Measurement 199 (2022) 111559

Contents lists available at ScienceDirect

Measurement

journal homepage: www.elsevier.com/locate/measurement

A pedestrian POS for indoor Mobile Mapping System based on foot-mounted visual–inertial sensors☆

Xiaoji Niu [a,b,*], Yan Wang [a], Jian Kuang [a,**]

[a] GNSS Research Center, Wuhan University, Wuhan, Hubei, CO 430072, PR China
[b] Artificial Intelligence Institute of Wuhan University, PR China

ARTICLE INFO

Keywords:
Indoor mobile mapping system
Pedestrian dead reckoning
Visual navigation system

ABSTRACT

The mobile mapping system (MMS) is an enabling technology for indoor location-based services. The position and orientation system (POS) is one of the cores of the mobile mapping system. This paper proposes a pedestrian POS solution for infrastructure-free environments based on the fusion of foot-mounted IMU and stereo camera. The structure from motion (SFM) constructs the trajectory and environment under the condition with rich visual textures; the stereo camera detects loop-closure for mitigating the accumulated error; the foot-mounted pedestrian dead reckoning (Foot-PDR) provides reliable continuous trajectories using the IMU when the visual-based system degrades or crashes. The feasibility and reliability of the proposed system were verified by an office building test and an underground parking lot test. The proposed system achieved 0.237 m and 0.227 m position errors in these two scenarios, respectively. Moreover, the system can maintain the average position accuracy of 0.3 m when the camera shortly failed in some areas.

1. Introduction

With the development of the Internet of Things(IoT) and Location-based services(LBS), a mobile mapping system(MMS) [1] that can provide geospatially and attribute information in an efficient and automated manner is widely applied in various geographic information applications. And the position and orientation system (POS) that determines the time-variant position and orientation of the mapping platform plays a key role for the MMS. For outdoor scenarios, where the traditional MMS aims to work, the multi-sensory navigation system consists of a global navigation satellite system (GNSS) [2], and inertial navigation system (INS) can provide continuous, reliable, and accurate position and orientation in real-time. Nevertheless, it is challenging to provide reliable and accurate position and orientation in an indoor environment because the GNSS signals are interfered with or even blocked. Thus, the indoor POS solution is vital and urgently needed for indoor MMS.

A reliable and accurate indoor positioning technology remains an open problem when considering the cost and practicality. WLAN [3–5] and Bluetooth [6–8] are widely adopted technique for indoor positioning. The accuracy of those technologies is only 1–3 m in general, which is insufficient for the requirement of indoor MMS. The RFID [9] is also widely adopted to provide higher accuracy positioning, but this technique needs many pre-installed tags. Ultrawideband (UWB) [10,11] can determine position with centimeter level in theory. Unfortunately, UWB cannot work well in complex environments since the signal is easily affected by the non-line-of-sight (NLOS) [12] and multipath errors caused by obstacles. Moreover, wireless position techniques depend on the installment of signal nodes, which requires many human resources, material, and time resources for deployment. Meanwhile, wireless signal-based positioning methods cannot provide accurate orientation. Therefore, from the perspective of the construction cost and system performance, it is unrealistic to provide an indoor POS by using wireless positioning techniques.

Infrastructure-free position techniques are better choices of indoor MMS in practice, which is independent of any pre-installed base stations. Such solutions can easily work in large-scale indoor buildings. The magnetic fingerprinting-based system can provide meter level positioning accuracy using an ambient magnetic field. However, the accuracy is not sufficient and it is suffering from frequent performance degradation caused by the similarity of indoor magnetic distribution and the soft iron effect caused by varying electromagnetic environments [13,14]. The simultaneous localization and mapping (SLAM)

☆ This work was supported in part by the National Natural Science Foundation of China (Grant No. 42174024) P.R.China and the Special Fund of Hubei Luojia Laboratory (220100007) P.R. China.
* Corresponding author at: GNSS Research Center, Wuhan University, Wuhan, Hubei, CO 430072, PR China.
** Corresponding author.
E-mail addresses: xjniu@whu.edu.cn (N. Xiaoji), wystephen@whu.edu.cn (W. Yan), kuang@whu.edu.cn (K. Jian).

https://doi.org/10.1016/j.measurement.2022.111559
Received 13 November 2021; Received in revised form 10 April 2022; Accepted 26 June 2022
Available online 30 June 2022

techniques including camera-based [15–17] or laser-based SLAM [18–20] techniques, can provide position of the mapping platform and a consistent map of environments at the same time. Laser-based SLAM can generate a point cloud map of the environment with high precision by fusing large continuous sets of point clouds from a laser scanner. However, a laser scanner, which is heavy and expensive, is not suitable for some occasions. The camera-based SLAM is an alternative to the laser-based one, which estimates self pose and environments by finding correspondence points between images. However, camera-based SLAM is significantly affected by bad illumination conditions, moving objects, and feature point distributions.

The inertial navigation system (INS), a completely self-contained navigation system without a required external signal, determines relative position and orientation based on measured angular rate and acceleration(i.e., specific force). However, INS has a significant drawback in that the accuracy degrades over time rapidly due to the error of inertial sensors(i.e., gyroscope and accelerometer). Thus, INS can only provide accurate relative position and orientation in the short-term period. The foot-mounted pedestrian dead reckoning(Foot-PDR) [21–23] utilizes the zero-velocity constraint coming from the walk characteristics of the pedestrian to mitigate the drift error of the velocity produced by INS, so as to achieve long-term positioning without external dependency. Nevertheless, the position error is still accumulated slowly. Then, the foot-mounted PDR [24] aided by the control points(i.e.point with known coordinates) is used as an alternative to provide high accuracy position during testing. However, the task of coordinate the surveying of the control points still requires significant human power.

The visual–inertial mapping system, such as the Maplab [25], ORB-SLAM3 [26] and VINS-Fusion [23], fuses the camera and IMU to build a map and reconstruct trajectories. Since it utilized visual and inertial measurements simultaneously, its robustness is significantly improved compared to visual-only systems. However, the visual–inertial mapping system still suffers failure frequently in scenes with insufficient illumination or textureless. Therefore, the mapping workflow should be carefully designed by experts of visual SLAM. This is hard to achieve in real-world applications.

In this study, a Pedestrian Positioning and Orientation System (P-POS) based on a foot-mounted stereo camera and a foot-mounted IMU is proposed for indoor MMS. The proposed system achieves better robustness than the visual–inertial mapping system and the foot-PDR. Compared with the general visual–inertial system, the proposed system is a foot-PDR-centered system. The trajectory of the inherently low-drift foot-PDR is further corrected through the loop-closure detected by the visual system. This high-quality corrected trajectory functions as the initial value of visual–inertial bundle adjustment. Such design relies on the closed-looped foot-PDR and makes the visual result a best-to-have. So, the proposed system can work robustly in harsh environments which contain some sub-areas where the visual-based method is invalid. The proposed design can take full advantage of the foot-mounted condition from the following two aspects: Firstly, the foot-mounted camera suffers serious image blurring problems in general but has guaranteed high-quality image at the moment when the foot attach to the floor. Secondly, the foot-mounted PDR can maintain position and orientation accuracy for a much longer-term compared to a traditional hand-held PDR. Compared to the foot-PDR, the proposed system can automatically detect loop-closure through the camera vision to eliminate accumulated error. The two foot-mounted sensors, i.e., the stereo camera and the IMU, can backup each other in an essential way based on their complementary characteristics.

The paper is organized as follows: the overview of the proposed system is presented in Section 2; the detailed algorithm description of the proposed system are provided in Section 3; the two experiment designed to illustrate the practicality and accuracy of the system is presented in Section 4; finally, Section 5 provides a summarize of this paper.

2. System overview

Fig. 1 shows the block diagram of the proposed visual–inertial mobile mapping system. The system includes Foot-PDR, Visual process, and Back-end.

