution.

### 2.3.3. Doppler Frequency Shift Measurement

The Doppler effect reveals a phenomenon in the spread of waves, that is, the wavelength of the radiation emitted by objects changes accordingly due to the relative motion of the wave source and the observer. As the wave source moves toward the observer, the wavelength becomes shorter and the frequency becomes higher due to the compression of the wave. On the contrary, as the wave source moves away from the observer, the wavelength becomes longer and the frequency becomes lower. Similarly, the Doppler effect can also occur between the satellite and the GNSS receiver. When a satellite orbits Earth in its elliptical orbit, due to the relative motion between the satellite and the GNSS receiver, the frequency of the satellite signal received by the GNSS receiver changes accordingly. This frequency change is called Doppler frequency shift, which can be modeled as:

\[
\Delta f_{s_j}^r = -\frac{1}{\lambda} \left[ \kappa_k^T \left( \mathbf{v}_s^j - \mathbf{v}_r^k \right) + c \left( \delta t_{r_k} - \delta t_{s_j} \right) \right] + \epsilon_{\Delta f_j^k},
\]

where \( \kappa_k = \frac{\left( \mathbf{p}_E^s - \mathbf{p}_E^r \right)}{\| \mathbf{p}_E^s - \mathbf{p}_E^r \|} \) is the unit vector pointing along the line of sight from the GNSS receiver \( r_k \) to the satellite \( s_j \). \( \mathbf{v}_s^j \) and \( \mathbf{v}_r^k \) denote the velocity of the \( j \text{th} \) satellite and the receiver at time \( k \) in the ECEF frame, respectively. \( \delta t_{s_j} \) is the satellite clock drifting rate, which can be calculated according to the satellite ephemeris. \( \epsilon_{\Delta f_j^k} \sim \left( 0, \sigma_{\Delta f_j^k}^2 \right) \) represents the multipath and random error of the Doppler frequency shift measurement, which is subject to the Zero-mean Gaussian distribution.
2.4. Carrier Phase Smoothed Pseudorange

As mentioned above, the pseudorange measurement proves fast and efficient, but can only achieve meter-level positioning precision. In contrast, the carrier phase measurement can achieve centimeter-level positioning precision but is required to solve the whole-cycle ambiguity, which is complex and time-consuming. Therefore, to combine the merits of the two measurements, we can use carrier phase smoothed pseudorange (short for smoothed pseudorange) to improve the positioning precision. In general, the positioning precision of the smoothed pseudorange measurement is several times higher than that of the pseudorange measurement. For a single-frequency receiver, when the whole-cycle ambiguity and ionospheric delay are nearly constant within a period of time, the pseudorange can be smoothed by using the carrier phase. The Hatch filter is the most widely used for carrier phase smoothed pseudorange, which assumes that the ionospheric delay is nearly constant among GNSS epochs and then averages the multiepoch whole-cycle ambiguity and ionospheric delay to improve the positioning accuracy of the pseudorange measurement. The carrier phase smoothed pseudorange based on the Hatch filter can be modeled as:

\[
\rho_{s, k}^{i} = \frac{1}{k_m} \rho_{s, k}^{i} + \left(1 - \frac{1}{k_m}\right) \left[\tilde{\rho}_{s, k}^{i} + \tilde{\phi}_{s, k}^{i} \Phi_{s, k}^{i} - \Phi_{s, k-1}^{i}\right] + \epsilon_{\rho_{s, k}^{i}}, 1 \leq k_m \leq m, \tag{15}
\]

where \(m\) is the smoothed interval of the Hatch filter, usually 20 to 200 epochs. \(\epsilon_{\rho_{s, k}^{i}} \sim \left(0, \sigma_{\rho_{s, k}^{i}}^2\right)\) denotes the multipath and random error of the carrier phase smoothed pseudorange measurement, which is also subject to the Zero-mean Gaussian distribution. According to Equation (15), the variance of the smoothed pseudorange is related to the variance of the pseudorange and carrier phase measurements. Assuming that the variance at different times is independent, then the variance of the smoothed pseudorange can be modeled as:

\[
\sigma_{\rho_{s, k}^{i}}^2 = \frac{1}{k_m} \sigma_{\rho_{s, k}^{i}}^2 + \left(1 - \frac{1}{k_m}\right) \left(\sigma_{\rho_{s, k}^{i}}^2 + \sigma_{\phi_{s, k}^{i}}^2\right), \tag{16}
\]

2.5. Factor Graph Representation

The factor graph of our system is shown in Figure 3, which shows the factors in a sliding window, including point feature factors, line feature factors, IMU factors, carrier phase smoothed pseudorange factors, and Doppler frequency shift factors. Visual observation consists of the point and line features detected by our system. In nonlinear optimization, the states are optimized according to the residuals of these factors. The states of our system include:

\[
\mathcal{X} = \left[ x_0, x_1, \cdots, x_n, \mu_0, \mu_1, \cdots, \mu_i, \varphi_0, \varphi_1, \cdots, \varphi_i, P_c^b, q_c^b, \Phi, P_n^E, P_r^E \right],
\]

\[
x_k = \left[ p_{w, b, k}, v_{w, b, k}, q_{w, b, k}, b_w, b_r, \delta t, \delta \Phi, \right], k = 0, 1, \cdots, n,
\]

\[
\sigma = [\psi, \phi], \tag{17}
\]

where \(n\) is the sliding window size, \(i\) is the number of point features, and \(j\) is the number of line features. \(\mu_i\) is the inverse depth of the \(i\)th point feature in the sliding window.
where \( p \) position of the receiver at time \( k \) is determined by the longitude \( L \) and the line feature residual can be obtained from Equation (10). In the following, we compute the smoothed pseudorange and Doppler frequency shift residuals in detail. The IMU preintegration and point feature residuals can be obtained according to [16], and the line feature residual can be obtained from Equation (10). In the following, we compute the smoothed pseudorange and Doppler frequency shift residuals in detail. The IMU preintegration and point feature residuals can be obtained according to [16], and the line feature residual can be obtained from Equation (10). In the following, we compute the smoothed pseudorange and Doppler frequency shift residuals in detail.

\[ r_p(\hat{z}_{k+1}^j, \mathcal{X}) = \frac{1}{k_m} \tilde{p}_{k+1}^j + \left( 1 - \frac{1}{k_m} \right) \left[ \tilde{p}_{k+1}^j + \Phi_{k-1}^j \right] - \tilde{p}_{k+1}^j \]

where \( p_{k+1}^w \) is the position of the receiver in the local world frame, and it can be modeled as:

\[ p_{k+1}^w = R_{N_k}^w b_k + p_{k+1}^w, \]

\( R_{N_k}^w \) is the rotation matrix of the local world frame with respect to the ENU frame. Since the two frames are gravity-aligned, the 1-DoF \( R_{N_k}^w \) is only related to the yaw offset \( \phi \) and can be modeled as:

\[ R_{N_k}^w = \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \]

\( R_{N_k}^w \) represents the rotation matrix of the ENU frame with respect to the ECEF frame, which is determined by the longitude \( L \) and latitude \( B \) of \( p_{N_k}^E \). Given a position \( p_{N_k}^E = [X, Y, Z]^T \) in the ECEF frame, it can be denoted in the geodetic frame as:

\[ \begin{align*}
L &= \arctan(Y/X) \\
B &= \arctan \left\{ Z(N_r + H) / \sqrt{X^2 + Y^2} (N_r(1 - e^2) + H) \right\} , \\
H &= Z / \sin B - N_r(1 - e^2) \\
N_r &= a / \sqrt{1 - e^2} \sin^2 B, \\
e^2 &= (a^2 - b^2) / a^2,
\end{align*} \]

where \( a \) and \( b \) denote the semimajor axis and the semiminor axis of the elliptical orbit, respectively. \( N_r \) is the radius of curvature in prime vertical, and \( e \) is the eccentricity that is related to \( a \) and \( b \).

