drones

MDPI

Article

Range–Visual–Inertial Odometry with Coarse-to-Fine Image Registration Fusion for UAV Localization

Yun Hao [1,*], Mengfan He [1], Yuzhen Liu [2], Jiacheng Liu [3] and Ziyang Meng [1]

[1] Department of Precision Instrument, Tsinghua University, Beijing 100084, China; hmf21@mails.tsinghua.edu.cn (M.H.); ziyangmeng@mail.tsinghua.edu.cn (Z.M.)
[2] Robotics X, Tencent, Shenzhen 518057, China; rickyyzliu@tencent.com
[3] Unmanned Aerial Vehicle Lab, Meituan, Beijing 100012, China; liujiacheng13@meituan.com
* Correspondence: haoy17@mails.tsinghua.edu.cn

Abstract: In Global Navigation Satellite System (GNSS)-denied environments, image registration has emerged as a prominent approach to utilize visual information for estimating the position of Unmanned Aerial Vehicles (UAVs). However, traditional image-registration-based localization methods encounter limitations, such as strong dependence on the prior initial position information. In this paper, we propose a systematic method for UAV geo-localization. In particular, an efficient range–visual–inertial odometry (RVIO) is proposed to provide local tracking, which utilizes measurements from a 1D Laser Range Finder (LRF) to suppress scale drift in the odometry. To overcome the differences in seasons, lighting conditions, and other factors between satellite and UAV images, we propose an image-registration-based geo-localization method in a coarse-to-fine manner that utilizes the powerful representation ability of Convolutional Neural Networks (CNNs). Furthermore, to ensure the accuracy of global optimization, we propose an adaptive weight assignment method based on the evaluation of the quality of image-registration-based localization. The proposed method is extensively evaluated in both synthetic and real-world environments. The results demonstrate that the proposed method achieves global drift-free estimation, enabling UAVs to accurately localize themselves in GNSS-denied environments.

Keywords: range–visual–inertial odometry; coarse-to-fine image registration; localization

Citation: Hao, Y.; He, M.; Liu, Y.; Liu, J.; Meng, Z. Range–Visual–Inertial Odometry with Coarse-to-Fine Image Registration Fusion for UAV Localization. *Drones* **2023**, *7*, 540. https://doi.org/10.3390/drones7080540

Academic Editor: Giordano Teza

Received: 9 August 2023
Accepted: 19 August 2023
Published: 21 August 2023

1. Introduction

Owing to small size, high agility, and low cost, UAVs have been widely applied in search and rescue [1], environmental monitoring [2], aerial photography, and other fields [3–5]. However, to successfully carry out these tasks, precise localization of UAVs is of utmost importance. Although the GNSS is the most commonly used localization system, providing the geographic coordinates of UAVs, its reliability is compromised by challenges such as multipath reception [6] and Non-Line-Of-Sight (NLOS) reception [7], which can introduce inaccuracies. In addition, the increasing threat of signal spoofing poses a significant risk to UAVs, particularly in defense and security applications [8]. Therefore, it is essential to develop methods for locating UAVs in GNSS-denied environments.

In GNSS-denied environments, UAVs can utilize onboard sensors, including cameras and Inertial Measurement Units (IMUs), to infer their position without depending on external infrastructure. These two sensors can be integrated to form Visual-Inertial Odometry (VIO), which estimates the UAV's position and orientation by processing data acquired from both the camera and the IMU. However, when the UAV moves in uniform linear motion, the acceleration is constant, resulting in VIO's inability to estimate the scale. To address this issue, an alternative approach involves incorporating a 1D LRF, which is a lightweight and accurate sensor that assists VIO in observing scale information. On the other hand, it is imperative to emphasize that without global information, odometry methods are susceptible to drift and can only provide the local pose of the UAV rather than a

more generic global pose. This limitation highlights the need for additional techniques to mitigate drift and obtain a more generic global pose estimation for UAVs operating in GNSS-denied environments.

Recently, image-registration-based localization has been used to attain the global pose of a UAV [9–12]. This method involves registering the images taken from the UAV to the pre-existing maps, thereby enabling the acquisition of accurate global positioning information for the UAV. The most-used pre-existing maps are orthorectified satellite imagery, which is easily accessible and GNSS-aligned. However, the primary challenge associated with this method lies in matching images from different sources. These images often exhibit variations in perspective, lighting conditions, seasons, and other factors, making traditional image matching methods such as template matching and feature matching susceptible to failure [13].

In this study, we present a novel method for performing global localization by utilizing RVIO to track UAV motion, registering UAV images to geo-referenced satellite imagery, and fusing measurements with global pose graph optimization. As depicted in Figure 1, our method comprises two modules: the RVIO module and the image-registration-based geo-localization module. The RVIO module utilizes a downward-facing monocular camera, an LRF, and an IMU to estimate the relative motion between consecutive frames of the UAV. The image-registration-based geo-localization module comprises two steps: coarse matching and fine matching. During the coarse matching step, aided by the LRF, the suitable map tiles are retrieved from the database generated through satellite image segmentation. These suitable map tiles serve as potential candidates for further registration. Subsequently, in the fine matching step, the image captured by the UAV is registered with the selected candidates from the coarse matching step. This registration process enables us to acquire global position measurements for the UAV. To obtain a globally consistent localization estimate, the UAV ego-motion measurements derived from RVIO and the global position measurements are fused through a global pose graph optimization.

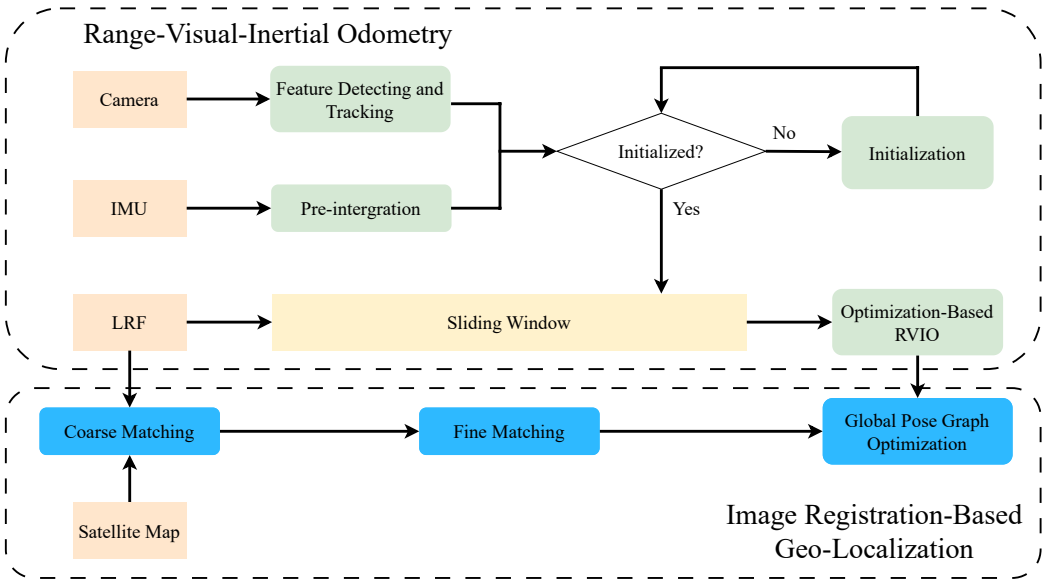


Figure 1. Pipeline of the proposed localization method.

This work makes the following contributions:

- We propose an optimization-based RVIO method which utilizes range measurements to effectively suppress scale drift from VIO, thereby enhancing localization accuracy.
- We propose a coarse-to-fine image-registration-based localization method that provides the global pose of the UAV. With the assistance of LRF, the retrieval efficiency of the coarse matching step is improved. Then, the fine matching step calculates the accurate geographic position of the UAV.

- The proposed method is evaluated on both synthesized and real-world datasets. The results demonstrate the effectiveness of our method.

The rest of this paper is organized as follows. In Section 2, we discuss the related works. Section 3 illustrates the proposed method used for UAV geo-localization. In Section 4, we describe the experiments we conducted on both synthesized and real-world datasets. The conclusions are summarized in Section 5.

2. Related Works

2.1. Range–Visual–Inertial Odometry

VIO frameworks can be classified into two main categories: tightly coupled and loosely coupled methods. Delmerico et al. [14] conducted a comprehensive investigation into various VIO frameworks and reported that the tightly coupled approach exhibited superior performance compared with the loosely coupled alternative. Tightly coupled VIO frameworks can be further divided into two categories: filter-based algorithms and optimization-based algorithms. Examples of the former include MSCKF [15], ROVIO [16], etc. However, filter-based algorithms have a theoretical limitation: they must linearize nonlinear measurements before processing, which can lead to significant linearization errors that may affect the accuracy of the estimation. In contrast, optimization-based algorithms [17–20] can perform relinearization during each iteration step, resulting in more accurate estimates and are, therefore, generally preferred.