In the Foot PDR block, the inertial measurements provide roughly poses by adopting a typical ZUPT-aided error-state Kalman filter (ESKF). The zero-velocity update function as a pseudo observation in the foot PDR is applied to correct the IMU mechanism's accumulation drift. Thus, the Foot-mounted IMU can provide long term position when the visual-based system crashed.

In the Visual process block, the keyframe selector chooses the keyframe based on inertial measurements. This operation can be utilized during the data collection stage to reduce the dataset size by ignoring redundant information. Specifically, the keyframe selector select image captured at the middle-time of the stance phase. Obviously, the image captured at other moments has obvious motion blur as shown in Fig. 2. Finally, the selected keyframes are sent to the loop-closure detection and feature tracker module. These two modules find corresponding feature points between adjacent keyframes and loop-closed keyframes, respectively.

The back-end block consists of two stages: the pose graph optimization (PGO) and the visual–inertial bundle adjustment (VIBA). The PGO module uses relative poses from the foot-mounted PDR and loop-closure detection to estimate trajectory. In other words, the PGO module estimates trajectory by minimizing the relative pose residual. Thus, in this stage, errors of the foot-mounted PDR are partially eliminated by adopting loop-closure information. The VIBA module reconstructs trajectory and environment by minimizing pre-integration residual and visual residual. This module is a necessity even it is computationally expensive, because it can improve the accuracy of trajectories and consistency of generated point clouds. Meanwhile, since the results of PGO can provide a good initial position for VIBA module, the calculation time of VIBA will reduce since the iteration times reduce the benefit from the good initial position.

3. Algorithm description

In this section, the algorithm adopted in this paper is described. Section 3.1 brief introduced coordinates used in the proposed algorithm. The foot PDR and the gait cycle of pedestrians are described in Section 3.2. The visual process block is described in Section 3.3. Sections 3.4 and 3.5 described cost function of pose graph optimization and visual–inertial bundle adjustment as shown in Fig. 1 respectively.

3.1. Coordinate definition

There are five coordinate frames, as illustrated in Fig. 3 that have been defined in this paper. The world frame is denoted by $(\cdot)^w$, which is an earth-fixed frame. In this paper, all control points are represented in the world frame. The navigation frame is a gravity-aligned frame denoted by $(\cdot)^n$. It is aligned with the IMU center at the initial moment. The left and right camera frame, as well as the IMU frame while taking the ith image, are denoted by $(\cdot)^{c_i^L}$, $(\cdot)^{c_i^R}$ and $(\cdot)^{b_i}$ respectively.

As shown in Fig. 3, the 3D location of map point k in navigation frame and the ith left camera frame are denoted by $\boldsymbol{p}_k^n$ and $\boldsymbol{p}_k^{b_i}$, and the relation of which are defined by

$$\boldsymbol{p}_k^{b_i} = \boldsymbol{R}_{nb_i}^T(\boldsymbol{p}_k^n - \boldsymbol{t}_{nb_i}) \tag{1}$$

where the $\boldsymbol{R}_{nb_i} \in SO(3)$ and $\boldsymbol{t}_{nb_i} \in \mathbb{R}^3$ represents the rotation from the i-the IMU frame to navigation frame and the coordinate of the origin point of the ith IMU frame in the navigation frame respectively.

The observation of feature points has been represented on a generalized image plane directly to simplify the description. For instance, the coordinates of feature point k observed by the left camera noted

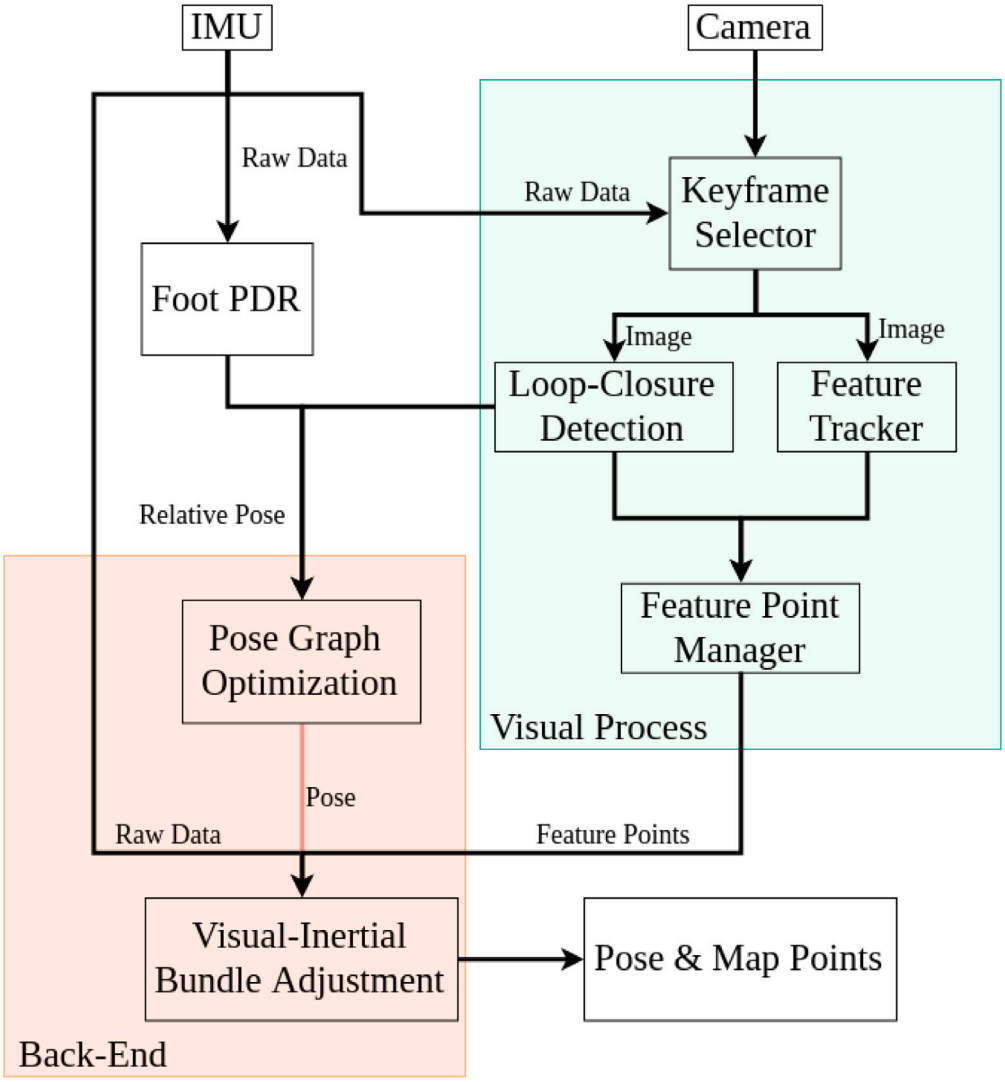

Fig. 1. The block diagram of the proposed visual–inertial mobile mapping system.

as $\hat{\boldsymbol{p}}_k^{c_i^L} = [\hat{x}_k^{c_i^L}, \hat{y}_k^{c_i^L}, 1]^T$. It could be calculated from $\boldsymbol{p}_k^{c_i^L} = [x_k^{c_i^L}, y_k^{c_i^L}, z_k^{c_i^L}]$ through following equation:

$$\hat{\boldsymbol{p}}_k^{c_i^L} = \boldsymbol{\Pi}(\boldsymbol{p}_k^{c_i^L}) \tag{2}$$

where $\boldsymbol{\Pi}(\cdot)$ represent the projection from three-dimensional to image plane as illustrated in Fig. 3.

3.2. Foot PDR

The gait cycle of pedestrians is regular and can provide rigid constraints to the velocity of the feet. As shown in Fig. 2, the gait cycle of pedestrians can be divided into two-phase here. According to the right foot's motion, the gait cycle consists of the stance fundamental system. In the swing phase, the foot swing and moves. In the stance phase, the foot is attached to the floor and nearly does not move. The velocity of the foot in the stance phase is nearly zero [27]. This information can be detected based on IMU measurements and provide observation of velocity for correcting system state. Moreover, the middle-time of the stance phase, which is marked in Fig. 2 is suitable to capture the image since the angular velocity and velocity are nearly zero which helps to reduce the motion blur. The foot PDR estimate trajectory use measurements of foot-mounted IMU.