After the longitude \( L \) and latitude \( B \) of \( p_{N_k}^E \) are computed, \( R_{N_k}^E \) can be obtained by:

\[ R_{N_k}^E = \begin{bmatrix} -\sin L & -\sin B \cos L & \cos B \cos L \\ \cos L & -\sin B \sin L & \cos B \sin L \\ 0 & \cos B & \sin B \end{bmatrix}, \]

According to Equation (15), the smoothed pseudorange residual can be represented as:

\[ r_p(\hat{z}_{k+1}^j, \mathcal{X}) = \frac{1}{k_m} \tilde{p}_{k+1}^j + \left( 1 - \frac{1}{k_m} \right) \left[ \tilde{p}_{k+1}^j + \Phi_{k-1}^j \right] - \tilde{p}_{k+1}^j \]

where \( R_{N_k}^E \) represents the rotation matrix of the ENU frame with respect to the ECEF frame, which is determined by the longitude \( L \) and latitude \( B \) of \( p_{N_k}^E \). Given a position \( p_{N_k}^E = [X, Y, Z]^T \) in the ECEF frame, it can be denoted in the geodetic frame as:

\[ \begin{align*}
L &= \arctan(Y/X) \\
B &= \arctan \left\{ Z(N_r + H) / \sqrt{X^2 + Y^2} (N_r(1 - e^2) + H) \right\} , \\
H &= Z / \sin B - N_r(1 - e^2) \\
N_r &= a / \sqrt{1 - e^2} \sin^2 B, \\
e^2 &= (a^2 - b^2) / a^2,
\end{align*} \]

where \( a \) and \( b \) denote the semimajor axis and the semiminor axis of the elliptical orbit, respectively. \( N_r \) is the radius of curvature in prime vertical, and \( e \) is the eccentricity that is related to \( a \) and \( b \).

After the longitude \( L \) and latitude \( B \) of \( p_{N_k}^E \) are computed, \( R_{N_k}^E \) can be obtained by:

\[ R_{N_k}^E = \begin{bmatrix} -\sin L & -\sin B \cos L & \cos B \cos L \\ \cos L & -\sin B \sin L & \cos B \sin L \\ 0 & \cos B & \sin B \end{bmatrix}, \]

According to Equation (15), the smoothed pseudorange residual can be represented as:

\[ r_p(\hat{z}_{k+1}^j, \mathcal{X}) = \frac{1}{k_m} \tilde{p}_{k+1}^j + \left( 1 - \frac{1}{k_m} \right) \left[ \tilde{p}_{k+1}^j + \Phi_{k-1}^j \right] - \tilde{p}_{k+1}^j \]
In nonlinear optimization, it is necessary to calculate the Jacobian matrix of the smoothed pseudorange residual with respect to the states. Therefore, from Equation (23), the Jacobian matrix of the smoothed pseudorange can be obtained as:

\[
J_p = \begin{bmatrix}
\frac{\partial \rho}{\partial p_{k-1}} & \frac{\partial \rho}{\partial p_k} & \frac{\partial \rho}{\partial p_{k+1}} & \cdots & \frac{\partial \rho}{\partial p_N} & \frac{\partial \rho}{\partial p_{N+1}}
\end{bmatrix}^T,
\]

(24)

with

\[
\frac{\partial \rho}{\partial p_{k-1}} = \left(1 - \frac{1}{k_m}\right) \frac{\left(p_{j_k}^E - p_{k-1}^E\right)}{\left|p_{j_k}^E - p_{k-1}^E\right|} R_N^E R_w^N,
\]

\[
\frac{\partial \rho}{\partial p_k} = \left(1 + \frac{1}{k_m}\right) \frac{\left(p_{j_k}^E - p_{k+1}^E\right)}{\left|p_{j_k}^E - p_{k+1}^E\right|} R_N^E R_w^N R_w^{-1} \left[p_r^b\right]_1,
\]

\[
\frac{\partial \rho}{\partial p_{k+1}} = -\frac{\left(p_{j_k}^E - p_{k+1}^E\right)}{\left|p_{j_k}^E - p_{k+1}^E\right|} R_N^E R_w^N R_w^{-1} \left[p_r^b\right]_1,
\]

\[
\frac{\partial \rho}{\partial \delta_{k-1}} = -c \left(1 - \frac{1}{k_m}\right),
\]

\[
\frac{\partial \rho}{\partial \delta_k} = \frac{\left(p_{j_k}^E - p_{k+1}^E\right)}{\left|p_{j_k}^E - p_{k+1}^E\right|} + \left(1 - \frac{1}{k_m}\right) \frac{\left(p_{j_k}^E - p_{k-1}^E\right)}{\left|p_{j_k}^E - p_{k-1}^E\right|},
\]

\[
\frac{\partial \rho}{\partial \delta_{k+1}} = -\frac{\left(p_{j_k}^E - p_{k+1}^E\right)}{\left|p_{j_k}^E - p_{k+1}^E\right|} R_N^E R_w^N R_w^{-1} + \left(1 - \frac{1}{k_m}\right) \frac{\left(p_{j_k}^E - p_{k-1}^E\right)}{\left|p_{j_k}^E - p_{k-1}^E\right|} R_N^E R_w^N R_w^{-1},
\]

(25)

where

\[
R_w = \frac{dR_w}{d\phi} = \begin{bmatrix}
-\sin \phi & -\cos \phi & 0 \\
\cos \phi & -\sin \phi & 0 \\
0 & 0 & 0
\end{bmatrix},
\]

(26)

The derivation details are provided in Appendix A.

In Equation (14), the velocity of the receiver in the ECEF frame is transformed from that in the local world frame:

\[
v_{r_k}^E \simeq R_N^E R_w^N v_{h_k}^w
\]

(27)

Therefore, the Doppler frequency shift residual can be obtained by:

\[
r_f(\tilde{\xi}_{r_k}, \lambda) = \frac{1}{\lambda} \left[k_k^T (v_{r_k}^E - v_{h_k}^E) + c \left(\delta t_k - \delta t_{h_k}\right) + \Delta f_{r_k}\right],
\]

(28)

Similar to the smoothed pseudorange, the Jacobian matrix of the Doppler frequency shift is:

\[
J_f = \begin{bmatrix}
\frac{\partial r_f}{\partial p_{k-1}} & \frac{\partial r_f}{\partial p_k} & \frac{\partial r_f}{\partial p_{k+1}} & \cdots & \frac{\partial r_f}{\partial p_N} & \frac{\partial r_f}{\partial p_{N+1}}
\end{bmatrix}^T
\]

(29)

with
when running in long trajectories. The observability of the system can be represented by

\[ \text{Observability matrix} \]

where the input and output are the external variables of the system, and the state is the
pseudorange and will not be repeated here.