In addition, range sensors can effectively reduce the scale drift of odometry. Various methods have been proposed using 2D or 3D LiDARs [21], RGB-D cameras [22], or stereo cameras, among others. However, the size and measurement distance of these sensors can limit their applicability to UAVs. Conversely, 1D LRFs offer a compelling solution, as they are compact, lightweight, and well-suited for integration with most UAV platforms. For instance, Giubilato et al. [23] used an LRF to recover and maintain the correct metric scale for visual odometry. Based on this, they proposed a range-enhanced monocular system and an extrinsic calibration pipeline [24]. For VIO applications, Delaune et al. [25] proposed an RVIO system that leverages a range measurement update model to assign depth to features and eliminate scale drift. Hu et al. [26] extended the RVIO method by fully exploiting distance measurements to constrain all coplanar features. Simultaneously, they conducted online extrinsic calibration between the LRF and camera. However, these methods only constrain the depth of feature points, without taking into account the constraint of UAV flight altitude. Furthermore, unlike the filter-based methods they use, our proposed method implements RVIO using an optimization-based framework.

2.2. Image-Registration-Based Localization

There have been various methods developed for UAV image registration and localization in GNSS-denied environments. As reviewed in [13], the works in the field of image registration and localization can be divided into three classes depending on the different image matching technologies: template matching, feature matching, and deep learning matching.

The key to the template matching method is the choice of a suitable similarity metric function. In early works, the Sum of Squared Differences (SSD) was the chosen metric, which involves a comparison of the luminance of each pixel [27]. However, it is sensitive to changes in scenarios. Recent works utilize more robust similarity measures, such as Mutual Information (MI) [28], Normalized Cross-Correlation (NCC) [29], and Normalized Information Distance (NID) [30]. Nevertheless, the template matching method is inefficient, and the computational burden is correlated to the size of the satellite image.

In addition, feature matching stands as another approach to accomplish image matching. The typical feature descriptors include SIFT [31], ORB [32], etc. However, the performance of these traditional descriptors decreases in the case of larger scenario changes and geometric transformations.

Semantic features have been used to accomplish the image-matching task. Nassar et al. [33] utilized semantic segmentation to extract the meaningful shapes of both UAV images and satellite imagery. A semantic shape-matching pipeline is performed to find correspondences between UAV images and satellite imagery. Choi et al. [34] constructed a feature descriptor called the building ratio feature by extracting the building areas from images. The matching algorithm based on the building ratio feature provides the global position estimation to correct the local odometry. Xu et al. [35] proposed a point–line–patch feature descriptor with rotation and scaling invariance. Based on the descriptors, the geo-localization is calculated and refined through the application of the Inverse Compositional Lucas–Kanade (ICLK) algorithm. However, the semantic features require the presence of buildings or roads in the scene, which limits the practical applications.

Moreover, utilizing the high-level features from deep learning methods is a natural choice. Goforth et al. [36] employed a deep CNN with an ICLK layer to align UAV images with satellite images. Then, they developed an optimization to refine the UAV's pose. Chen et al. [37] proposed a two-stage image-based geo-localization pipeline that adapts to downward-tilted camera configurations. However, this method requires offline preparation of the dataset in advance. Kinnari et al. [38] performed orthorectification of the UAV image based on VIO and planarity assumption, which makes the UAV camera orientation not strictly required. The ortho-projected image is used to match with the satellite imagery for geo-localization, and the results and the tracking pose from VIO are fused in a particle filter framework. Further, they used learned features to replace the classical image-matching metrics to obtain seasonally invariant localization [39]. However, the works mentioned above directly employ the satellite map with suitable regions and resolutions, while in practice, the area covered by satellite images is much larger than that of UAV images. Successfully aligning UAV images with larger satellite maps becomes an impractical task. Therefore, the interested area needs to be retrieved in the satellite image. In contrast to the aforementioned methods, we present a novel image-registration-based geo-localization approach that follows a coarse-to-fine strategy, eliminating the need for any initial prior information. Moreover, our method fuses both global position measurements and local position measurements obtained from RVIO using a global pose graph optimization. This fusion process facilitates the attainment of a globally consistent localization estimate.

3. Method

In this section, we elaborates on the specifics of our proposed localization method. The method comprises the RVIO module and the image-registration-based geo-localization module that executes the ensuing operations: (1) estimating the local pose of the UAV using RVIO, (2) registering the UAV image with satellite imagery for geo-localization, and (3) fusing the global position measurements and local pose through a global pose graph optimization to yield the global pose estimation of the UAV.

3.1. Range–Visual–Inertial Odometry

3.1.1. Visual–Inertial Odometry

The architecture of our RVIO is based on [17], which is a sliding-window keyframe-based nonlinear optimization framework. All states in the sliding window are defined as

$$\gamma_{VIO} = [\gamma_I, \gamma_\lambda], \tag{1}$$

where $\gamma_\lambda = [\lambda_0, \lambda_1, \ldots, \lambda_n]$ represents the inverse depth of the landmarks upon their initial observation in the camera frame, and $\gamma_I = [\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_m]$ represents m IMU states. Specifically, the k-th IMU state is defined as the UAV's position ${}^L\boldsymbol{p}_{I_k} = [x_{I_k}, y_{I_k}, z_{I_k}]^T$, velocity ${}^L\boldsymbol{v}_{I_k}$, orientation quaternion ${}^L\boldsymbol{q}_{I_k}$, and IMU bias $\boldsymbol{b}_a, \boldsymbol{b}_g$. The transformation between the IMU frame $\{I\}$ and the camera frame $\{C\}$, as well as the transformation between $\{C\}$ and the LRF frame $\{F\}$, have been previously calibrated and are regarded as known. Without loss of generality, these frames are considered to be coincident in what follows.

For each camera image, features are detected and tracked between consecutive frames using an optical flow algorithm [40]. Meanwhile, the IMU measurements between two consecutive frames undergo preintegration [41]. After initialization, all the measurements are placed in a sliding window for nonlinear optimization. The states γ_{VIO} are estimated through the minimization of the cost function, which takes into account the marginalization information, inertial residuals, and visual residuals. Specifically, the inertial residuals $\boldsymbol{e}_I^k$ are calculated by leveraging IMU preintegration between two successive frames within the sliding window. The visual residuals $\boldsymbol{e}_C^{h,j}$ delineate the reprojection error by reprojecting the landmark ${}^L\boldsymbol{P}_h$ onto keyframe K_j and subsequently comparing it with the original raw visual measurements $\hat{\boldsymbol{z}}_{h,j}$. $\boldsymbol{e}_P$ stands as the marginalization residuals that encompass information pertaining to previously marginalized states. The cost function is expressed as follows:

$$J_{VIO} = \sum_{h,j \in \mathcal{C}} \left\| \boldsymbol{e}_C^{h,j} \right\|^2_{\boldsymbol{W}_C^{h,j}} + \sum_{k \in \mathcal{I}} \left\| \boldsymbol{e}_I^k \right\|^2_{\boldsymbol{W}_I^k} + \left\| \boldsymbol{e}_P \right\|^2, \tag{2}$$

where $\mathcal{I}$ represents the set of all IMU measurements, while $\mathcal{C}$ represents the set of all features that have been observed at least twice within the sliding window. To calculate the norm of the residuals, the Mahalanobis distance weighted by covariance $\boldsymbol{W}$ is used. The notation $|| \cdot ||_{\boldsymbol{W}}$ denotes the Mahalanobis distance.

3.1.2. Range Measurements

Whenever a new frame is added to the sliding window, the most recent range measurement $\boldsymbol{d}_i^F$ is stored. As the optical center of the camera and the origin of the LRF are coincident, the range measurement can be represented as a 3D landmark of depth d_i^F in the camera frame $\{C\}$. In the ideal scenario, LRF would emit rays for measurement. However, due to the detection angle, the LRF emits a rectangular spot of light instead. The size of the spot varies at different distances, with the spot becoming larger as the distance increases. If the detected object's length fall short of the dimensions of the detection area's edges, the LRF is unable to provide valid data. Therefore, we adopt the assumption that the detection area is locally flat when the data from the LRF is valid.

Figure 2 illustrates the association of range measurement and features. Based on the flat assumption, the features associated with a range measurement are those for which the corresponding landmark is within the detection area. Let $\boldsymbol{f}_i^F$ denote the projection of the range measurement onto the image plane. We can formulate the range measurement as $\boldsymbol{d}_i^F = \left[\boldsymbol{f}_i^F, d_i^F\right]^T$, where $\boldsymbol{f}_i^F = [u_i, v_i]$ is the pixel coordinate of the projection point. Each associated feature $\boldsymbol{f}_j = \left[u_j, v_j\right]$ must satisfy the following formula:

$$\begin{cases} ||u_j - u_i|| < d_u \\ ||v_j - v_i|| < d_v, \end{cases} \tag{3}$$

where d_u and d_v are determined by edge length of the detection area of the LRF. To avoid incorrect constraints from landmarks that do not satisfy the planar assumption, we perform a standard deviation test. Only features that meet the following condition are optimized:

$$\sqrt{\frac{1}{n_i} \sum_{j=1}^{n_i} ||d_j - \mu||^2} < \sigma, \tag{4}$$

where n_i is the number of associated features, d_j is the depth of the associated landmark, μ is the average depth of the associated landmarks, and σ is an empirical threshold.