In foot PDR algorithm, the IMU data is processed through INS mechanization to compute the pose and velocity. In order to adopting the zero-velocity state to eliminate accumulated drift, the error-state Kalman filter (ESKF) will be utilized in this paper. The system state is a 15-dimensional vector defined as following: navigation system

$$\boldsymbol{X}_t = [\boldsymbol{t}_{nb_t}^T, \boldsymbol{R}_{nb_t}^T, \boldsymbol{v}_{nb_t}^T, \boldsymbol{b}_a^T, \boldsymbol{b}_g^T]^T \tag{3}$$

Here, $\boldsymbol{R}_{nb_0}$ and $\boldsymbol{t}_{nb_0}$ same to the definition given before, $\boldsymbol{v}_{nb_t} \in \mathbb{R}^3$ represent velocity of IMU in the navigation frame. The $\boldsymbol{b}_a \in \mathbb{R}^3$ and $\boldsymbol{b}_g \in \mathbb{R}^3$ are bias of accelerometer and gyroscope measurement respectively.

In foot-mounted IMU-based navigation, the zero-velocity state during the stance phase can be adopted to suppressing the accumulation drift of INS mechanization. Generally, the generalized likelihood ratio test can detect the zero-velocity state (GLRT) [27] using accelerometers and gyroscopes without additional equipment required. When the zero-velocity testing passed, a pseudo observation that the IMU velocity equal to zero could be employed in the error-state Kalman filter (ESKF).

Moreover, to eliminate the drift on altitude, which is significantly affect the accuracy of foot-mounted INS, the horizontal motion detection method and altitude correction technique are employed [21]. The horizontal motion detection is based on the change of vertical position. When the vertical position change in a step is lower than a threshold, this step is recognized as horizontal motion mode. If a horizontal motion is detected, the altitude correction uses the altitude of the previous keyframe to constraint the altitude of the current keyframe.

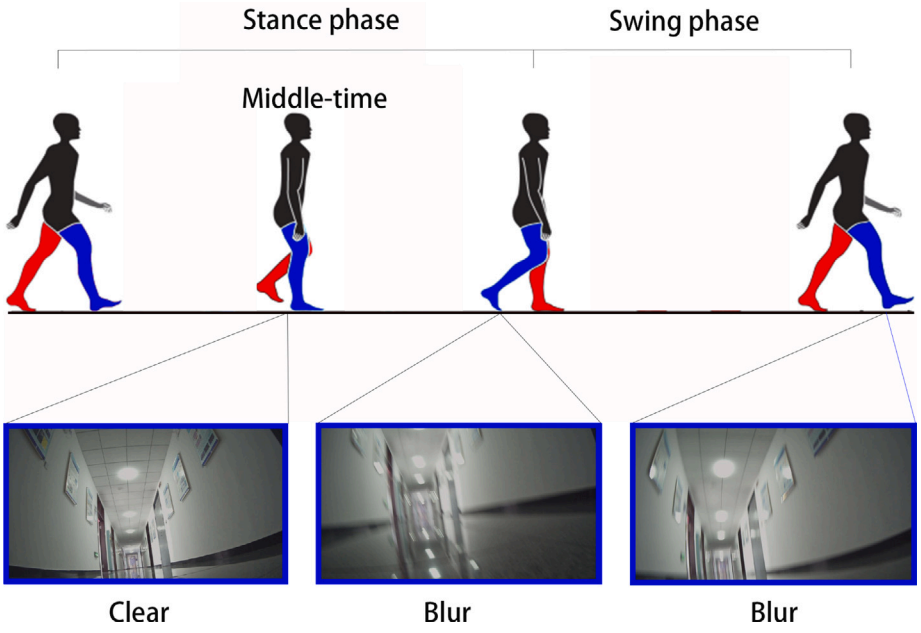

Fig. 2. One gait cycle of the right foot. The left foot and right foot are marked as red and blue, respectively. The moment is denoted as Middle-time with the most less angle velocity in the stance phase and adapted to capture an image. And image captured at each moment shown in gait cycle are shown. The camera mounted at the right foot (blue).

In this paper, the ZUPT-aided ESKF is employed to provide the initial value and the constraint between the adjacent keyframe pair for pose graph optimization in the back-end block. Furthermore, when the vision system degraded, the Foot PDR can estimate trajectory alone over a lengthy period. This function significantly improves the robustness of the proposed system.

3.3. Visual processing

The images collected by the camera are selected based on inertial measurements first. This process is denoted as keyframe selector in Fig. 1. In more detail, during each sequence stance phase, the image with minimum angle velocity will be marked as the keyframe of this period. The selector will select the image captured at middle-time, which is illustrated in Fig. 2. This method of selecting a keyframe is different from a traditional visual-based system. The traditional system selects a keyframe based on parallax and can automatically select more keyframes during a turn to maintain accuracy. However, due to the special keyframe selection mechanism, the proposed system should take shorter steps around corners to maintain accuracy.

Thus, the following modules will process only the keyframe image in loop-closure-detection and feature tracking. Then, the feature points are detected, and descriptors of each feature point are calculated [28]. In the feature tracker block, corresponding points are associated based on the descriptors and two-frame geometry constraints [29]. The loop closure detection module is a two-stage validation. First, the similarity score [30] between each keyframe is calculated. The matching frames should have a high similarity score. Then, the pair of keyframes with high similarity are examined by the geometry constraint of points in those frames. After finding the matching keyframes, the relative pose is calculated and sent to the PGO module. Meanwhile, matching feature points are sent to the feature point manager.

3.4. Loop closured aided pose graph optimization

Since the relative pose function as a measurement in the section, both the uncertainty and the observed value should be calculated. The relative pose could be easily obtained through the equation, which could be written as follows:

$$\begin{cases} \boldsymbol{R}_{b_i b_{i+1}} &= \boldsymbol{R}_{nb_i}^T \boldsymbol{R}_{nb_{i+1}} \\ \boldsymbol{t}_{b_i b_{i+1}} &= \boldsymbol{R}_{nb_i}^T (\boldsymbol{t}_{nb_{i+1}} - \boldsymbol{t}_{nb_i}) \end{cases} \tag{4}$$

where, $\boldsymbol{R}_{nb_i}, \boldsymbol{t}_{nb_i}$ represent pose of ith keyframe represented in the navigation frame. Meanwhile, the covariance matrix of the relative pose $\{\boldsymbol{R}_{b_i b_{i+1}}, \boldsymbol{t}_{b_i b_{i+1}}\}$, which calculated from the pose of IMU at ith and $i+1$th keyframes, are needed for ensure the uncertainty of the relative pose constraint. However, in the classical Kalman filter, the covariance matrix of ith and $i+1$th keyframes are not independent of a probability point of view.