For our proposed system, some simplifications were required. First, the accelerometer and
internal variables of the system. If the states of the system can be completely represented
by Ceres solver [25]. The cost function of our system is:

\[ C = \arg \min_{\Delta f} \sum_{(k,j) \in \Delta f} r_{\Delta f}(\hat{z}_{k|j}, \lambda) \]

The corresponding derivation rule is similar to the Jacobian matrix of the smoothed
pseudorange and will not be repeated here.

After the residuals of all factors are obtained, then the system can optimize the states
by Ceres solver [25]. The cost function of our system is:

\[ \min_{\Delta f} \left\{ \| r_{\Delta f}(\hat{z}_{k|j}, \lambda) \|^2 + \sum_{k \in B} \| r_{\Delta f}(\hat{z}_{k|j}, \lambda) \|^2 + \sum_{(l,j) \in C} a \left( \| r_{\Delta f}(\hat{z}_{l|j}, \lambda) \| \right)^2 \right\} \]

where \( r_p \) is the prior residual for marginalization. \( B \) is the set of IMU preintegration
measurements in the sliding window. \( C \) and \( L \) are the set of point and line features in the
sliding window, respectively. \( a \) is the Cauchy robust function used to suppress outliers.

2.6. GNSS-IMU Calibration

\( p^b_k \) is the extrinsic translation parameter between the GNSS receiver and IMU. After
our system performs ENU Origin Estimation and Yaw Estimation, then we can calibrate
\( p^b_k \). We estimate \( p^b_k \) as the initial value of nonlinear optimization through the following
optimization problem:

\[ \min_{p^b_k} \sum_{(k,j) \in \Delta f} \| r_{\Delta f}(\hat{z}_{k|j}, \lambda) \|^2 \]

After the GNSS-IMU calibration is successfully initialized, our system performs the
nonlinear optimization.

2.7. Observability Analysis of Tightly Coupled GNSS-VIO System

A SLAM system can be described using a state equation and an output equation,
where the input and output are the external variables of the system, and the state is the
internal variables of the system. If the states of the system can be completely represented
by the output, the system is fully observable; otherwise, the system is not fully observable.
Observability plays a very important role in the state estimation problem. If some states
of the system are not observable, the positioning precision of the system will be affected
when running in long trajectories. The observability of the system can be represented
by the observability matrix. If the dimension of the null space of the observability matrix is
equal to 0, then the system is fully observable. To facilitate the observability analysis of
our proposed system, some simplifications were required. First, the accelerometer
and gyroscope biases were not included in the states, because the biases were observable and
they did not change the results of the observability analysis [26]. Second, we considered a single point and line feature [27]. Third, the translation parameter $p^b_k$ was successfully calibrated. Then the discrete-time linear error state model and residual of the system are:

$$
\delta x_{k+1} \simeq \Phi_k \delta x_k + w_k, \\
r_k = H_k \delta x_k + n_k,
$$

where $\delta x_k$ is the error state, and $r_k$ is the residual. $\Phi_k$ and $H_k$ are the error-state transition matrix and the measurement Jacobian matrix, respectively. $w_k$ and $n_k$ represent the system noise process and the measurement noise process, respectively. The noise process is modeled as a Zero-mean white Gaussian process.

According to [28], the observability matrix can be obtained as:

$$
M = \begin{bmatrix}
H_k \\
H_{k+1} \Phi_k \\
\vdots \\
H_{k+t} \Phi_{k+t-1} \cdots \Phi_k
\end{bmatrix},
$$

In Equation (34), the observability matrix is defined as a function of the error-state transition matrix $\Phi_k$ and the measurement Jacobian matrix $H_k$.

Therefore, given the linearized system in Equation (33), its observability can be demonstrated according to Equation (34). The proof is as follows:

**Theorem 1.** The states represented in the ECEF frame are fully observable.

**Proof of Theorem 1.** The simplified states include:

$$
x^E_k = \begin{bmatrix}
p^E_{b_k}, v^E_{b_k}, q^E_{b_k}, \mu, \phi, p^E_N
\end{bmatrix},
$$

Generally, the raw accelerometer and gyroscope measurements from IMU can be obtained by:

$$
\tilde{a}_b = R^b_e(a_w + g_w) + b_a + n_a, \\
\tilde{\omega}_b = \omega_a + b_g + n_g,
$$

where $\tilde{a}_b$ and $\tilde{\omega}_b$ are the accelerometer and gyroscope measurements. The measurements are represented in the local world frame.

Given two time instants, position, velocity, and orientation states can be propagated by the IMU measurements:

$$
p^w_{b_{k+1}} = p^w_{b_k} + v^w_{b_k} \Delta t_k + \frac{1}{2} \int_{t \in (t_k, t_{k+1})} \left[ R^b_e (\tilde{a}_b - b_a) - g_w \right] dt^2, \\
v^w_{b_{k+1}} = v^w_{b_k} + \int_{t \in (t_k, t_{k+1})} \left[ R^b_e (\tilde{a}_b - b_a) - g_w \right] dt, \\
q^w_{b_{k+1}} = \int_{t \in (t_k, t_{k+1})} q^w_{b_k} \otimes \left[ \frac{1}{2} (\tilde{\omega}_b - b_g) \right] dt,
$$

where $\otimes$ denotes the quaternion multiplication operation.

Equation (37) is the continuous state propagation model. To analyze the observability of the system, it was necessary to perform the discretization of the continuous-time system model. Therefore, we used the Euler method to compute the discrete-time model of Equation (37):

$$
p^w_{b_{k+1}} = p^w_{b_k} + v^w_{b_k} \Delta t_k + \frac{1}{2} \left[ R^b_e (\tilde{a}_b - b_a) - g_w \right] \Delta t^2_k, \\
v^w_{b_{k+1}} = v^w_{b_k} + \left[ R^b_e (\tilde{a}_b - b_a) - g_w \right] \Delta t_k, \\
q^w_{b_{k+1}} = q^w_{b_k} \otimes \left[ \frac{1}{2} (\tilde{\omega}_b - b_g) \Delta t_k \right].
$$
According to Equation (38), the error propagation equation of position, velocity, and orientation can be obtained by:

\[
\begin{bmatrix}
\delta \mathbf{p}^E_{N(k+1)} \\
\delta \mathbf{v}^E_{N(k+1)} \\
\delta \theta_{k+1} \\
\delta \Psi_{k+1}
\end{bmatrix} =
\begin{bmatrix}
I & \Delta t_k & I & I \\
0 & I & 0 & 0 \\
0 & 0 & I & 0 \\
0 & 0 & 0 & I
\end{bmatrix}
\begin{bmatrix}
\delta \mathbf{p}^E_k \\
\delta \mathbf{v}^E_k \\
\delta \theta_k \\
\delta \Psi_k
\end{bmatrix}
+ \mathbf{w}'_{k+1}.
\]

(39)

The details of Equation (39) are provided in Appendix B.