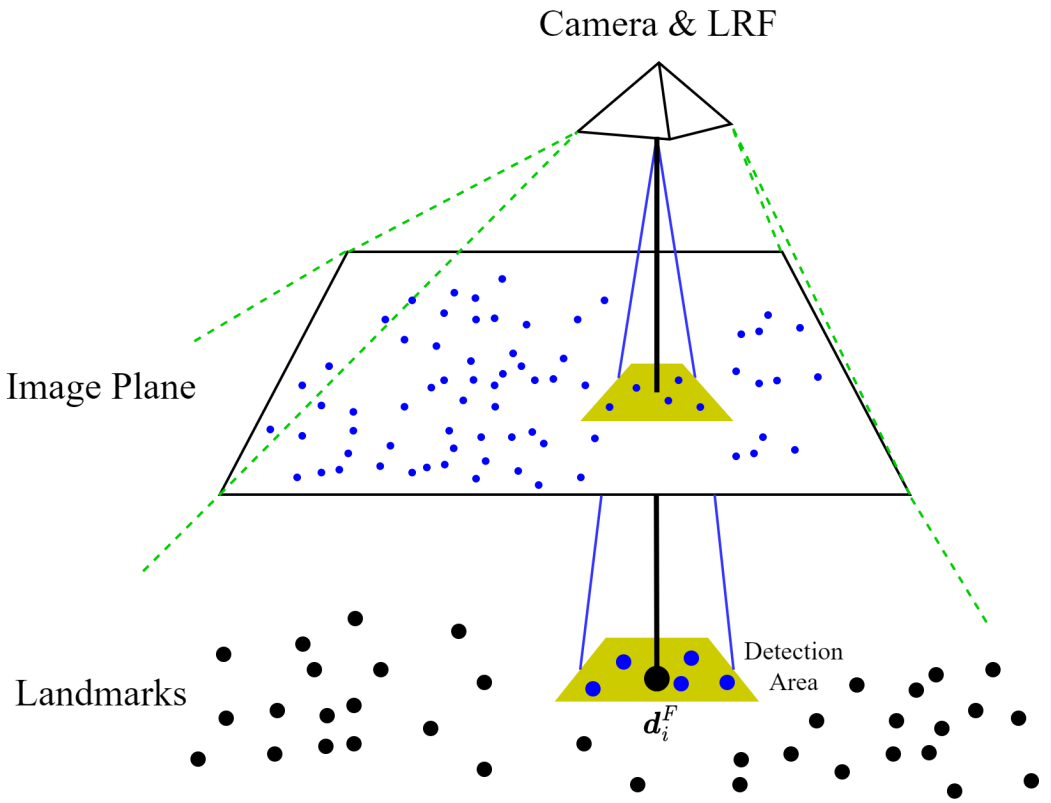


Figure 2. Association of features and range measurement: The blue points and black points represent the features and their corresponding landmarks, respectively. Range measurement is interpreted as a 3D landmark in $\{C\}$. The yellow trapezoid represents the detection area of the LRF. The depth of the landmarks within the detection area is constrained by the range measurements.

3.1.3. Joint Optimization

VIO triangulation suffers from scale ambiguity. By the range measurements from LRF, there is an additional constraint for the depth of the landmarks. Therefore, we can formulate the residuals of the j-th feature in the i-th frame as

$$\boldsymbol{e}_D^{i,j} = d_i^F - \frac{1}{\lambda_j}. \tag{5}$$

Furthermore, the downward-facing LRF can measure the height of the UAV directly. By setting the first measurement as the origin height, subsequent measurements of height can be obtained. The height residuals $\boldsymbol{e}_H^i$ are expressed as

$$\boldsymbol{e}_H^i = d_i^F - h^L(\boldsymbol{x}_i), \tag{6}$$

where $h^L(\boldsymbol{x}_i) = z_{I_i}$.

To fuse all the measurements in the sliding window, we construct a pose graph, as shown in Figure 3. The cost function (2) needs to be extended with additional terms, resulting in the following equation:

$$J_{RVIO} = J_{VIO} + \sum_{i \in \mathcal{F}} \|\boldsymbol{e}_H^i\|^2_{\boldsymbol{W}_H^i} + \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{D}} \|\boldsymbol{e}_D^{i,j}\|^2_{\boldsymbol{W}_D^{i,j}}, \tag{7}$$

where $\mathcal{F}$ represents the set of all range measurements within the sliding window, and $\mathcal{D}$ denotes the set of landmarks for which the depth d_j is constrained by the range measurements d_i^F. To solve this nonlinear problem, we employ the Google Ceres Solver [42], which utilizes the Levenberg–Marquadt approach.

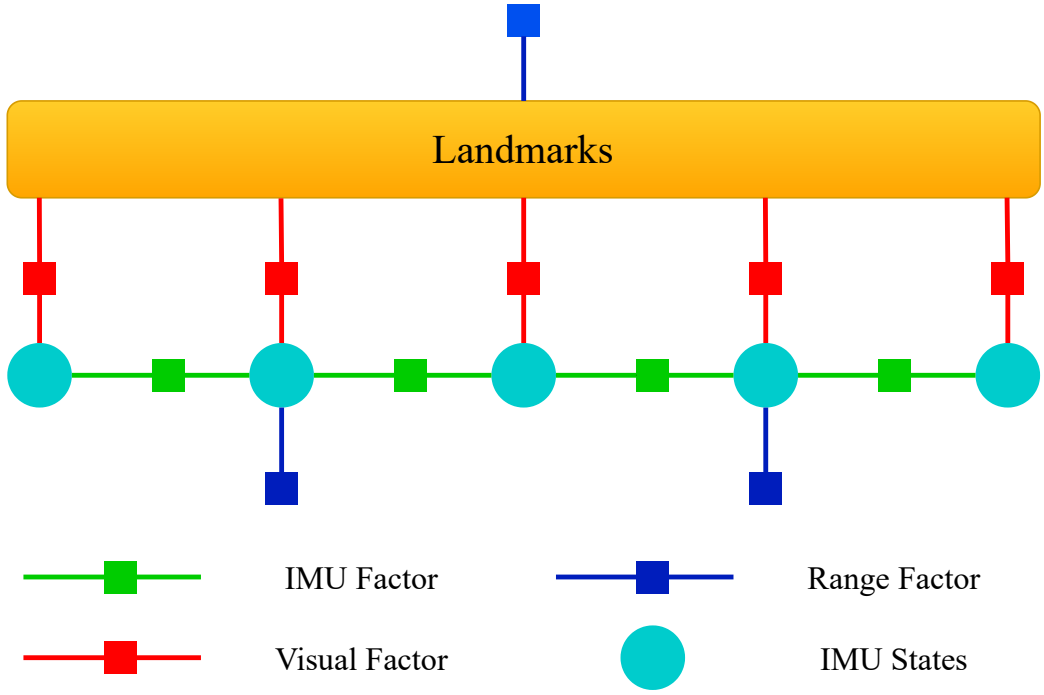


Figure 3. Structure of the pose graph: The orange block represents the landmarks that are observed. The cyan circles represent the IMU states in the sliding window. We distinguish three types of factors: visual (red), inertial (green), and range (blue).

3.2. Coarse-to-Fine Image-Registration-Based Localization

3.2.1. Coarse Matching

Registering UAV imagery directly within large satellite imagery is a highly challenging task. Therefore, it is necessary to crop the satellite imagery into smaller map tiles. These map tiles are intentionally cropped with suitable overlap to ensure a successful registration. The goal of coarse matching is to identify the most appropriate map tile from the database of cropped map tiles for a subsequent fine matching step.

Rather than storing the original images of the map tiles, we extract the descriptors from the tiles to construct a database. We use NetVLAD [43] to extract the descriptors of the map tiles. NetVLAD is a deep learning implementation of the VLAD [44] process, which extracts convolutional feature maps and estimates both the cluster centers and the residual vectors in an end-to-end manner. We encode each map tile as a 4096-dimensional descriptor and add it to the database. Furthermore, we include two geographical coordinates for each map tile in the database, representing the latitude and longitude of the upper-left corner and lower-right corner of the map tile, respectively. Since the area of the map tile is small, we can posit a linear correlation between the pixel coordinates and the corresponding geographical coordinates. Therefore, the geographical coordinates of each point within the map tile can be calculated utilizing the two geographical coordinates stored within the database. For a $W \times H$ pixels map tile, the geographical coordinates (lat, lon) of point (c_x, c_y) are calculated as

$$\begin{aligned} lon &= c_x \cdot \frac{lon_r - lon_l}{W} + lon_l \\ lat &= c_y \cdot \frac{lat_r - lat_l}{H} + lat_l, \end{aligned} \tag{8}$$

where (lat_l, lon_l) and (lat_r, lon_r) represent the geographical coordinates of the upper-left corner and lower-right corner of the map tile, respectively.