Aim to solve this problem, a technique introduced in the stochastic cloning [31] can be utilized. Through adopting stochastic cloning method, the extended covariance matrix of ith and $i+1$th keyframes could be written as:

$$\boldsymbol{P}_{t+1}^{E} = \begin{bmatrix} \boldsymbol{\Sigma}_{\delta X_{i+1} \delta \boldsymbol{X}_{i+1}} & \boldsymbol{\Sigma}_{\delta \boldsymbol{X}_{i+1} \delta \boldsymbol{X}_i} \\ \boldsymbol{\Sigma}_{\delta \boldsymbol{X}_{i+1}^T \delta \boldsymbol{X}_i} & \boldsymbol{\Sigma}_{\delta \boldsymbol{X}_i \delta \boldsymbol{X}_i} \end{bmatrix}. \tag{5}$$

where, $\Sigma_{\delta \boldsymbol{X}_{i+1} \delta \boldsymbol{X}_{i+1}}$ and $\Sigma_{\delta \boldsymbol{X}_i \delta \boldsymbol{X}_i}$ are represent covariance matrix of the system error state at current and previous moment. And, $\boldsymbol{\Sigma}_{\delta \boldsymbol{X}_{i+1} \delta \boldsymbol{X}_i}$ represent the covariance between current and previous moment, which could be calculated through following equation:

$$\boldsymbol{\Sigma}_{\delta \boldsymbol{X}_{i+1} \delta \boldsymbol{X}_i} = \left(\prod_{t \in \{i, i+1\}} \boldsymbol{\Phi}_t \right) \boldsymbol{\Sigma}_{\delta \boldsymbol{X}_i \delta \boldsymbol{X}_i} \tag{6}$$

Here, $t \in \{i, i+1\}$ meant all IMU measurements in the period from ith to $i+1$th keyframe, $\boldsymbol{\Phi}_t$ represents the state propagation matrix at t moment.

In order to construct the loop-closure constraint, the relative pose between matching keyframes needs to be calculated. Firstly, the corresponding points in the paired keyframe could be matched through feature point descriptors. Then, the relative pose could be easily estimated through a two-frame stereo bundle adjustment. The number of inlier feature points in this bundle adjustment should be larger than a threshold. Perhaps enough feature points are reconstructed, the relative pose of these two frames is adopted in the next stage.

Two components include the relative pose constraint and the altitude constraint, contained in the optimization problem.

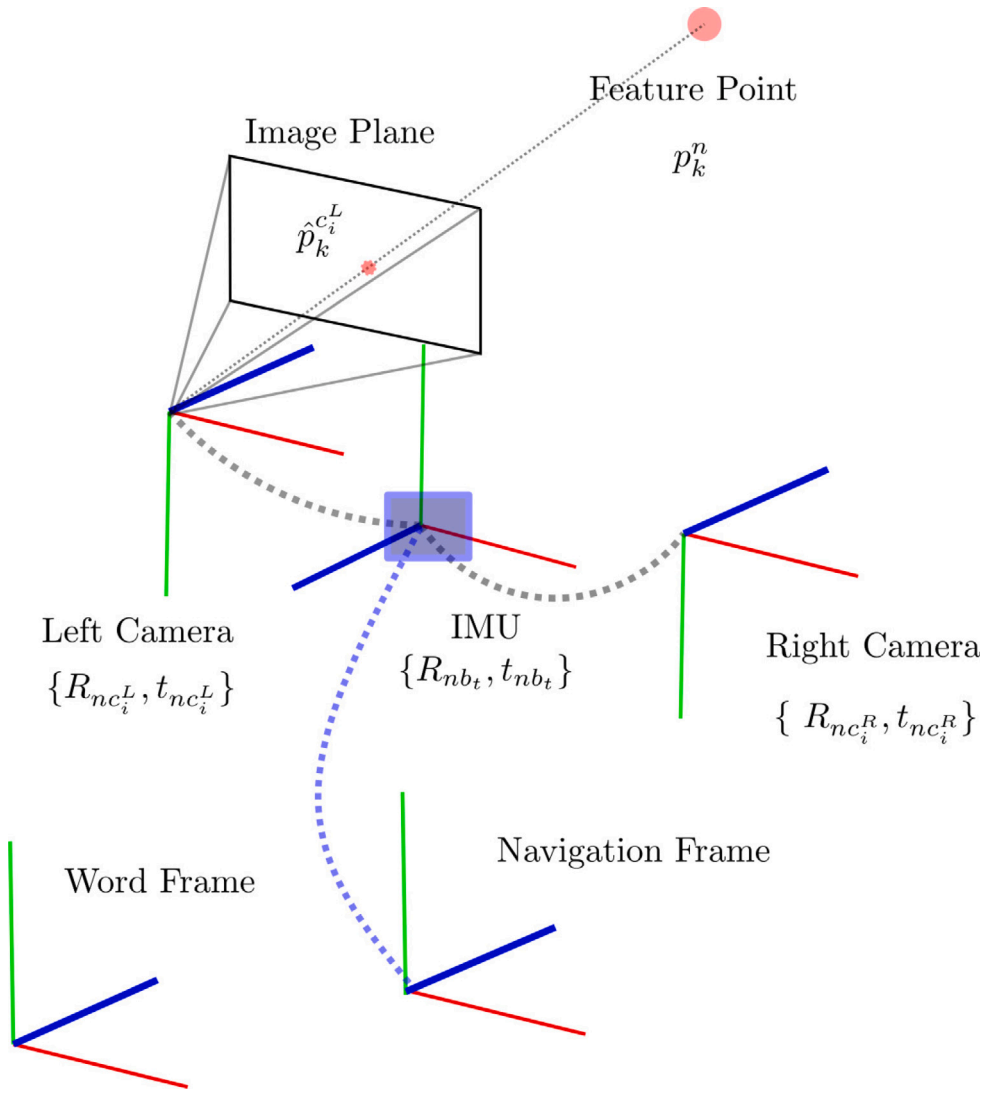

Fig. 3. Coordination transformation between camera, IMU, the navigation frame, and the world frame. Red, green, and blue lines in each coordinate frame represent the x,y, and z-axis, respectively.

The cost function of relative pose constraint used to modeling the information from the INS and the loop closure detection. It can be written as:

$$e_{rel}(\boldsymbol{R}_{nb_i}, \boldsymbol{t}_{nb_i}, \boldsymbol{R}_{nb_j}, \boldsymbol{t}_{nb_j}) = \begin{bmatrix} Log(\boldsymbol{R}_{b_i b_j}^T \boldsymbol{R}_{nb_i}^T \boldsymbol{R}_{nb_j}) \\ \boldsymbol{t}_{b_i b_j} - \boldsymbol{R}_{nb_i}^T (\boldsymbol{t}_{nbj} - \boldsymbol{t}_{nbi}) \end{bmatrix} \tag{7}$$

where, $Log(\cdot)$ represent the converting from $SO(3)$ to $\mathfrak{so}(3)$; $\boldsymbol{R}_{b_i b_j}$ and $\boldsymbol{t}_{b_i b_j}$ are the relative rotation and relative position between ith and jth keyframe respectively.

The information obtained from the horizontal detection is formulated as the altitude constraint, which meant the altitude of the adjacent keyframe is similar. So, its cost function can be defined as following:

$$e_{alt}(\boldsymbol{t}_{nb_i}, \boldsymbol{t}_{nb_j}) = \boldsymbol{t}_{nb_i}[2] - \boldsymbol{t}_{nb_j}[2] \tag{8}$$

here, $\boldsymbol{t}_{nb_i}[2]$ and $\boldsymbol{t}_{nb_j}[2]$ represented z components of $\boldsymbol{t}_{nb_i}$ and $\boldsymbol{t}_{nb_j}$ respectively.

To summarize, the target function of global graph optimization, which is illustration in Fig. 4, can be written in following form:

$$\{\boldsymbol{R}_{nb_i}, \boldsymbol{t}_{nb_i}\}_{\forall i} = \underset{\{\boldsymbol{R}_{nb_i}, \boldsymbol{t}_{nb_i}\}_{\forall i}}{\arg\min} \left\{\boldsymbol{E}_{ins} + \boldsymbol{E}_{loop} + \boldsymbol{E}_{alt}\right\} \tag{9}$$

where $\boldsymbol{E}_{ins}$ and $\boldsymbol{E}_{loop}$ represent the relative pose constraint, which defined in (7), based on INS output and loop closure respectively, $\boldsymbol{E}_{alt}$ represent the altitude constraint, which defined in (8), between adjacent keyframe.