Similarly, when the states are represented in the ECEF frame, Equation (39) is still held, namely:

\[
\begin{bmatrix}
\delta \mathbf{p}^E_{N(k+1)} \\
\delta \mathbf{v}^E_{N(k+1)} \\
\delta \theta_{N(k+1)} \\
\delta \Psi_{N(k+1)}
\end{bmatrix} =
\begin{bmatrix}
I & \Delta t_k & I & I \\
0 & I & 0 & 0 \\
0 & 0 & I & 0 \\
0 & 0 & 0 & I
\end{bmatrix}
\begin{bmatrix}
\delta \mathbf{p}^E_k \\
\delta \mathbf{v}^E_k \\
\delta \theta_k \\
\delta \Psi_k
\end{bmatrix}
+ \mathbf{w}_{k+1}.
\]

(40)

Therefore, the discrete-time error state model of Equation (35) can be obtained:

\[
\begin{bmatrix}
\delta \mathbf{x}^E_{k+1} \\
\delta \mathbf{x}^E_{k+1}
\end{bmatrix} =
\begin{bmatrix}
I & I & I & I \\
0 & I & 0 & 0 \\
0 & 0 & I & 0 \\
0 & 0 & 0 & I
\end{bmatrix}
\begin{bmatrix}
\delta \mathbf{x}^E_k \\
\delta \mathbf{x}^E_k \\
\delta \mathbf{x}^E_k \\
\delta \mathbf{x}^E_k
\end{bmatrix}
+ \mathbf{w}_{k+1}.
\]

(41)

where

\[
\begin{bmatrix}
\Phi^1_k = \frac{1}{2} \mathbf{R}^E_k \left[ \mathbf{	ilde{a}}_k - b_k \right] \times \Delta t^2_k \\
\Phi^2_k = \mathbf{R}^E_k \left[ \mathbf{	ilde{a}}_k - b_k \right] \times \Delta t_k \\
\Phi^3 = I - \mathbf{	ilde{a}}_k - b_k \times \Delta t_k 
\end{bmatrix}
\]

(42)

Since \( \Phi_k \) is an upper triangular matrix, the error-state transition matrix \( \Phi_{k+l,k} \) from time step \( k \) to \( k + l \) is also an upper triangular matrix, namely:

\[
\begin{bmatrix}
\Phi^1_{k+l} = \Phi^1_{k+l-1} \cdots \Phi^1_k \\
\Phi^2_{k+l} = \Phi^2_{k+l-1} \cdots \Phi^2_k \\
\Phi^3_{k+l} = \Phi^3_{k+l-1} \cdots \Phi^3_k 
\end{bmatrix}
\]

(43)

where \( \Phi^1_{k+l}, \Phi^2_{k+l}, \) and \( \Phi^3_{k+l} \) are nonzero entries.

For the measurement Jacobian matrix, our system consists of visual observations and GNSS measurements. GNSS measurements include pseudorange, carrier phase, and Doppler frequency shift, and any of them yields the same results for observability analysis. Therefore, in the following, we used the smoothed pseudorange measurement for observability analysis.

The Jacobian matrix of the smoothed pseudorange measurement with respect to the states in Equation (35) can be obtained by:

\[
\mathbf{H}^E_{N(k)} =
\begin{bmatrix}
p_{k+l} - p_{k+l}^{E_1} \\
p_{k+l} - p_{k+l}^{E_2} \\
p_{k+l} - p_{k+l}^{E_3}
\end{bmatrix}
\]

(44)
The derivation details are similar to the Jacobian matrix of the smoothed pseudorange, which has been computed in Appendix A. Since the smoothed pseudorange measurement does not include velocity, orientation, the inverse depth of point feature, and the orthonormal representation of line feature, the corresponding entry of the Jacobian matrix is equal to zero.

In addition, we can obtain the Jacobian matrix of the visual observations by:

\[
\mathbf{H}_v^{o} = \begin{bmatrix}
0_{2\times 9} & \frac{\partial r_r}{\partial \delta_{\mu_{k,t}}} & 0_{2\times 4} & 0_{2\times 4} \\
0_{2\times 9} & 0_{2\times 1} & 0_{2\times 4} & 0_{2\times 4}
\end{bmatrix}
\]  

(45)

where \( \frac{\partial r_r}{\partial \delta_{\mu_{k,t}}} \) and \( \frac{\partial r_r}{\partial v_{k,t}} \) can be obtained from [10].

Therefore, we can obtain the entry of the observability matrix:

\[
\begin{bmatrix}
\mathbf{H}_v^{o} \\
\mathbf{H}_k^{o}
\end{bmatrix} \mathbf{\Phi}_{k+t,k} = \begin{bmatrix}
0_{4\times 3} & 0_{4\times 3} & 0_{4\times 3} & 0_{4\times 3} \\
h_4^{k+t} & h_4^{k+t} & h_4^{k+t} & h_4^{k+t} \\
0_{3\times 5} & 0_{3\times 5} & 0_{3\times 5} & 0_{3\times 5} \\
h_5^{k+t} & h_5^{k+t} & h_5^{k+t} & h_5^{k+t}
\end{bmatrix}
\]

(46)

where

\[
\mathbf{\Pi}' = \begin{bmatrix}
0_{2\times 4} \\
0_{2\times 1}
\end{bmatrix}
\]  

(47)

According to Equation (46), we can clearly observe that the dimension of the null space of \( \mathbf{M} \) is equal to zero, which means the states represented in the ECEF frame are fully observable and the tightly coupled GNSS-VIO state estimator is consistent. \( \square \)

The fact that the tightly coupled GNSS-VIO system is fully observable means that even if the system runs in long trajectories, the accumulated error can be eliminated. By leveraging the global measurements from GNSS, our system can achieve high-precision and drift-free positioning compared with the VIO system, which has four unobservable directions.

3. System Overview

The architecture of our proposed system is shown in Figure 4.

![Figure 4](image)

**Figure 4.** A block diagram illustrating the workflow of our proposed system.

The proposed system implements four threads including data input, preprocessing, initialization, and nonlinear optimization. As shown in Figure 4, the white block diagrams represent the work that has been implemented by VINS-Mono [16] and GVINS [21], and the green block diagrams represent improvements we made. The inputs of our system are
image, IMU, and GNSS measurements. In the preprocessing step, point and line feature detection and tracking were performed, IMU measurements were preintegrated, and the pseudorange measurements were smoothed by the carrier phase. In the initialization step, we followed the routine of VINS-Mono [16] for VI-Alignment. After VI-Alignment was completed, we performed GNSS initialization, which was divided into four stages: ENU Origin Estimation (Coarse), Yaw Estimation, ENU Origin Estimation (Fine), and GNSS-IMU Calibration. The first three stages were implemented in GVINS [21]. Finally, nonlinear optimization was performed. Nonlinear optimization will optimize the states in Equation (17) by leveraging the residuals and Jacobian matrices of different factors, which were computed in Section 2.5.