It is noteworthy to mention that the UAV operates at different heights, resulting in varying ranges of the photographed areas. These differences in scale pose a challenge to the matching process. To address this issue, it is necessary to store map tiles of different sizes

that simulate the different heights at which the UAV operates. However, including additional data of varying heights in the database not only expands the vertical working range of the UAV but also increases the complexity of retrieval. Consequently, to enhance retrieval efficiency, we leverage LRF measurements to limit the search to the data corresponding to the height obtained from the LRF measurements stored in the database.

Similar to the construction of the map tile database, NetVLAD is also utilized to extract the descriptor of the UAV image. The similarities between the descriptor of the UAV image and every descriptor in the database are computed and sorted in descending order. To identify potential candidates for further registration, we select the top n similarities from the sorted list. It should be noted that the selected similarities must undergo a threshold test. Only the map tiles associated with the selected similarities that exceed the predefined threshold are considered as candidates for the subsequent fine matching step. Empirically, we set $n = 3$ for the experiment.

3.2.2. Fine Matching

After the coarse matching step, we need to find the correct retrieval from the candidates. We rerank the candidates by using local feature extraction and matching. For robustness in handling scene changes, we select SuperPoint [45] as the local descriptor. To match the local descriptors extracted from two images, we leverage SuperGlue [46] due to its exceptional proficiency in matching features in significantly different images. The map tile with maximum number of matched points is considered the best candidate and is used in the subsequent registration process.

Consequently, we can establish the correspondences between the local keypoints of the UAV image and those of the best candidate. As the UAV images are captured by a downward-facing camera and the satellite imagery is orthorectified, we can describe the position of UAV using the planar homography by flat-world assumption. The homography transformation is calculated using the correspondences, and RANdom SAmple Consensus (RANSAC) is applied to avoid outliers. The homography transformation allows us to compute the corresponding point to the center of the UAV image in the best candidate map tile, from which we can obtain the geographical coordinates of the UAV by Equation (8). Then, we convert the geographical coordinate into ENU (East North Up) coordinates, representing the global position measurement denoted as ${}^{G}\hat{\boldsymbol{p}}_{I_i} = [{}^{G}x_{I_i}, {}^{G}y_{I_i}, 0]^T$.

However, due to the vibration of the UAV, we cannot guarantee that the camera remains strictly pointing vertically downward during UAV flight, which results in deviations of the global position measurements. For instance, at a flight height of 150 m, an angle of 5° results in a deviation of 13 m. Therefore, we compensate for the deviation to obtain more accurate global positions. During the initialization procedure of RVIO, the z-axis of the local RVIO frame, denoted as $\{L\}$ (the first camera frame), is aligned with the direction of the gravity. We utilize the intersection vector between the plane where the LRF measurement hits and is perpendicular to the direction of gravity and the plane formed by the optical axis and gravity to compensate for the deviation

3.2.3. Global Pose Graph Optimization

Once we obtain the geographical coordinates, we fuse them with the local odometry estimates from RVIO into a global pose graph optimization scheme. The estimated local poses of the UAV are added to the global pose graph and serve as vertices after the RVIO process. Each vertex is connected to others by a sequential factor. Upon successful registration of the UAV image to the map tile, we obtain the UAV's global position to derive a global positional factor. This enables us to incorporate the global factors to the global pose graph. Moreover, the global frame $\{G\}$ where the global position measurements are located and the local RVIO frame $\{L\}$ need to be aligned. We introduce the transformation ${}^{G}\boldsymbol{T}_{L} = [{}^{G}\boldsymbol{q}_{L}, {}^{G}\boldsymbol{p}_{L}]$ into the states. As a result, the states subjected to optimization are defined as

$$\gamma = \left[{}^{L}\boldsymbol{p}_{I_0}, {}^{L}\boldsymbol{q}_{I_0}, \ldots, {}^{L}\boldsymbol{p}_{I_n}, {}^{L}\boldsymbol{q}_{I_n}, {}^{G}\boldsymbol{p}_{L}, {}^{G}\boldsymbol{q}_{L}\right], \tag{9}$$

where n is the number of all poses of frames that are included in the graph. Figure 4 illustrates the structure of the global pose graph.

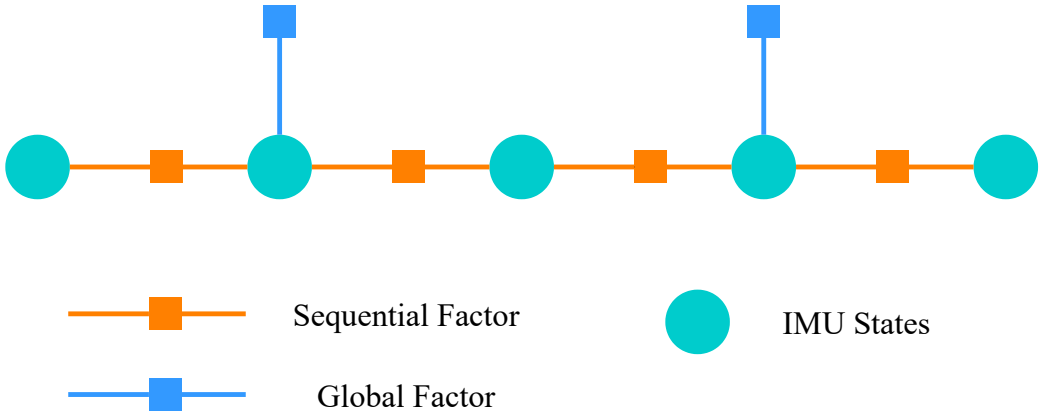


Figure 4. Structure of the global pose graph: The states are represented by cyan circles, with the local factor edge connecting two consecutive nodes and the other edge representing the global factor.

The sequential factor between two frames $i+1$ and i is derived using their relative pose. The residual of the sequential factor between frames $i+1$ and i can be expressed as

$$\boldsymbol{e}_S^i({}^L\boldsymbol{p}_{I_i}, {}^L\boldsymbol{q}_{I_i}, {}^L\boldsymbol{p}_{I_{i+1}}, {}^L\boldsymbol{q}_{I_{i+1}}) = \begin{bmatrix} {}^L\boldsymbol{R}_{I_{i+1}}^{-1}({}^L\boldsymbol{p}_{I_i} - {}^L\boldsymbol{p}_{I_{i+1}}) - {}^{I_{i+1}}\hat{\boldsymbol{p}}_{I_i} \\ {}^{I_{i+1}}\hat{\boldsymbol{q}}_{I_i}\,{}^L\boldsymbol{q}_{I_i}^{-1}\,{}^L\boldsymbol{q}_{I_{i+1}} \end{bmatrix}, \tag{10}$$

where the notation $\hat{(\cdot)}$ represents the noisy measurement. The residual of the global factor for frame i is defined as

$$\boldsymbol{e}_G^i({}^L\boldsymbol{p}_{I_i}, {}^L\boldsymbol{q}_{I_i}, {}^G\boldsymbol{p}_L, {}^G\boldsymbol{q}_L) = {}^G\boldsymbol{R}_L\,{}^L\boldsymbol{p}_{I_i} - {}^G\hat{\boldsymbol{p}}_{I_i} + {}^G\boldsymbol{p}_L. \tag{11}$$

Hence, the cost function can be formulated as

$$J = \sum_{i,i+1\in\mathcal{S}} \|\boldsymbol{e}_S^i\|^2_{\boldsymbol{W}_S^{i,i+1}} + \sum_{i\in\mathcal{G}} \|\boldsymbol{e}_G^i\|^2_{\boldsymbol{W}_G^i}, \tag{12}$$

where $\mathcal{S}$ denotes the set of all sequential factors, and $\mathcal{G}$ denotes the set of all global factors.

3.2.4. Image Registration Evaluation

In order to effectively perform the global pose graph optimization, it is crucial to evaluate the quality of the global position measurements derived from image registration. We, therefore propose an evaluation strategy that considers four factors jointly: the number of matched points ψ_n, the angle between the optical axis of the camera and the direction of gravity ψ_a, the discrepancy between the global position measurement and the previous estimation from the optimization ψ_d, and prior information ψ_p (inherent errors in satellite imagery, etc.).

We define the quality of the global position measurement as

$$Q = \beta_n\psi_n + \beta_a\psi_a + \beta_d\psi_d + \beta_p\psi_p, \tag{13}$$

where β_n, β_a, β_d, and β_p are the balance factors. Consequently, the residual weights of the global factor $\boldsymbol{W}_G^i$ can be represented in terms of the quality Q as $\boldsymbol{W}_G^i = Q \cdot \boldsymbol{I}$. This approach ensures that the optimization process takes into account the quality of the global position measurements to obtain more accurate results.

4. Experiments

In this section, we conduct evaluations of the proposed method on both synthesized dataset and real-world datasets. The implementation of our proposed method relies on the open-source framework VINS-Fusion [47]. All experiments were performed on an Intel NUC Mini PC equipped with a 4.70 GHz Intel Core i7 processor and Nvidia Geforce RTX 2060 discrete graphics.