3.5. Visual–inertial bundle adjustment

This section discussed the detail of the visual–inertial bundle adjustment(visual–inertial BA) module, which gives the final result in the proposed algorithm. The initial poses of keyframes are given by the method expressed in Section 3.4. The initial position of feature points represented in the navigation frame should be estimated through initial poses. In practice, feature points observed less than three times or without enough parallax will be ignored. Then, the position of feature points estimates by minimizing re-projection residual based on given camera poses.

The XYZ parameterization of feature points is utilized in visual–inertial BA. The cost function of re-projection error can be written as:

$$\begin{aligned} &e_{proj}(\boldsymbol{p}_k^n, \boldsymbol{R}_{nb_i}, \boldsymbol{t}_{nb_i}) \\ &\quad = \hat{\boldsymbol{p}}_k^{c_i^L} - \boldsymbol{\Pi}(\boldsymbol{R}_{li} \boldsymbol{R}_{nb_i}^T (\boldsymbol{p}_k^n - \boldsymbol{t}_{nb_i}) + \boldsymbol{t}_{li}) \end{aligned} \tag{10}$$

Here, $\boldsymbol{\Pi}(\cdot)$ is the function project point in the camera frame to the image plane, which is provided at (2). The $\boldsymbol{R}_{li}$ and $\boldsymbol{t}_{li}$ represent extrinsic parameters between the left camera and the IMU, which should be

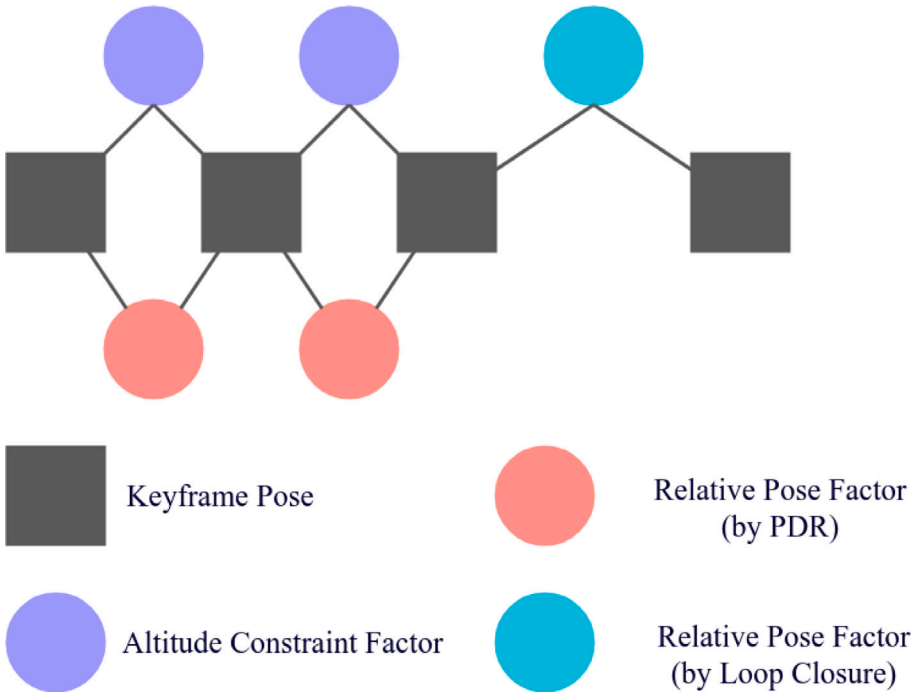

Fig. 4. Illustration of the pose graph optimization definition.

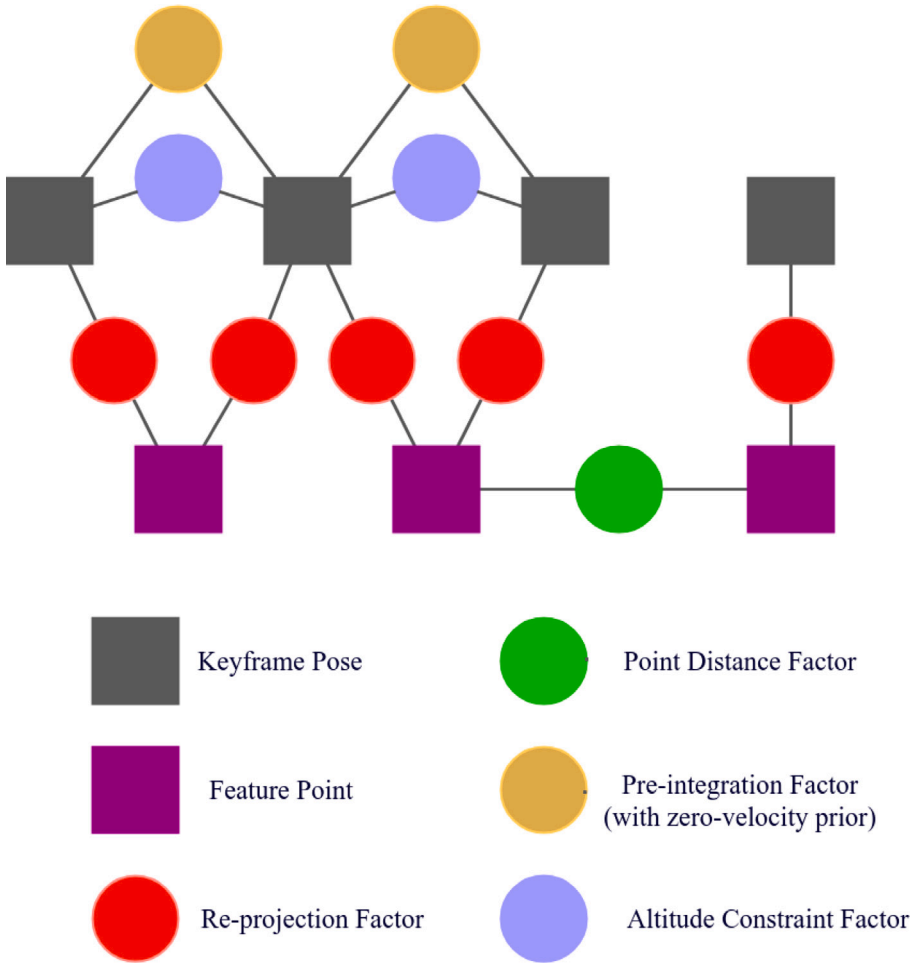

Fig. 5. Illustration of visual–inertial bundle adjustment.

calibrated before experimentation. Moreover, the $\hat{\boldsymbol{p}}_k^{c_i^L}$ represent the position of kth feature point at ith keyframe in left camera.

The pre-integration technique is utilized to modeling the IMU measurements. Furthermore, the zero-velocity observation is added to the cost function to limit the drift of velocity. Thus, the cost function using the IMU measurement can be written as:

$$e_{imu}\left(\begin{bmatrix}\boldsymbol{R}_{nb_i}\\ \boldsymbol{t}_{nb_i}\\ \boldsymbol{v}_{nb_i}\end{bmatrix},\begin{bmatrix}\boldsymbol{R}_{nb_{i+1}}\\ \boldsymbol{t}_{nb_{i+1}}\\ \boldsymbol{v}_{nb_{i+1}}\end{bmatrix}\right)=\begin{bmatrix}e_{pre-integration}\\ \boldsymbol{v}_{zero}-\boldsymbol{v}_{nb_{i+1}}\end{bmatrix} \tag{11}$$

here, $e_{pre-integration}$ denote the cost function of pre-integration on manifold [32], $\boldsymbol{v}_{zero}=[0,0,0]^T$ meant the zero velocity vector.

Another constraint utilized in this stage is the distance constraint between corresponding points. Here, those matching are established through loop-closure. Compare to the way directly represent corresponding points tracked by loop closure as the same point, add a distance constraint between those points can reduce the effect of wrong matching. Especially, the wrong matching is hard to avoid in practice. If ith and jth feature points are the same points observed twice from paired keyframes generated by loop detection, the distance between them should be lower than a certain threshold. This cost function can be written as:

$$e_{dist}(\boldsymbol{p}_i^n,\boldsymbol{p}_j^n)=\|\boldsymbol{p}_i^n-\boldsymbol{p}_j^n\|_2 \tag{12}$$

here, $\|\cdot\|_2$ represents $l2$-norm.