4. Experimental Results

Our experiments were conducted on the public dataset GVINS-Dataset [29], which captured scenes from the Hong Kong University of Science and Technology. The measurements were collected by a helmet that is equipped with a VI-Sensor and a u-blox ZED-F9P GNSS receiver. The dataset sports field captured a sports field scene where the device followed an athletic track for five laps. The sports field is an outdoor environment with an open area where the satellites are well locked and the RTK solution remains fixed. The other dataset complex environment was a complex indoor–outdoor environment where many challenging scenes were captured. For example, point and line features cannot be detected in bright or dim scenes, and the GNSS signal was highly corrupted or unavailable in cluttered or indoor environments (about 25 m). The overall distance of the complex environment dataset was over 3 km. We compared our proposed system with some open-source SLAM systems, including GVINS and VINS-Mono where VINS-Mono includes results with and without loop closure. For comparison with our results, we transformed the trajectories of VINS-Mono from the local world frame to the ECEF frame. Furthermore, since RTK has centimeter-level positioning precision, we compared it with our trajectories as the ground truth.

4.1. Sports Field

We plotted the trajectories of the sports field on Google Earth as shown in Figure 5. We see that VINS-Mono without loop closure suffers from accumulated drift among all three directions, which leads to the worst positioning precision among methods. Obviously, the drift increases with each lap around the sports field. VINS-Mono with loop closure can significantly improve the positioning precision, but there is still an obvious drift in yaw direction, which is mainly because the VINS-Mono has four unobservable directions. As a comparison, since GVINS is fully observable in the ECEF frame, it has a smaller positioning error and is drift-free. As depicted in Figure 5, its trajectory is very close to RTK. In addition, since our method is also a tightly coupled GNSS-VIO system, which is fully observable in the ECEF frame and the ENU frame, the positioning result of our method is also very accurate.

To quantitatively analyze results, we compared the positioning errors of GVINS and our method in detail. As shown in Figure 6, we visualized the positioning errors of GVINS and our proposed method. We see that the results of GVINS and our system fluctuate within a small range in the ENU frame and ECEF frame. The positioning errors of our method are less than 1m among all three directions of the ENU frame. Compared with GVINS, the error of our system is smaller in the east and north directions. In addition, the error of our method is smooth and stable in the ECEF frame, which is because we smooth the pseudorange measurements with the carrier phase. In addition, we listed the RMSE and Spherical Error Probable (SEP) for the positioning error of GVINS and our method in the ENU frame and ECEF frame. From Table 1, we can see that the RMSE of our positioning errors is lower than that of GVINS except in the up direction. The SEP refers to the radius of a sphere in which 50% of the estimated positions occur, and it can also evaluate the positioning precision of our system. The smaller the SEP, the higher the
positioning precision of our system. We see that the SEP of our method in the ECEF frame is lower than that of GVINS, which shows that the positioning precision of our method is more accurate in the ECEF frame.

Figure 5. The trajectories of VINS-Mono, GVINS, RTK, and our proposed method in the sports field.

Figure 6. The positioning errors of GVINS and our proposed system in the sports field. The left three rows are the positioning errors in the east, north, and up directions of the ENU frame. The right three rows are the positioning errors in the X, Y, and Z directions of the ECEF frame.

Table 1. RMSE[m] and SEP[m] statistic of GVINS and our proposed method in the sports field.

| ENU Frame | RMSE | SEP | ECEF Frame | RMSE | SEP |
|-----------|------|-----|------------|------|-----|
|           | East | North | Up | X | Y | Z |       |
| GVINS     | 0.661 | 0.477 | 0.296 | 0.834 | 0.766 | 0.581 | 1.002 |
| Proposed  | 0.524 | 0.360 | 0.725 | 0.522 | 0.440 | 0.466 | 0.651 |

4.2. Complex Environment

In the following, we conducted experiments on the dataset complex environment, and we compared VINS-Mono, GVINS, RTK, and our proposed method. As shown in Figure 7, since VINS-Mono has four unobservable directions, the trajectory of VINS-Mono has a
large drift. In addition, we saw that a section of the trajectory of VINS-Mono deviated from RTK by about 34 m, which is an unacceptable result. On the contrary, since GVINS and our proposed system are tightly coupled GNSS-VIO systems, there is no accumulated drift theoretically. In addition, due to a section of the trajectory being collected indoors, the GNSS receiver cannot obtain measurements at this time, which causes the failure of the RTK solution. In contrast, even if the GNSS measurements cannot be obtained, our system works well with VIO, and our trajectory is still accurate.

4.2. Complex Environment

In the following, we conducted experiments on the dataset complex environment, and we compared VINS-Mono, GVINS, RTK, and our proposed method. As shown in Figure 7, since VINS-Mono has four unobservable directions, the trajectory of VINS-Mono has a large drift. In addition, we saw that a section of the trajectory of VINS-Mono deviated from RTK by about 34 m, which is an unacceptable result. On the contrary, since GVINS and our proposed system are tightly coupled GNSS-VIO systems, there is no accumulated drift theoretically. In addition, due to a section of the trajectory being collected indoors, the GNSS receiver cannot obtain measurements at this time, which causes the failure of the RTK solution. In contrast, even if the GNSS measurements cannot be obtained, our system works well with VIO, and our trajectory is still accurate.

![Figure 7. The trajectories of VINS-Mono, GVINS, RTK, and our proposed method in complex environments.](image)

To further compare the performance of GVINS and our method, we analyzed the positioning errors of the two methods in the dataset complex environment as shown in Figure 8. It should be noted that we compared the RTK solution with our method as the ground truth. However, when the receiver is occluded, the satellite signal cannot be received, and the RTK solution is useless at this time. Thus, we only compared with segments where the RTK solution is available. Due to the complexity of the dataset, the overall positioning precision of the system is inferior to that of the sports field. The RMSE and SEP of GVINS and our method are shown in Table 2. We see that the positioning errors of GVINS in the east and north directions are slightly lower than that of our method, while the positioning error of our method in the up direction is much lower than that of GVINS. The SEP metric also shows that our method outperforms GVINS on this dataset.
Figure 8. The positioning errors of GVINS and our proposed system in the complex environment.

| ENU Frame  | RMSE      | SEP   |
|------------|-----------|-------|
| East       | North     | Up    |       |
| GVINS      | 1.091     | 0.782 | 3.016 | 2.540 |
| Proposed   | 1.230     | 1.404 | 2.499 | 2.390 |

Furthermore, since the ENU frame is gravity-aligned, we further analyzed the 2D trajectory error in the East–North plane as shown in Figure 9. The trajectory projected on the East–North plane is stained in red. The z-axis represents the positioning errors in the corresponding East–North plane, and the errors are represented by a scatter plot. In some bright or dim scenes, only a few point and line features can be detected, which will affect the positioning precision of the system. In addition, buildings, indoor environments, and trees will block satellite signals, which also degrade the performance of our system. Therefore, the positioning error in the complex environment is larger than that in the sports field, and the maximum error exceeds 2 m. As depicted in Figure 9, a sudden increase in positioning error occurs at the curves of the trajectory due to the large change in the orientation direction. On the contrary, the positioning error of the trajectory with small curvature is smoother compared with that of the trajectory with large curvature, which shows that our method is still effective even in complex scenes.