4.1. Synthesized Dataset

4.1.1. Setup

To validate the proposed method, we develop a simulation environment using the Unreal Engine 4 as the development engine. We import real physics world data into the engine using the Cesium for Unreal plugin. The data comprises 3D topographic scenes with realistic geographic coordinates. To collect data in the simulation environments, we use AirSim [48] to control a UAV equipped with multiple sensors, including a downward-facing camera, IMU, and a downward-facing LRF. All the measurements are recorded through a Robot Operating System (ROS). GNSS data are recorded as the ground truth.

4.1.2. RVIO Performance

To evaluate the performance of our proposed RVIO, we compare it with other methods, including the method without range measurements (marked as VIO [17]) and the method that only constrains height by the range measurements (marked as HVIO [47]). We conduct the evaluation on sequence 1, which is collected in the simulation environment. The trajectories of different methods on this sequence are shown in Figure 5. All estimated trajectories are aligned with the ground truth using [49]. As shown in Figure 6, the estimated height of UAV from VIO is inaccurate, while the heights estimated by HVIO and RVIO are closer to the ground truth. We also calculated the root mean square error of the Absolute Trajectory Error (ATE) on sequence 1, and the results are presented in Table 1. The results demonstrate that RVIO outperforms HVIO. These findings suggest that the constraints provided by the range measurements for the depth of landmarks can improve localization performance.

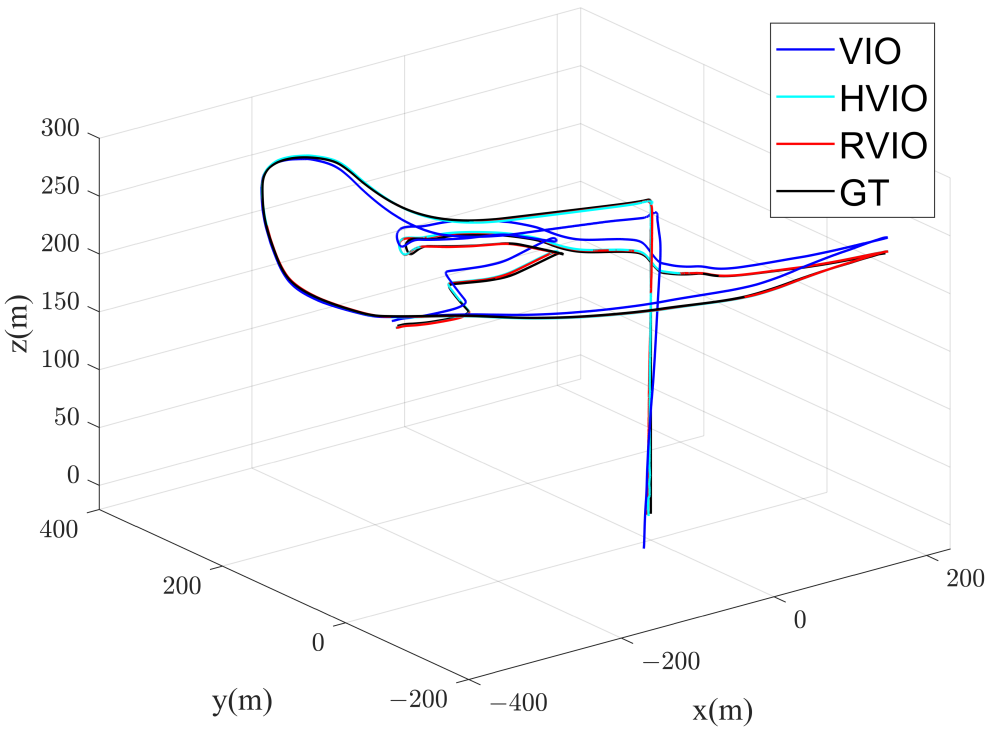


Figure 5. Trajectories estimated by different methods on sequence 1.

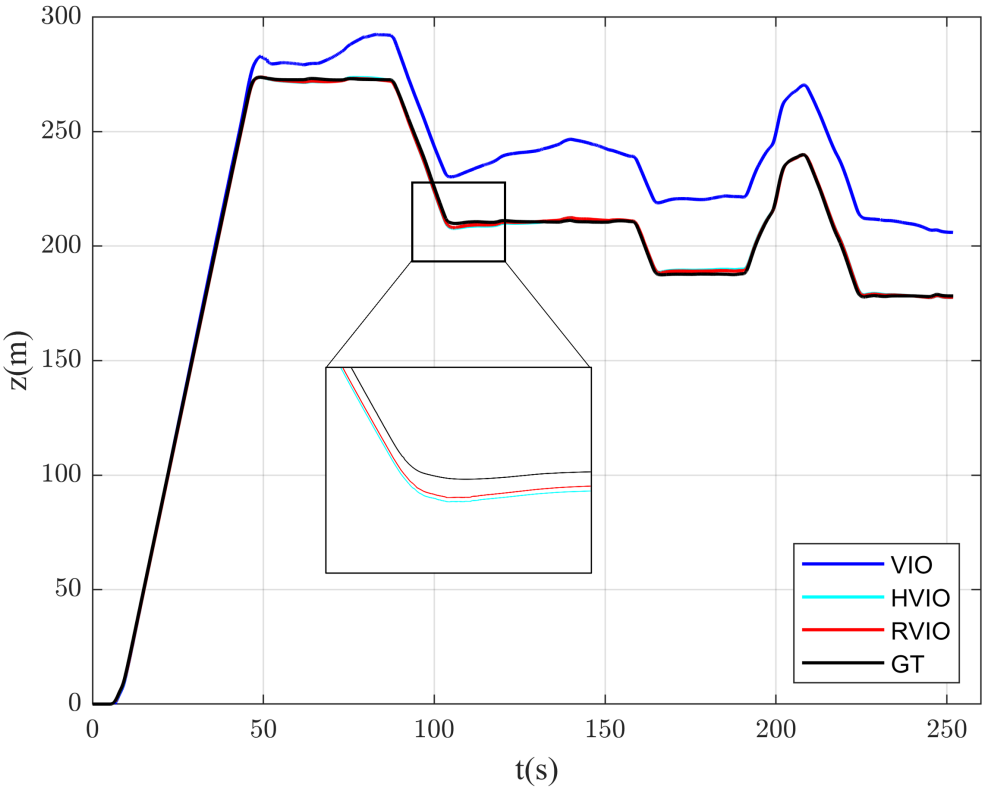


Figure 6. The estimated heights of different methods of the UAV during sequence 1 in comparison with the ground-truth height.

Table 1. RMSE(m) of ATE for different methods on synthesized datasets.

Sequence	VIO	HVIO	RVIO	Proposed Method
1	14.846	4.566	4.288	-
2	49.503	27.479	21.453	16.149

4.1.3. Geo-Localization Performance

We collected another sequence in the simulation environment to evaluate the performance of our proposed method, which fuses the RVIO estimation and the image-registration-based geo-localization results. In this sequence, we fly the UAV along a route that is 3.49 km long and 300 m high. The geo-referenced satellite imagery is obtained from Google Earth Pro, and we used a 3.2 km $\times$ 2.1 km satellite map that was captured in January 2019. Figure 7 illustrates the trajectories obtained from different methods on sequence 2.

An example of successfully matched UAV image and map tile is shown in Figure 8. The results of this experiment, presented in Table 1, show that our proposed method suppresses the drifts of the odometry thanks to image-registration-based geo-localization.

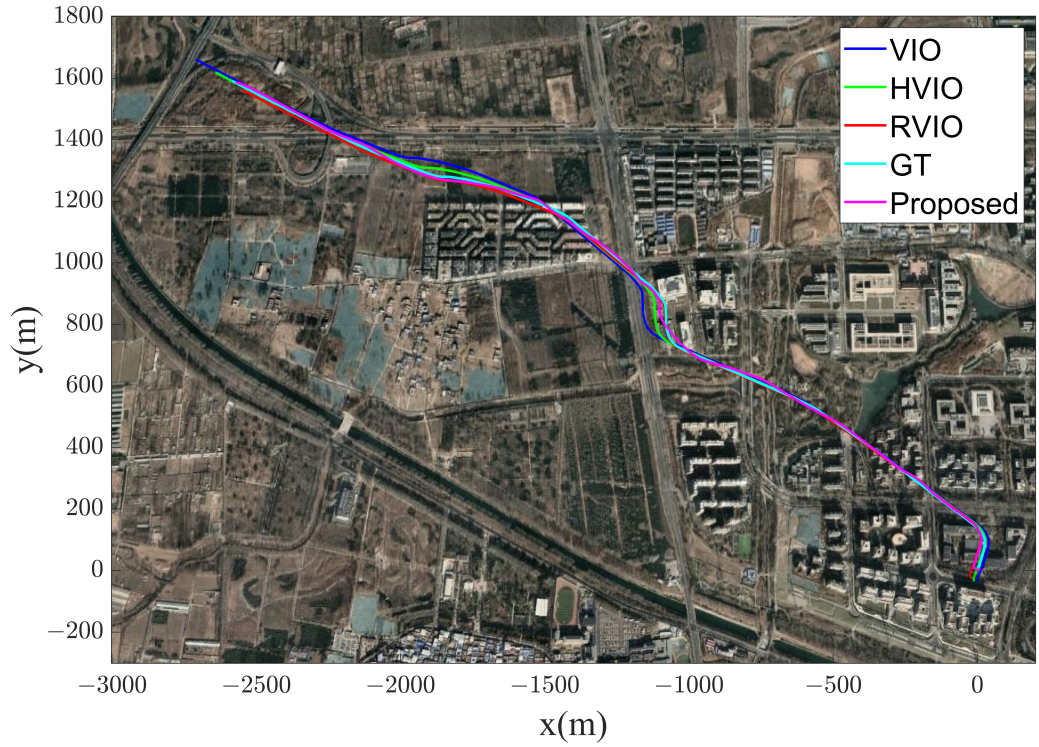


Figure 7. Trajectories estimated by different methods on sequence 2.