Algorithm 1: Visual–Inertial Bundle Adjustment

Input: $\{\boldsymbol{R}_{nb_i},\boldsymbol{t}_{nb_I}\}_{\forall i},\{\hat{\boldsymbol{p}}_k^{C_i^L},\hat{\boldsymbol{p}}_k^{C_i^R}\},\mathbb{C}$
Output: $\{\boldsymbol{R}_{nb_i},\boldsymbol{t}_{nb_i}\}_{\forall i},\{\boldsymbol{p}_k^n\}_{\forall k}$

1 **for** $\forall k$ **do**
2 Triangulate($\boldsymbol{p}_k^n$);
3 **end**
4 $\{\boldsymbol{R}_{nb_i},\boldsymbol{t}_{nb_i}\}_{\forall i}\leftarrow\arg\min\left\{\boldsymbol{E}_{proj}+\boldsymbol{E}_{imu}+\boldsymbol{E}_{alt}\right\}$;
5 **for** $\forall k$ **do**
6 **if** $AvgPixelError(\boldsymbol{p}_k^n)>\gamma_{pixel}$ **then**
7 RemoveObservation($\boldsymbol{p}_k^n$);
8 **else**
9 RemoveRobustKernel($\boldsymbol{p}_k^n$);
10 **end**
11 **end**
12 **for** $\forall j,k\in\mathbb{C}$ **do**
13 **if** $\|\boldsymbol{p}_j^n-\boldsymbol{p}_k^n\|_2<\gamma_{dist}$ **then**
14 AddDistCost($\boldsymbol{p}_j^n,\boldsymbol{p}_k^n$);
15 **end**
16 **end**
17 Use (13) ;// `Final fine tuning`

The total definition of visual–inertial bundle adjustment is illustrated in Fig. 5, and this problem can be formulated as follows:

$$\{\boldsymbol{R}_{nb_i},\boldsymbol{t}_{nb_i}\}_{\forall i}=\underset{\{\boldsymbol{R}_{nb_i},\boldsymbol{t}_{nb_i}\}}{\arg\min}\left\{\boldsymbol{E}_{proj}+\boldsymbol{E}_{imu}+\boldsymbol{E}_{dist}+\boldsymbol{E}_{alt}\right\} \tag{13}$$

here, $\boldsymbol{E}_{proj}$ represents the re-projection constraint defined in (10), $\boldsymbol{E}_{imu}$ represents the constraint based on IMU measurements defined in (11), and $\boldsymbol{E}_{dist}$ represents the distance constraint for corresponding feature point pairs defined in (12). The $\boldsymbol{E}_{alt}$ same to the definition in (9).

In this method, a multi-stage estimation workflow is adopted to obtain robust and accurate poses estimation. The pseudo-code can be found at Algorithm 1. It is noticing that the optimization in line 4 at Algorithm 1 not employed the constraint of the distance between corresponding feature points detected by loop-closure keyframes. Meanwhile, robust kernel functions are adopted to avoid the effect of abnormal observation.

Fig. 6. Device installment. The stereo camera is tightly mounted to the right foot.

4. Test and result

This section is organized as follows. Section 4.1 describes the device, environment and coordinate frame alignment strategy. Section 4.2 given the result of experiment.

4.1. Test platform and experiment scenario

Fig. 6 shows the test platform. A stereo camera(mynt-1200) is used for data collection, which is marked by a green rectangle in Fig. 6. It consists of two RGB cameras with 1280 × 720 resolution and a MEMS IMU (Bosch BMI088). The stereo camera is mounted on the right foot and connected to a laptop computer through a USB cable. Image and motion measurements are synchronized with each other in the embedded processor of the stereo camera. The data rate of the stereo image and inertial measurement is 20 Hz and 200 Hz, respectively. The camera should be tightly mounted to the foot over the whole experiment, to ensure the foot-mounted PDR's performance does not degrade.

The method is evaluated in two scenarios include an office building and an underground parking lot.

The office building is shown in Fig. 7 which is a real office environment. This scenario aims to validate the performance of the proposed method in an environment with multi-floor. The dimensions of the floor map, as shown in Fig. 7(a), were approximately 86 m × 22 m. The experimental path marked by a group of reference points includes corridors, offices, and stairwells. The reference points are marked by pasting marks on the ground of the experiment route used to evaluate the proposed system's performance. A Leica manual total station measures the coordinates of these points. Thus, its accuracy is better than 5 mm on the plane.

Fig. 8 shows the underground parking lot. It is used to evaluate the positioning performance in another type of environment. There are 17 reference points with known coordinates, and 4 of them function as control points to align the navigation frame and the world frame. These reference points are distributed on a rectangle area of approximately 40 m × 20 m.

It is worth noticing that the output of the visual–inertial bundle adjustment is the trajectory represented in the navigation frame since the proposed system is without adopting external information about the world frame. However, to evaluate the accuracy, the estimated trajectory should be represented in the world frame. In our experiment,

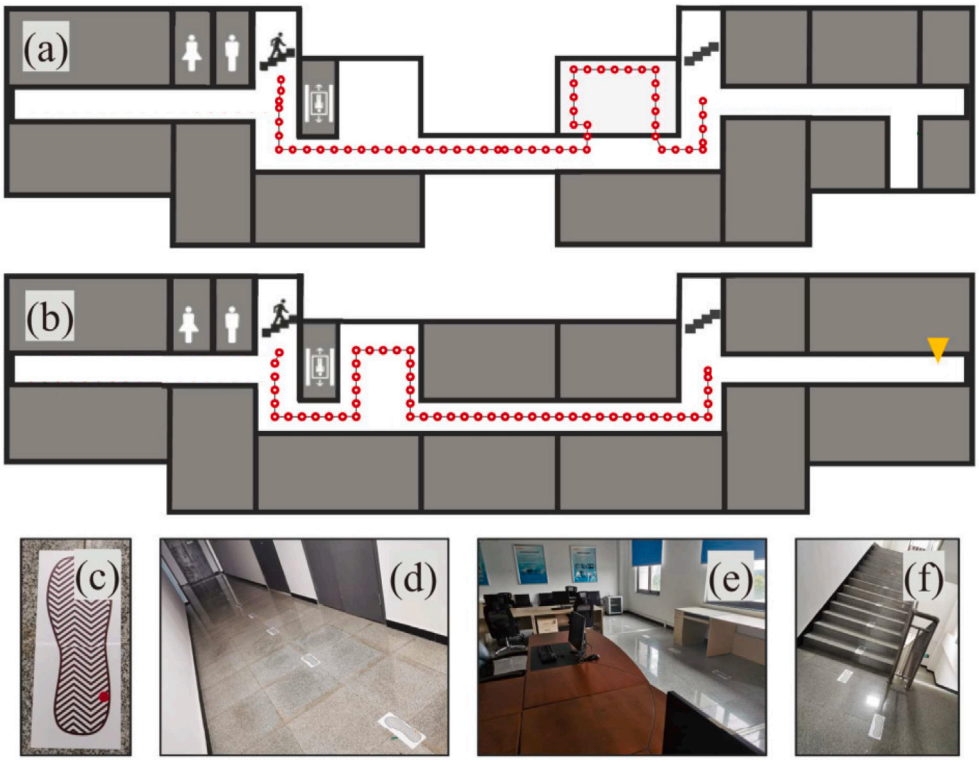

Fig. 7. The office building and reference points. (a) The layout of 2nd floor. (b) The layout of 3rd floor. (c) Marker of the reference points. (d) The corridor scenario. (e) The office scene. (f) The stairwell.