4.3. Observability Analysis

In Section 2.7, we demonstrated that the states represented in the ECEF are fully observable. However, according to [20], if the states are represented in the local world frame, the GNSS-VIO system has still four unobservable directions. Therefore, we can perform observability analysis from experimental results. Figure 10 shows the 3D trajectories of the complex environment represented in the ECEF frame and local world frame. We zoomed in on the trajectories of turning and climbing stairs for the convenience of the analysis. From Figure 10b, we see that since the states are unobservable in the local world frame, the two trajectories of climbing the same stairs have a drift. Even if the GNSS measurements are included in our system, the two trajectories are still inconsistent. In contrast, due to the states represented in the ECEF frame being fully observable, the two trajectories of climbing the same stairs are consistent and very close. Similarly, the same conclusions can be obtained for the trajectories of turning as shown in Figure 10. According to the results
of the observability analysis, there is an obvious drift in the two trajectories of turning in the local world frame, while the phenomenon does not occur in the ECEF frame. In addition, the consistency of the system also shows that even in large-scale scenarios, it can still achieve high-precision positioning results as illustrated in Figure 10a.

![Figure 9. The 2D trajectory error of the complex environment on the East–North plane.](image)

![Figure 10. Comparison of the 3D trajectories represented in different frames. (a) is the ECEF frame and (b) is the local world frame.](image)

4.4. Smoothed Interval

Equation (15) requires the receiver to continuously lock the carrier phase. If the receiver loses lock, the smoothed pseudorange is corrupted by any cycle slip that occurs and must be reinitialized when that happens. Therefore, it is necessary to investigate the influence of different smoothed intervals on the positioning precision of our system. The positioning error with six different settings is illustrated in Figure 11. When the smoothed interval is set to 200 epochs, the system has the largest positioning error in the three directions of the ENU frame. This happens because the receiver loses lock frequently within a smoothed interval and then cycle slip occurs, which causes an increase in the positioning error. By reducing the smoothed interval, the phenomenon that the receiver loses lock can be eliminated, which can improve the positioning precision of the carrier phase smoothed pseudorange. Obviously, with the smoothed intervals of 5, 10, 20, and 50 epochs, our system has smaller errors in east and north directions compared to the smoothed intervals of 100 or 200 epochs. However, if the smoothed interval is too small, the positioning precision in up direction cannot be further improved, and may even become worse. As shown in Figure 11, when the smoothed interval is set to 5 epochs, the positioning error in the up direction is the largest among different settings. The reason for this phenomenon
is that when the smoothed interval is too small, the carrier phase cannot smooth the pseudorange well, which causes the smoothed pseudorange measurement to degenerate into the pseudorange measurement.

![The positioning errors of different smoothed interval](image)

**Figure 11.** The positioning errors with different smoothed interval settings ($m = 5, 10, 20, 50, 100, 200$).

### 4.4. Smoothed Interval

Equation (15) requires the receiver to use the carrier phase smoothed pseudorange as an observation. However, when the smoothed interval is too small, the carrier phase cannot smooth the pseudorange well, which causes the smoothed pseudorange measurement to degenerate into the pseudorange measurement.

### 4.5. Line Feature Tracking Threshold

In the nonlinear optimization stage, if the number of keyframes for which line features are continuously observed is less than a threshold, then these line features are filtered out. A different threshold has a great impact on the robustness and positioning precision. As shown in Figure 12, a threshold of 2 means that those line segments that are observed by two consecutive keyframes or more can be added to the factor graph. However, since these line segments are continuously tracked for too few frames, they are not stable enough for nonlinear optimization, which will definitely reduce the positioning precision. On the contrary, if the line feature tracking threshold varies between 3 and 4, the proposed system can obtain more accurate positioning results in the ENU frame. Because the line segments observed by three or four consecutive keyframes are more stable and more suitable for nonlinear optimization.

![The positioning errors of different threshold](image)

**Figure 12.** The positioning errors of different line feature tracking thresholds.
5. Conclusions

In this paper, we propose a tightly coupled GNSS-VIO system based on point-line features. First, since line features contain more environmental structure information, we added line features to the system. Second, the pseudorange measurement can only achieve meter-level positioning precision but is fast and unambiguous. On the contrary, the carrier phase measurement can achieve the centimeter-level positioning precision but is required to solve the whole-cycle ambiguity, which is time-consuming. Therefore, we propose to combine the advantages of the two measurements and replace the pseudorange with the carrier phase smoothed pseudorange, which can greatly improve the performance of our system. Third, we considered the extrinsic translation parameter between the GNSS receiver and IMU, and our system can perform real-time parameter calibration. Finally, we demonstrated that if the states are represented in the ECEF frame, the tightly coupled GNSS-VIO system is fully observable. We conducted experiments on public datasets, which show our system achieves high-precision, robust, and real-time localization. In the future, we will further focus on improved methods for tightly coupled GNSS-VIO systems.

Author Contributions: Conceptualization, B.D. and K.Z.; methodology, B.D. and K.Z.; validation, B.D. and K.Z.; investigation, B.D.; writing—original draft preparation, B.D.; writing—review and editing, B.D. and K.Z.; All authors have read and agreed to the published version of the manuscript.

Funding: This research work was supported by the Key-Area Research and Development Program of Guangdong Province (No. 2020B0909050003).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

To derive the Jacobian matrix of the smoothed pseudorange residual with respect to the states, Equation (15) is expanded into:

\[ \frac{\partial \rho_j}{\partial \rho_{k-1}} = \frac{1}{\rho_{k-1}} \left( \left\| \mathbf{p}_j^E - \mathbf{p}_{k-1}^E \right\| + c \left( \delta t_k - \delta t_j \right) + T_k^j + I_k \right) 
+ \left( 1 - \frac{1}{\kappa_m} \right) \left\{ \frac{\partial \rho_j}{\partial \rho_{k-1}} + \left\| \mathbf{p}_j^E - \mathbf{p}_{k-1}^E \right\| + c \left( \delta t_k - \delta t_j \right) + T_k^j + I_k \right) + \lambda N_k \]  \tag{A1}