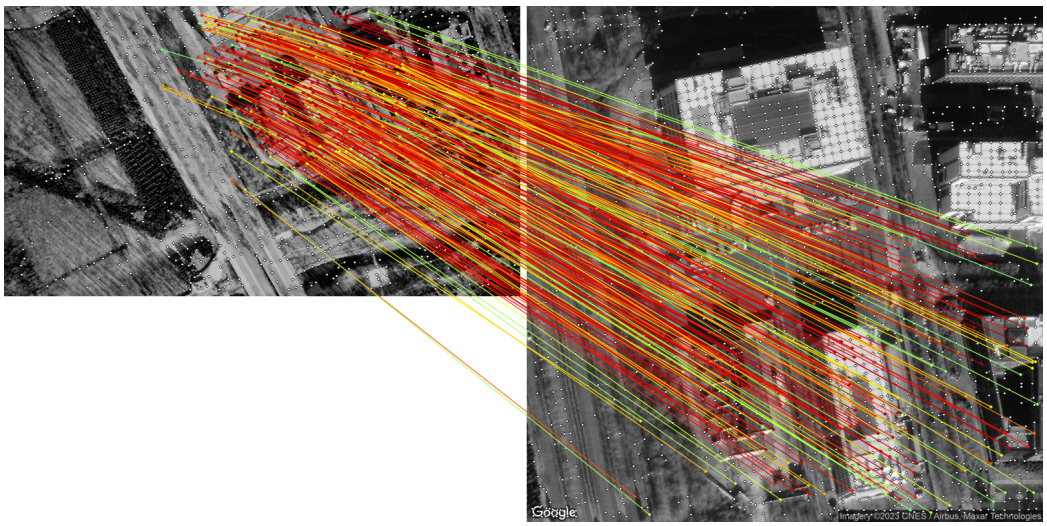


Figure 8. An example of successfully matched UAV image and map tile: The UAV image is shown on the left, and the map tile retrieved from the database is shown on the right. The matched features are connected by colored lines.

4.2. Real-World Dataset

4.2.1. Setup

The real-world experimental setup is displayed in Figure 9. A GNSS receiver is attached to the UAV to allow for ground-truth comparison. The camera, IMU, and LRF are calibrated offline and integrated to place under the UAV. All the measurements are time-synchronized and recorded on an Intel NUC Mini PC through ROS. Based on the setup, we collected two real-world datasets in Tsingtao in March 2023. The trajectories of the UAVs are plotted on the satellite map in Figure 10, encompassing a variety of terrains, such as fields, residential zones, pools, and roads. Additional characteristics of these two datasets are listed in Table 2.

The satellite imagery employed for these experiments is 3.31 km × 5.27 km in size and was obtained from Google Earth Pro in November 2021.

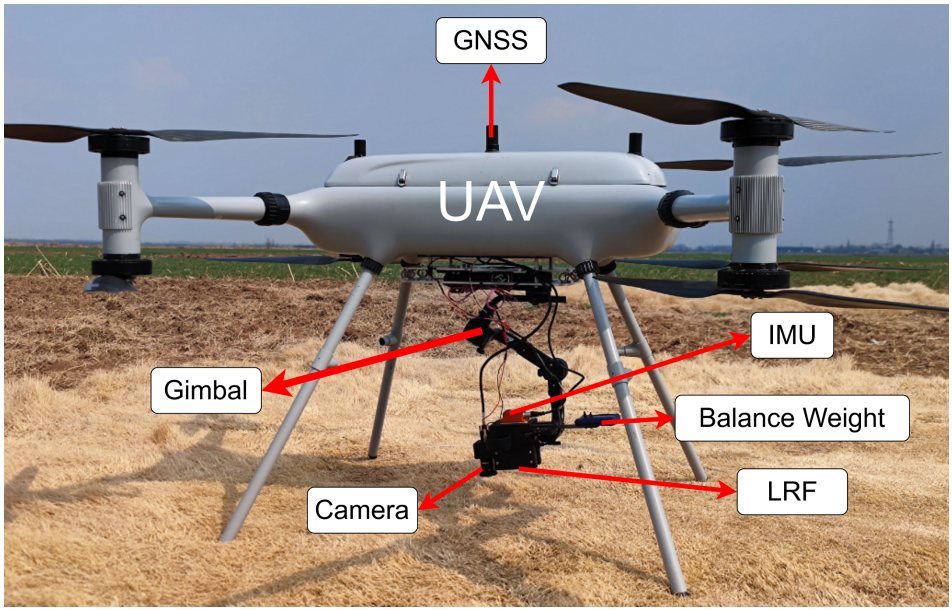


Figure 9. The real-world experimental setup.

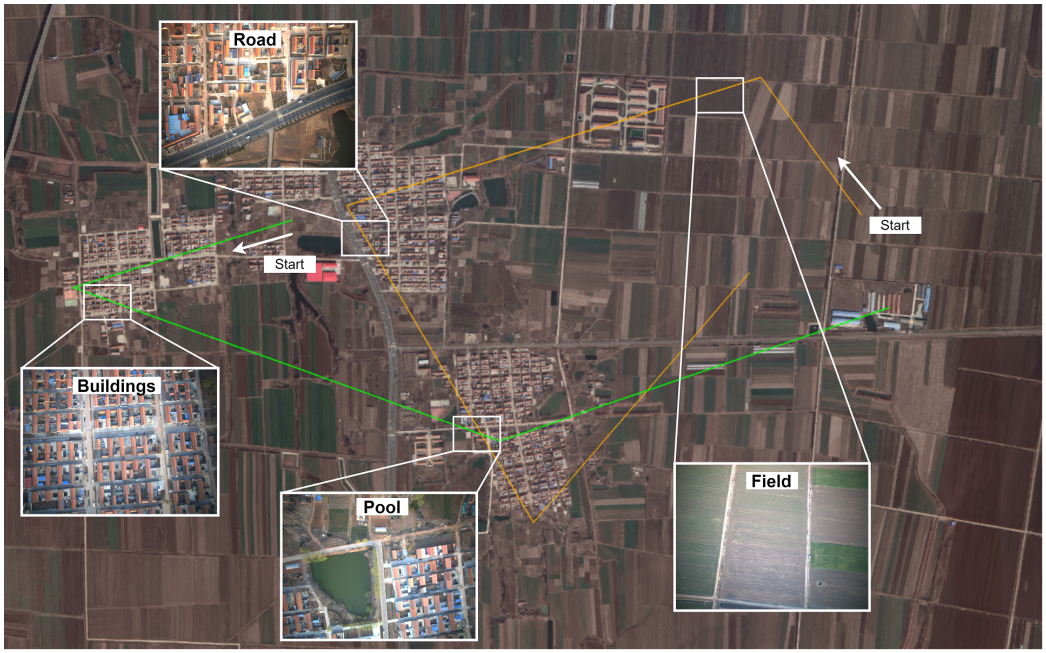
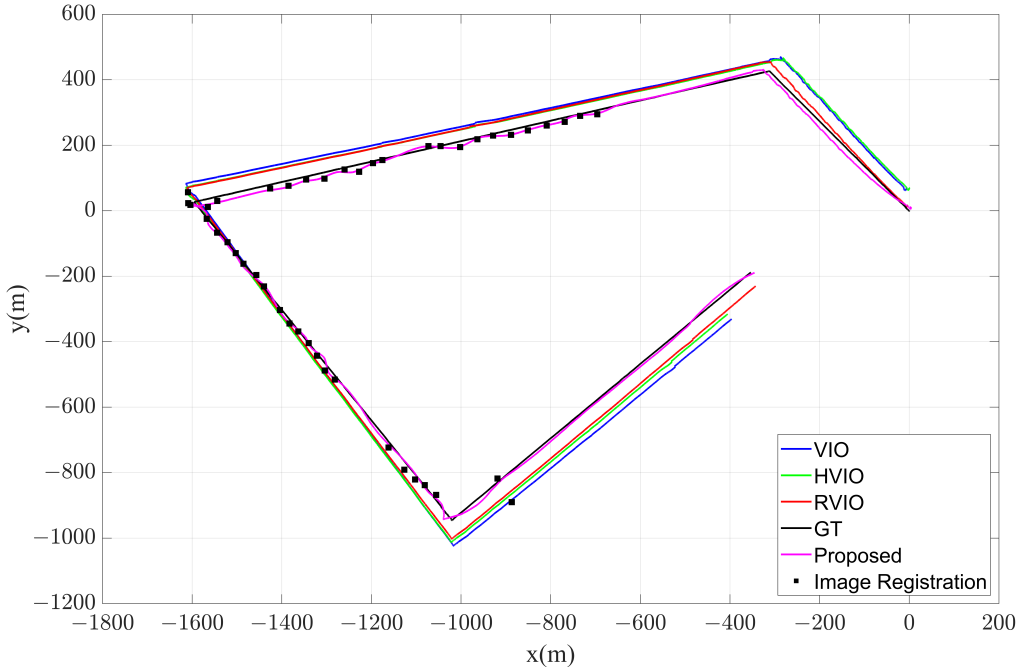


Figure 10. The trajectories of the real-world experiment: The captured frames cover various scenarios such as pools, buildings, fields, and roads.