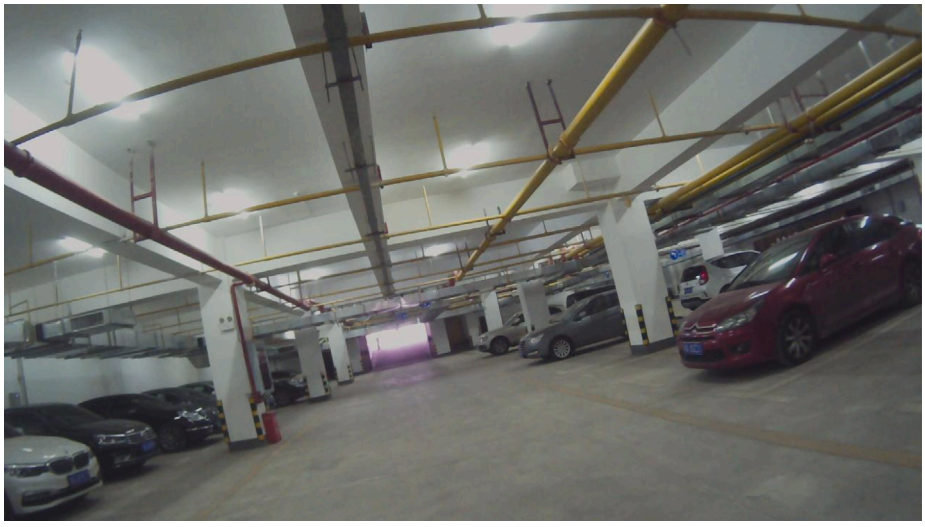

Fig. 8. The underground parking lot environment.

we select four reference points as control points. The transformation between the navigation frame and the world frame, denoted as $\{R_{wn}, t_{wn}\}$, could be obtained through the alignment control points. It can be formulated to an optimization problem shown as follow,

$$\{R_{wn}, t_{wn}\} = \underset{\{R_{wn}, t_{wn}\}}{\arg\min} \sum R_{wn}\hat{P}_i^n + t_{wn} - P_i^w \tag{14}$$

where $\hat{P}_i^n$ represent the coordinate of the control point in the navigation frame, P_i^w represents the coordinate of the control point in the world frame. After the transformation estimated, the position error at ith reference point is defined as:

$$\mathbf{e_i} = \| R_{wn}\hat{P}_i^n + t_{wn} - P_i^w \|_2 \tag{15}$$

where $\| \cdot \|_2$ represent L2 norm.

4.2. Experimental results

The experiment results in the office building and the underground parking lot are provided. And the case called Ignore is presented to verify the performance when the vision system is invalid. Finally, the performance with/without visual–inertial bundle adjustment is compared.

4.2.1. The office building

The estimated trajectory of the proposed algorithm is denoted as Trajectory, and the measurement coordinates of the reference points

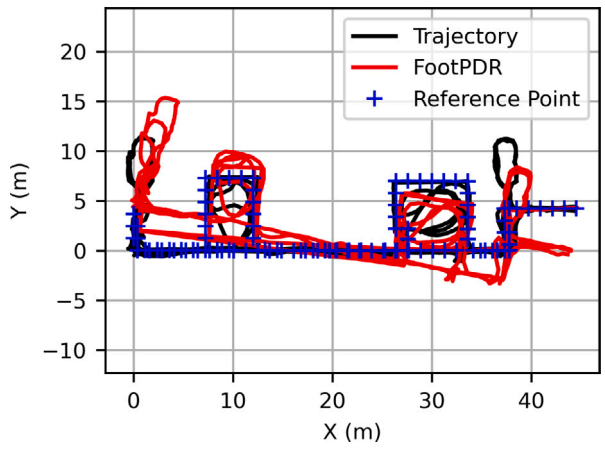

Fig. 9. The trajectories and reference points in 2nd floor and 3rd floor. The black line represents trajectory estimated through the proposed method. The red line represents the Foot PDR only results. (The office building).

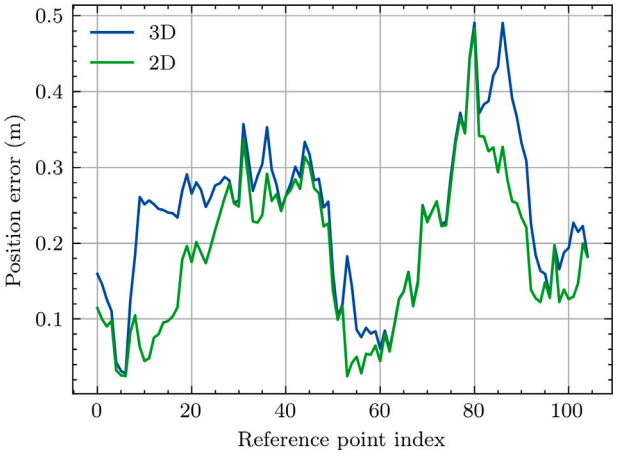

Fig. 10. Position Error(3D) and horizontal position error(2D). The blue line and green line represent 3D and 2D respectively. (The office building).

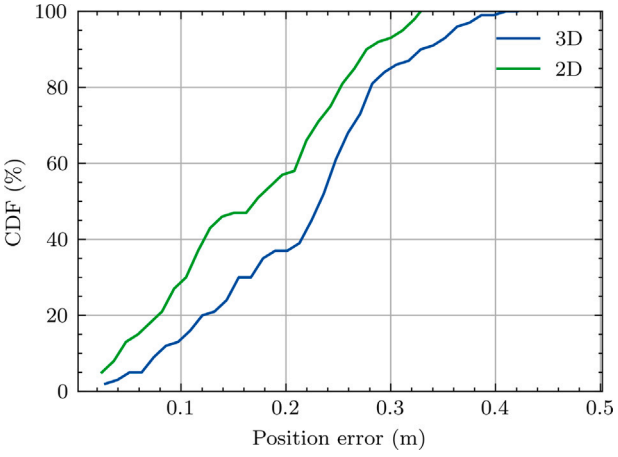

Fig. 11. CDF of the position error and horizontal position error. The green line and blue line represent 3D and 2D, respectively. (The office building).

are denoted as Reference Point in Fig. 9. Since the foot PDR trajectory is represented in the local navigation frame, the first two frames are utilized to align this trajectory and the reference points. The aligned trajectory of foot PDR is denoted as FootPDR in Fig. 9. The position

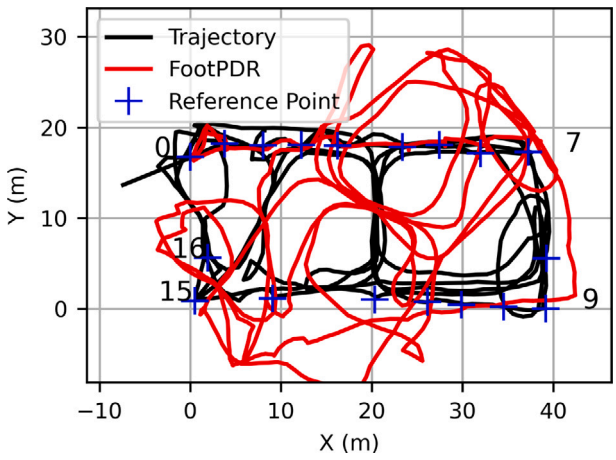

Fig. 12. Trajectories and reference points in the underground parking. The black line represents the estimated trajectory. The red line represents the Foot PDR result. The blue point represents reference points used to calculate position error. Furthermore, reference points 0,7,9,15 function as control points for the alignment coordinate frame. (The underground parking lot).