The derivations can be obtained by the chain rule. For the translation \( \mathbf{p}_{k-1}^w \) and rotation matrix \( \mathbf{R}_{k-1}^w \), we can calculate their derivations by:

\[ \frac{\partial \mathbf{r}_j}{\partial \mathbf{p}_{k-1}^w} = \frac{\partial \mathbf{r}_j}{\partial \mathbf{p}_{k-1}^w} = \left( 1 - \frac{1}{\kappa_m} \right) \left( \mathbf{p}_j^E - \mathbf{p}_{k-1}^E \right)^T \cdot \mathbf{R}_{k-1}^w \mathbf{R}_{k-1}^N \cdot \mathbf{I}, \]

\[ \frac{\partial \mathbf{r}_j}{\partial \mathbf{R}_{k-1}^w} = \frac{\partial \mathbf{r}_j}{\partial \mathbf{R}_{k-1}^w} = \left( 1 - \frac{1}{\kappa_m} \right) \left( \mathbf{p}_j^E - \mathbf{p}_{k-1}^E \right)^T \cdot \mathbf{R}_{k-1}^w \mathbf{R}_{k-1}^N \cdot \left( -\mathbf{R}_{k-1}^w \left[ \mathbf{p}_j^E \right] \times \right), \]  \tag{A2}
The derivation of the states at time $k$ is similar to that at time $k-1$ and will not be repeated here. In the following we introduce the Jacobian matrix of the smoothed pseudorange with respect to the yaw offset $\varphi$:

\[
\frac{\partial r_p}{\partial \varphi} = \frac{1}{\kappa_m} \frac{\partial \mathbf{e}_p}{\partial \varphi} + \left(1 - \frac{1}{\kappa_m}\right) \left[ \frac{\partial \mathbf{e}_p}{\partial \varphi} - \frac{\partial \mathbf{e}_p}{\partial \varphi} \right]
\]

\[
= \frac{1}{\kappa_m} \frac{\partial \mathbf{e}_p}{\partial \varphi} - \left(1 - \frac{1}{\kappa_m}\right) \frac{\partial \mathbf{e}_p}{\partial \varphi}
\]

\[
= -\left( p_{E}^E - p_{k-1}^E \right) \cdot \mathbf{R}_N^E \mathbf{R}_w^E \mathbf{p}_w^E + \left(1 - \frac{1}{\kappa_m}\right) \left( p_{E}^E - p_{k-1}^E \right) \mathbf{R}_N^E \mathbf{R}_w^E \mathbf{p}_w^E',
\]

where

\[
\mathbf{R}_w^E = \frac{d \mathbf{R}_w^E}{d \varphi} = \begin{bmatrix} -\sin \varphi & -\cos \varphi & 0 \\ \cos \varphi & -\sin \varphi & 0 \\ 0 & 0 & 0 \end{bmatrix},
\]

The Jacobian matrix of the smoothed pseudorange with respect to the translation $\mathbf{p}_N^E$ can be calculated by:

\[
\frac{\partial r_p}{\partial \mathbf{p}_N^E} = \frac{1}{\kappa_m} \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_N^E} + \left(1 - \frac{1}{\kappa_m}\right) \left[ \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_N^E} - \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_N^E} \right]
\]

\[
= \frac{1}{\kappa_m} \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_N^E} - \left(1 - \frac{1}{\kappa_m}\right) \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_N^E}
\]

\[
= -\left( p_{E}^E - p_{k-1}^E \right) \cdot \mathbf{I} + \left(1 - \frac{1}{\kappa_m}\right) \left( p_{E}^E - p_{k-1}^E \right) \mathbf{I}
\]

Similarly, the Jacobian matrix of the smoothed pseudorange with respect to the extrinsic translation parameter $\mathbf{p}_w^E$ can be calculated by:

\[
\frac{\partial r_p}{\partial \mathbf{p}_w^E} = \frac{1}{\kappa_m} \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_w^E} + \left(1 - \frac{1}{\kappa_m}\right) \left[ \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_w^E} - \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_w^E} \right]
\]

\[
= \frac{1}{\kappa_m} \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_w^E} - \left(1 - \frac{1}{\kappa_m}\right) \frac{\partial \mathbf{e}_p}{\partial \mathbf{p}_w^E}
\]

\[
= -\left( p_{E}^E - p_{k-1}^E \right) \cdot \mathbf{R}_N^E \mathbf{R}_w^E \mathbf{p}_w^E + \left(1 - \frac{1}{\kappa_m}\right) \left( p_{E}^E - p_{k-1}^E \right) \mathbf{R}_N^E \mathbf{R}_w^E \mathbf{p}_w^E',
\]

Appendix B

We rewrite the discrete-time propagation equation of position in Equation (38):

\[
\mathbf{p}_{E}^E_{k+1} = \mathbf{p}_{E}^E_k + \mathbf{v}_{E}^E_k \Delta t_k + \frac{1}{2} \left[ \mathbf{R}_{E}^E_k \left( \mathbf{a}_{E} - \mathbf{b}_k \right) - \mathbf{g}_{E} \right] \Delta t_k^2,
\]

(7)
To derive the derivation of the position at time $k + 1$ with respect to the orientation at time $k$, we add a tiny perturbation to the rotation matrix at time $k$ by:

$$
\frac{\partial \mathbf{p}_{b,k+1}^w}{\partial \delta \theta_{b,k+1}^w} = \frac{1}{2} \Delta t_k^2 \frac{\partial}{\partial \delta \theta_{b,k}^w} \lim_{\delta \theta_{b,k}^w \to 0} R_k^w \exp \left( \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) \left( \mathbf{a}_{b,k} - \mathbf{b}_k \right) - R_k^w \left( \mathbf{a}_{b,k} - \mathbf{b}_k \right)
$$

$$
\approx \frac{1}{2} \Delta t_k^2 \frac{\partial}{\partial \theta_{b,k}^w} \left[ R_k^w \left( \mathbf{a}_{b,k} - \mathbf{b}_k \right) - R_k^w \left( \mathbf{a}_{b,k} - \mathbf{b}_k \right) \right]
$$

$$
= \frac{1}{2} \Delta t_k^2 \frac{\partial}{\partial \theta_{b,k}^w} \left[ R_k^w \left( \mathbf{a}_{b,k} - \mathbf{b}_k \right) \right] - \frac{1}{2} R_k^w \left( \mathbf{a}_{b,k} - \mathbf{b}_k \right) \Delta t_k^2
$$

(A8)

The derivation of the orientation at time $k + 1$ with respect to the orientation at time $k$ is similar to the routine of Equation (A8). We can also add a tiny perturbation to both sides of the orientation propagation equation by:

$$
\mathbf{q}_{b_{k+1}}^w \approx \left[ \begin{array}{c} \frac{1}{2} \delta \theta_{b_{k+1}}^w \\
R_{b_{k+1}}^w \end{array} \right] = \mathbf{q}_{b_k}^w \otimes \left[ \begin{array}{c} \frac{1}{2} \delta \theta_{b_k}^w \\
R_{b_k}^w \end{array} \right] \otimes \left[ \begin{array}{c} \frac{1}{2} \left( \mathbf{\omega}_{b_k} - \mathbf{b}_g \right) \Delta t_k \\
1 \end{array} \right],
$$

(A9)

then

$$
\left[ \begin{array}{c} \frac{1}{2} \delta \theta_{b_{k+1}}^w \\
R_{b_{k+1}}^w \end{array} \right] = \left( \mathbf{q}_{b_{k+1}}^w \right)^* \otimes \mathbf{q}_{b_k}^w \otimes \left[ \begin{array}{c} \frac{1}{2} \delta \theta_{b_k}^w \\
R_{b_k}^w \end{array} \right] \otimes \left[ \begin{array}{c} \frac{1}{2} \left( \mathbf{\omega}_{b_k} - \mathbf{b}_g \right) \Delta t_k \\
1 \end{array} \right]
$$