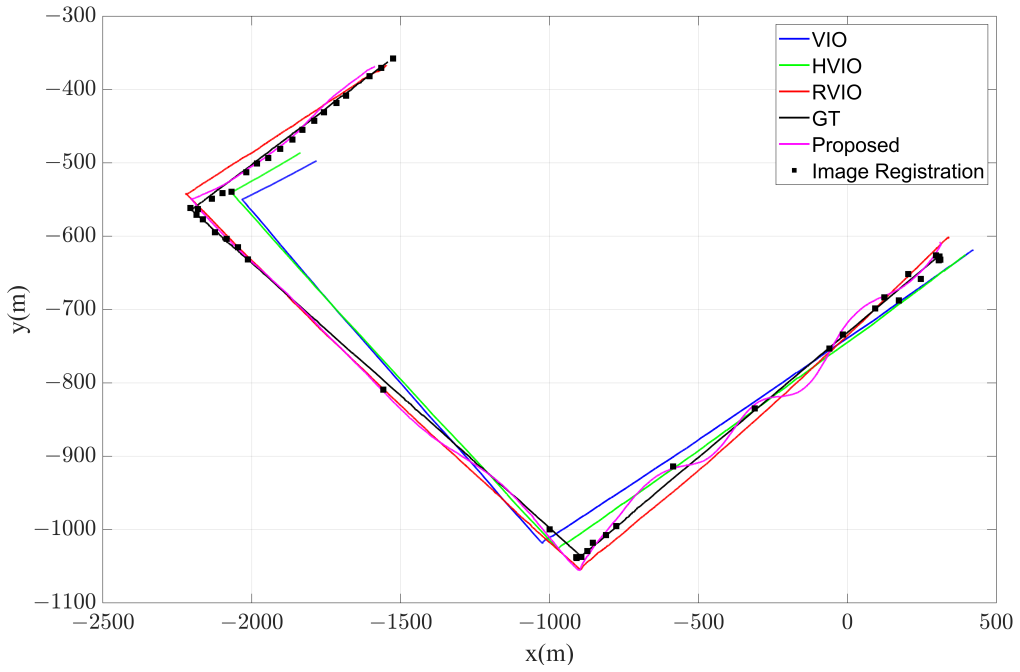
4.2.2. Results and Discussion

To simulate different flight heights, the satellite imagery is categorized into three distinct sizes. Each size consists of a specific number of map tiles: 4701, 4420, and 4150 tiles, corresponding to flight heights of 120 m, 240 m, and 320 m, respectively. Despite the extensive size of the database, we leverage LRF measurements to efficiently access a specific subset of the database. This enables us to retrieve relevant information more efficiently and streamline the retrieval process. Additionally, we conducted tests to evaluate the recall rate for retrieving the best candidate from the dataset. In dataset 1, the recall rate for the images is measured at 73.12%, while in dataset 2, the recall rate is recorded as 70.83%.

Similarly, we compare our proposed method to other methods outlined above. The ground-truth trajectory is provided by GNSS. The trajectories of different methods on the real-world dataset are shown in Figure 11. The black square points indicate the geo-localization results obtained from successful image registration. As shown in Figure 11a, the trajectory of the proposed method more closely matches the ground truth. Figure 11b reveals a scale drift in the results of VIO and HVIO in the initial stage of constant acceleration. However, with the additional constraints for the depths of feature points, the scale drift can be effectively suppressed.



(**a**)



(**b**)

Figure 11. (**a**) Trajectories of different methods on dataset 1. (**b**) Trajectories of different methods on dataset 2.

Table 2. Characteristics of the real-world datasets.

Dataset	Color	Length (m)	Altitude (m)	Duration (s)
1	orange	4426	160	663
2	green	3371	160	477

To ensure efficiency, the image-registration-based localization process is performed every five seconds, considering its time-consuming nature. The average runtime of the image-registration-based localization on the two datasets is presented in Table 3. As a result, the coarse-to-fine image-registration-based localization module does not impact the global optimization thread. The mean errors of the image-registration-based localization on the datasets are reported as 21.012 m and 23.594 m, respectively. Figure 11 illustrates that image-registration-based localization tends to encounter challenges when the UAV is positioned above the field area. This is attributed to the similarity in global descriptors formed by the map tiles within the fields, leading to incorrect matches. For comparative analysis, we reimplemented the state-of-the-art approach [36]. Despite providing the initial GNSS coordinates of the UAV as prior information, this approach still fails to deliver accurate results on the datasets.

Table 3. Average runtime (s) of the image-registration-based localization on real-world datasets.

Dataset	Encoding Time	Retrieval Time
1	0.11	2.96
2	0.13	2.93

The results of geo-localization are fused with the RVIO estimates through global optimization. It is observed that our method exhibits a slight shift toward the image-registration-based localization results. The extent of this offset is influenced by the quality of the global position measurements. To evaluate the performance, we present the Root Mean Square Error (RMSE) for each case in Table 4. Overall, the proposed method provides global position measurements and effectively bounds the drift in odometry, thus enhancing the accuracy and reliability of the system.

Table 4. RMSE (m) of ATE for different methods on real-world datasets.

Dataset	VIO	HVIO	RVIO	[36]	Proposed Method
1	71.944	65.442	38.820	-	12.744
2	101.395	87.541	29.064	-	23.099

5. Conclusions

In this paper, we propose a geo-localization method for UAVs by fusing RVIO estimates and image-registration-based localization results. In particular, the proposed optimization-based RVIO utilizes range measurements to ensure scale consistency. The coarse-to-fine image-registration-based geo-localization module provides an accurate geographical coordinate of the UAV. Then, the global pose graph optimization adaptively adjusts the weights of global position measurements to obtain the global estimation of the UAV. From the experimental results, we can draw these conclusions:

- Based on the planar measurement characteristics of the LRF detection area, it is possible to achieve data association between range measurements and visual feature point depths, thereby achieving accurate and scale-consistent estimation for RVIO.
- The coarse-to-fine image-registration-based geo-localization enables global localization of the UAV and eliminates the drift of odometry methods. Additionally, with the assistance of LRF, the retrieval efficiency can be improved.

- By employing global graph optimization, the results of image-registration-based geo-localization and the outputs from odometry can be effectively fused. Experimental results demonstrate that the proposed method exhibits better localization performance compared with state-of-the-art image-registration-based geo-localization methods.

It should be noted that the proposed method is more suitable for UAV localization in scenarios where the flight distance is in the range of tens to hundreds of meters. In higher-altitude or lower-altitude scenarios, the image-registration-based localization method may not be applicable for UAV localization. Furthermore, the limitation of our method lies in its inability to achieve localization in low-texture environments, such as deserts and oceans. In future work, we will explore more robust features to enable UAV localization in complex environments.

Author Contributions: Conceptualization, Y.H. and Z.M.; methodology, Y.H. and M.H.; software, Y.H. and M.H.; validation, Y.H. and M.H.; formal analysis, Y.H.; investigation, Y.H. and M.H.; data curation, Y.H. and M.H.; writing—original draft preparation, Y.H.; writing—review and editing, Y.H., Y.L., J.L. and Z.M.; visualization, Y.H.; supervision, Z.M.; funding acquisition, Z.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Beijing Natural Science Foundation under Grant JQ20013 and the National Natural Science Foundation of China under Grants 61833009, 62273195, and U19B2029.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

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

References

1. Scherer, J.; Yahyanejad, S.; Hayat, S.; Yanmaz, E.; Andre, T.; Khan, A.; Vukadinovic, V.; Bettstetter, C.; Hellwagner, H.; Rinner, B. An autonomous multi-UAV system for search and rescue. In Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use, Florence, Italy, 18 May 2015; pp. 33–38.
2. Messinger, M.; Silman, M.R. Unmanned aerial vehicles for the assessment and monitoring of environmental contamination: An example from coal ash spills. *Environ. Pollut.* **2016**, *218*, 889–894. [CrossRef] [PubMed]
3. Liu, Y.; Meng, Z.; Zou, Y.; Cao, M. Visual Object Tracking and Servoing Control of a Nano-Scale Quadrotor: System, Algorithms, and Experiments. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 344–360. [CrossRef]
4. Ganesan, R.; Raajini, X.M.; Nayyar, A.; Sanjeevikumar, P.; Hossain, E.; Ertas, A.H. BOLD: Bio-Inspired Optimized Leader Election for Multiple Drones. *Sensors* **2020**, *20*, 3134. [CrossRef]
5. Yayli, U.; Kimet, C.; Duru, A.; Cetir, O.; Torun, U.; Aydogan, A.; Padmanaban, S.; Ertas, A. Design optimization of a fixed wing aircraft. *Int. J. Adv. Aircr. Spacecr. Sci.* **2017**, *4*, 65–80. [CrossRef]
6. Kos, T.; Markezic, I.; Pokrajcic, J. Effects of multipath reception on GPS positioning performance. In Proceedings of the Proceedings ELMAR-2010, Zadar, Croatia, 15–17 September 2010; pp. 399–402.
7. Jiang, Z.; Groves, P.D. NLOS GPS Signal Detection Using a Dual-Polarisation Antenna. *GPS Solut.* **2014**, *18*, 15–26. [CrossRef]
8. Huang, K.W.; Wang, H. Combating the Control Signal Spoofing Attack in UAV Systems. *IEEE Trans. Veh. Technol.* **2018**, *67*, 7769–7773. [CrossRef]
9. Fragoso, A.T.; Lee, C.T.; McCoy, A.S.; Chung, S.J. A seasonally invariant deep transform for visual terrain-relative navigation. *Sci. Robot.* **2021**, *6*, eabf3320. [CrossRef]
10. Shetty, A.; Gao, G.X. UAV Pose Estimation using Cross-view Geolocalization with Satellite Imagery. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 1827–1833.
11. Hao, Y.; Liu, J.; Liu, Y.; Liu, X.; Meng, Z.; Xing, F. Global Visual-Inertial Localization for Autonomous Vehicles with Pre-Built Map. *Sensors* **2023**, *23*, 4510. [CrossRef]
12. Mughal, M.H.; Khokhar, M.J.; Shahzad, M. Assisting UAV Localization Via Deep Contextual Image Matching. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2445–2457. [CrossRef]
13. Couturier, A.; Akhloufi, M.A. A review on absolute visual localization for UAV. *Robot. Auton. Syst.* **2021**, *135*, 103666. [CrossRef]
14. Delmerico, J.; Scaramuzza, D. A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 2502–2509.

15. 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, Roma, Italy, 10–14 April 2007; pp. 3565–3572.
16. Bloesch, M.; Omari, S.; Hutter, M.; Siegwart, R. Robust visual inertial odometry using a direct EKF-based approach. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–3 October 2015; pp. 298–304.
17. 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]
18. Zhang, M.; Han, S.; Wang, S.; Liu, X.; Hu, M.; Zhao, J. Stereo Visual Inertial Mapping Algorithm for Autonomous Mobile Robot. In Proceedings of the 2020 3rd International Conference on Intelligent Robotic and Control Engineering (IRCE), Oxford, UK, 10–12 August 2020; pp. 97–104.
19. Campos, C.; Elvira, R.; Rodríguez, 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. Liu, Y.; Meng, Z. Online Temporal Calibration Based on Modified Projection Model for Visual-Inertial Odometry. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 5197–5207. [CrossRef]
21. He, X.; Gao, W.; Sheng, C.; Zhang, Z.; Pan, S.; Duan, L.; Zhang, H.; Lu, X. LiDAR-Visual-Inertial Odometry Based on Optimized Visual Point-Line Features. *Remote Sens.* **2022**, *14*, 622. [CrossRef]
22. Tyagi, A.; Liang, Y.; Wang, S.; Bai, D. DVIO: Depth-Aided Visual Inertial Odometry for RGBD Sensors. In Proceedings of the 2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Bari, Italy, 4–8 October 2021; pp. 193–201.
23. Giubilato, R.; Chiodini, S.; Pertile, M.; Debei, S. Scale Correct Monocular Visual Odometry Using a LiDAR Altimeter. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 3694–3700.
24. Giubilato, R.; Chiodini, S.; Pertile, M.; Debei, S. MiniVO: Minimalistic Range Enhanced Monocular System for Scale Correct Pose Estimation. *IEEE Sens. J.* **2020**, *20*, 11874–11886. [CrossRef]
25. Delaune, J.; Bayard, D.S.; Brockers, R. Range-Visual-Inertial Odometry: Scale Observability Without Excitation. *IEEE Robot. Autom. Lett.* **2021**, *6*, 2421–2428. [CrossRef]
26. Hu, J.; Hu, J.; Shen, Y.; Lang, X.; Zang, B.; Huang, G.; Mao, Y. 1D-LRF Aided Visual-Inertial Odometry for High-Altitude MAV Flight. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 22–27 May 2022; pp. 5858–5864.
27. Martínez, C.; Mondragón, I.F.; Campoy, P.; Sánchez-López, J.L.; Olivares-Méndez, M.A. A Hierarchical Tracking Strategy for Vision-Based Applications On-Board UAVs. *J. Intell. Robot. Syst.* **2013**, *72*, 517–539. [CrossRef]
28. Yol, A.; Delabarre, B.; Dame, A.; Dartois, J.É.; Marchand, E. Vision-based absolute localization for unmanned aerial vehicles. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 3429–3434.
29. Van Dalen, G.J.; Magree, D.P.; Johnson, E.N. Absolute localization using image alignment and particle filtering. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, San Diego, CA, USA, 4–8 January 2016; p. 0647.
30. Patel, B.; Barfoot, T.D.; Schoellig, A.P. Visual Localization with Google Earth Images for Robust Global Pose Estimation of UAVs. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 6491–6497.
31. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
32. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571.
33. Nassar, A.; Amer, K.; ElHakim, R.; ElHelw, M. A Deep CNN-Based Framework For Enhanced Aerial Imagery Registration with Applications to UAV Geolocalization. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; pp. 1594–159410.
34. Choi, J.; Myung, H. BRM Localization: UAV Localization in GNSS-Denied Environments Based on Matching of Numerical Map and UAV Images. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 4537–4544.
35. Xu, Y.; Wu, S.; Du, C.; Li, J.; Jing, N. UAV Image Geo-Localization by Point-Line-Patch Feature Matching and ICLK Optimization. In Proceedings of the 2022 29th International Conference on Geoinformatics, Beijing, China, 15–18 August 2022; pp. 1–7.
36. Goforth, H.; Lucey, S. GPS-Denied UAV Localization using Pre-existing Satellite Imagery. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 2974–2980.
37. Chen, S.; Wu, X.; Mueller, M.W.; Sreenath, K. Real-Time Geo-Localization Using Satellite Imagery and Topography for Unmanned Aerial Vehicles. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 2275–2281.
38. Kinnari, J.; Verdoja, F.; Kyrki, V. GNSS-denied geolocalization of UAVs by visual matching of onboard camera images with orthophotos. In Proceedings of the 2021 20th International Conference on Advanced Robotics (ICAR), Ljubljana, Slovenia, 6–10 December 2021; pp. 555–562.
39. Kinnari, J.; Verdoja, F.; Kyrki, V. Season-Invariant GNSS-Denied Visual Localization for UAVs. *IEEE Robot. Autom. Lett.* **2022**, *7*, 10232–10239. [CrossRef]

40. Kanade, T.; Amidi, O.; Ke, Q. Real-time and 3D vision for autonomous small and micro air vehicles. In Proceedings of the 2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No. 04CH37601), Nassau, Bahamas, 14–17 December 2004; Volume 2, pp. 1655–1662.
41. Forster, C.; Carlone, L.; Dellaert, F.; Scaramuzza, D. On-Manifold Preintegration for Real-Time Visual–Inertial Odometry. *IEEE Trans. Robot.* **2017**, *33*, 1–21. [CrossRef]
42. Agarwal, S.; Mierle, K.; Team, T.C.S. Ceres Solver. 2022. Available online: https://github.com/ceres-solver/ceres-solver (accessed on 1 May 2023).
43. Arandjelović, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1437–1451. [CrossRef]
44. Jégou, H.; Douze, M.; Schmid, C.; Pérez, P. Aggregating local descriptors into a compact image representation. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 3304–3311.
45. DeTone, D.; Malisiewicz, T.; Rabinovich, A. SuperPoint: Self-Supervised Interest Point Detection and Description. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; pp. 337–33712.
46. Sarlin, P.E.; DeTone, D.; Malisiewicz, T.; Rabinovich, A. SuperGlue: Learning Feature Matching With Graph Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 4937–4946.
47. Qin, T.; Cao, S.; Pan, J.; Shen, S. A General Optimization-based Framework for Global Pose Estimation with Multiple Sensors. *arXiv* **2019**, arXiv:1901.03642.
48. Shah, S.; Dey, D.; Lovett, C.; Kapoor, A. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. In *Field and Service Robotics*; Springer: Cham, Switzerland, 2018; pp. 621–635.
49. Grupp, M. evo: Python Package for the Evaluation of Odometry and SLAM. 2017. Available online: https://github.com/MichaelGrupp/evo (accessed on 1 May 2023).