(A10)

$$
= \left[ \begin{array}{c} \frac{1}{2} \mathbf{R} \delta \theta_{b_k}^w \\
- \frac{1}{2} \left( \mathbf{\omega}_{b_k} - \mathbf{b}_g \right) \Delta t_k \end{array} \right]
$$

where $\mathbf{R} = \exp \left( - \left[ \mathbf{\omega}_{b_k} - \mathbf{b}_g \right] \times \Delta t_k \right)$. Considering only the imaginary part of Equation (A10), then we have:

$$
\delta \theta_{b_{k+1}}^w = \mathbf{R} \delta \theta_{b_k}^w
$$

$$
= \exp \left( - \left[ \mathbf{\omega}_{b_k} - \mathbf{b}_g \right] \times \Delta t_k \right) \delta \theta_{b_k}^w
$$

$$
\approx \left[ 1 - \left[ \mathbf{\omega}_{b_k} - \mathbf{b}_g \right] \times \Delta t_k \right] \delta \theta_{b_k}^w
$$

(A11)

Therefore, according to Equation (A11), we can obtain:

$$
\frac{\partial \delta \theta_{b_{k+1}}^w}{\partial \delta \theta_{b_k}^w} = 1 - \left[ \mathbf{\omega}_{b_k} - \mathbf{b}_g \right] \times \Delta t_k
$$

(A12)

References

1. Weiss, S.; Siegwart, R. Real-time metric state estimation for modular vision-inertial systems. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 4531–4537.
2. Kneip, L.; Weiss, S.; Siegwart, R. Deterministic initialization of metric state estimation filters for loosely-coupled monocular vision-inertial systems. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), San Francisco, CA, USA, 25–30 September 2011; pp. 2235–2241.
3. Bloesch, M.; Burri, M.; Omari, S.; Hutter, M.; Siegwart, R. Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback. Int. J. Robot. Res. 2017, 36, 1053–1072. [CrossRef]
4. Leutenegger, S.; Lynen, S.; Bosse, M.; Siegwart, R.; Furgale, P. Keyframe-based visual–inertial odometry using nonlinear optimization. Int. J. Robot. Res. 2015, 34, 314–334. [CrossRef]
5. Davison, A.J.; Reid, I.D.; Molton, N.D. MonoSLAM: Real-time single camera SLAM. IEEE Trans. Pattern Anal. Mach. Intell. 2007, 29, 1052–1067. [CrossRef] [PubMed]
6. Shi, J. Good features to track. In Proceedings of the 1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 21–23 June 1994; pp. 4531–4537.
7. Klein, G.; Murray, D. Parallel tracking and mapping for small AR workspaces. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, 13–16 November 2007; pp. 225–234.
8. Mur-Artal, R.; Montiel, J.M.M.; Tardós, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015,** *31,* 1147–1163. [CrossRef]
9. Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017,** *33,* 1255–1262. [CrossRef]
10. He, Y.; Zhao, J.; Guo, Y.; He, W.; Yuan, K. PL-VIO: Tightly-Coupled Monocular Visual–Inertial Odometry Using Point and Line Features. *Sensors* **2018,** *18,* 1159. [CrossRef] [PubMed]
11. Von, G.; Jakubowicz, J.; Morel, J.M.; Randall, G. LSD: A fast line segment detector with a false detection control. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008,** *32,* 722–732.
12. Kaehler, A.; Bradski, G. *Learning OpenCV 3: Computer vision in C++ with the OpenCV Library*; O’Reilly Media, Inc.: Sebastopol, CA, USA, 2016.
13. Yang, Y.; Geneva, P.; Eckenhoft, K.; Huang, G. Visual-inertial odometry with point and line features. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 2447–2454.
14. Fu, Q.; Wang, J.; Yu, H.; Ali, I.; Guo, F.; He, Y.; Zhang, H. PL-VINS: Real-time monocular visual-inertial SLAM with point and line features. *arXiv* **2020,** arXiv:2009.07462.
15. Lee, J.; Park, S.Y. PLF-VINS: Real-time monocular visual-inertial SLAM with point-line fusion and parallel-line fusion. *IEEE Robot. Autom. Lett.* **2021,** *6,* 7033–7040. [CrossRef]
16. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018,** *34,* 1004–1020. [CrossRef]
17. Mourikis, A.I.; Roumeliotis, S.I. A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA), Roma, Italy, 10–14 April 2007; pp. 3565–3572.
18. Li, M.; Mourikis, A.I. Improving the accuracy of EKF-based visual-inertial odometry. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA), St Paul, MN, USA, 14–18 May 2012; pp. 828–835.
19. Campos, C.; Elvira, R.; Rodriguez, J.J.G.; Montiel, J.M.; Tardós, J.D. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robot.* **2021,** *37,* 1874–1890. [CrossRef]
20. Lee, W.; Eckenhoft, K.; Geneva, P.; Huang, G. Intermittent gps-aided vio: Online initialization and calibration. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–June 2020; pp. 5724–5731.
21. Cao, S.; Lu, X.; Shen, S. GVINS: Tightly Coupled GNSS–Visual–Inertial Fusion for Smooth and Consistent State Estimation. *IEEE Trans. Robot.* **2022,** *1–18.* [CrossRef]
22. Li, X.; Wang, X.; Liao, J.; Li, X.; Li, S.; Lyu, H. Semi-tightly coupled integration of multi-GNSS PPP and S-VINS for precise positioning in GNSS-challenged environments. *Satell. Navig.* **2021,** *2,* 1. [CrossRef]
23. Liu, J.; Gao, W.; Hu, Z. Optimization-based visual-inertial SLAM tightly coupled with raw GNSS measurements. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi’an, China, 30 May–5 June 2021; pp. 11612–11618.
24. Zhang, G.; Lee, J.H.; Lim, J.; Suh, I.H. Building a 3-D Line-Based Map Using Stereo SLAM. *IEEE Trans. Robot.* **2015,** *31,* 1364–1377. [CrossRef]
25. Agarwal, S.; Mierle, K. Ceres Solver. Available online: http://ceres-solver.org (accessed on 25 April 2022).
26. Jones, E.S.; Soatto, S. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *Int. J. Robot. Res.* **2011,** *30,* 407–430. [CrossRef]
27. Roumeliotis, S.I.; Kottas, D.G.; Guo, C.; Hesch, J. Observability-Constrained Vision-Aided Inertial Navigation. U.S. Patent 9,243,916, 26 January 2016.
28. Li, M.; Mourikis, A.I. High-precision, consistent EKF-based visual-inertial odometry. *Int. J. Robot. Res.* **2013,** *32,* 690–711. [CrossRef]
29. Cao, S.; Lu, X. GVINS-Dataset. Available online: https://github.com/HKUST-Aerial-Robotics/GVINS-Dataset (accessed on 25 April 2022).