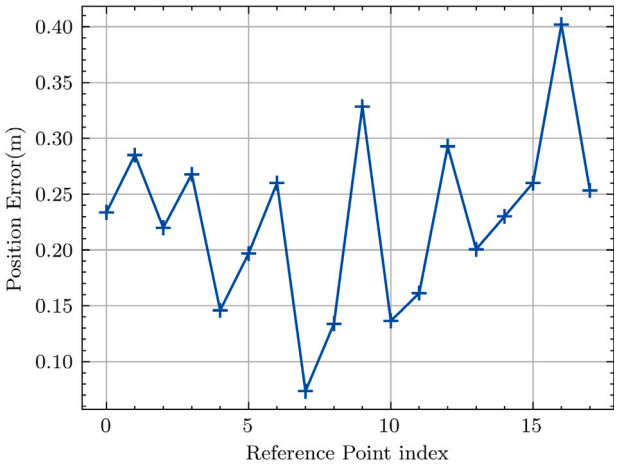

Fig. 13. Position error at each reference point. (The underground parking lot).

error of foot PDR accumulated quickly because of the yaw's drift. However, the yaw's drift cannot be correct without external information. Fig. 10 illustrated the coordinate error at each reference point, where x-axis represents the indexing of the reference point, and y-axis represents the position error. And Fig. 11 shown the cumulative distribution function (CDF) of the position error. The position error of coordinates in three-dimensional is denoted as 3D, and the horizontal position error is denoted as 2D. The mean values of the 3D position error and horizontal position error of the proposed algorithm is 0.237 m and 0.187 m, respectively.

4.2.2. *The underground parking lot*

The trajectory of the experiment taken in the underground parking lot is shown in Fig. 12. Reference Points and the index are illustrated in Fig. 12. The reference points utilized for alignment coordinate frames are at the vertex of the rectangle trajectory(0,7,9,15). Fig. 13 shown the position error at each reference point. Moreover, the average position error of all reference points in this scenario is 0.227 m.

4.2.3. *The ignore case*

In order to test the performance when the visual system is invalid in short periods, the case called Ignore is taken. In this case, we ignored some images to mimic the visual blockages in real-world harsh

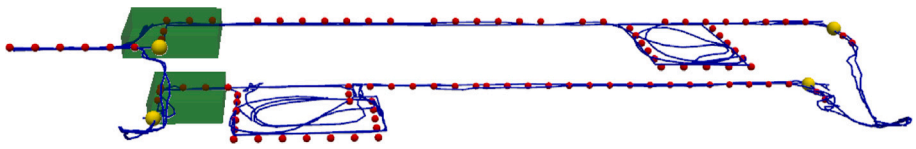

Fig. 14. The trajectory and reference points represent in 3D. The blue line represents the trajectory. The red point represents reference points, and the yellow point represents control points use to align the world frame and the navigation frame. The green box shows the ignored area that mimic the visual failure. (The office building).

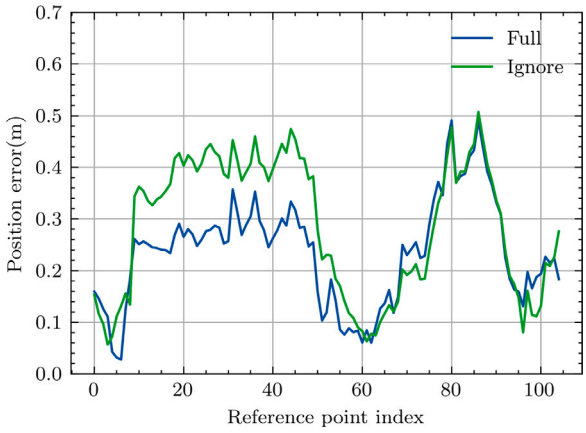

Fig. 15. Position error compare between Full case and Ignore case. (The office building).

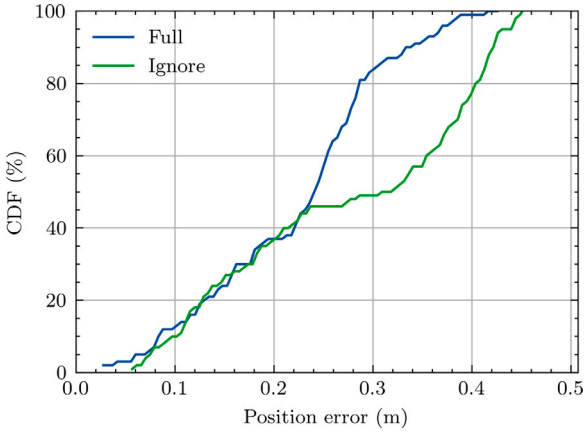

Fig. 16. CDF of position error of the Full case and the Ignore case. The blue and green line represent Full case and Ignore case, respectively. (The office building).

Table 1
Position error of the proposed system in Full case and Ignore case.

		Position error (m)				
		Mean	70%	80%	90%	Max
Full	3D	0.237	0.274	0.286	0.333	0.491
	2D	0.187	0.195	0.222	0.249	0.483
Ignore	3D	0.286	0.386	0.403	0.422	0.507
	2D	0.229	0.286	0.316	0.345	0.463

scenarios. In detail, we select images captured in particular areas to be ignored, rather than randomly ignored some images in the whole dataset. We named these areas as ignore areas, which are marked by green boxes in Fig. 14. These ignore areas are set in the same location at the corners of the corridor on different floors. This situation is representative and easy to occur because the visual system is easy to fails at narrow corners caused by poor texture and image blur.

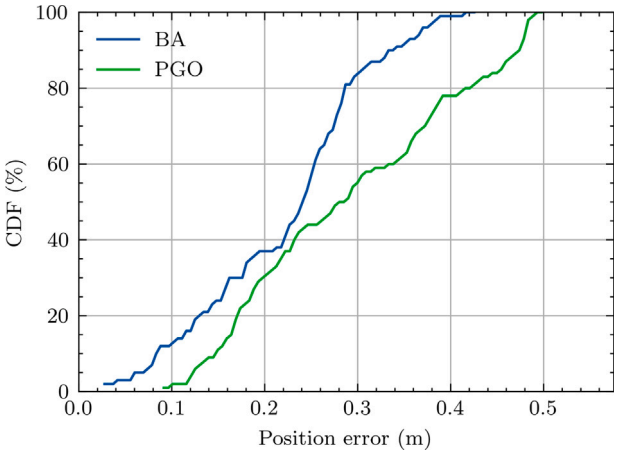

Fig. 17. CDF of the position error with/without visual–inertial BA. The result with/without visual–inertial BA denoted as BA and PGO respectively. (The office building).

Figs. 15 and 16 was given the comparison of position error of the trajectories without/with the Ignore area. Table 1 gave the position error of the full case and the ignore case for clear comparison. The row in the table marked by 3D and 2D represents the position error and the horizontal position error. The column marked by 70% represents that 70% of position errors are less than this value. Moreover, the columns are marked by 80% and 90% with similar definitions. The mean position error and mean horizontal position error of the Ignore case are 0.286 m and 0.229 m. The horizontal position error of the Ignore case is comparable to the results of the Full case. The Ignore case shows that the proposed system can achieve acceptable positioning performance even without visual observation in some places during data collecting for mobile mapping.

4.2.4. *The effect of visual–inertial bundle adjustment*

In order to verify the necessity of VIBA, the compare of result with/without VIBA are provided. The result without BA is denoted as PGO (abbreviation of pose-graph optimization), and the result with BA is denoted as BA. Fig. 17 shown the CDF of BA and PGO. Moreover, the average position error of BA and PGO is 0.237 m and 0.308 m, respectively. The position accuracy of BA is significantly improved. Furthermore, benefit from the adoption of a camera in this system, a sparse point cloud could be generated by the proposed solution. This sparse point cloud represents the set of reconstructed feature points detected in image sequences. Fig. 18(a) and (b) given the left view and the top view of the point cloud of BA, and Fig. 18(c) given the top view of the point cloud of PGO. It can be seen from Fig. 18 that although the point cloud with BA contains a small number of abnormal points, the point cloud can still reflect some outlines of the building, such as ceiling, walls, and corridors. However, the point cloud of PGO is worse, i.e., those line features became scattered. This is because the pose-graph optimization only optimization the relative pose between keyframes. The co-visible feature point by multiple keyframes has not been utilized sufficiently in pose-graph optimization. This result indicated that the visual–inertial BA is necessary for this solution.

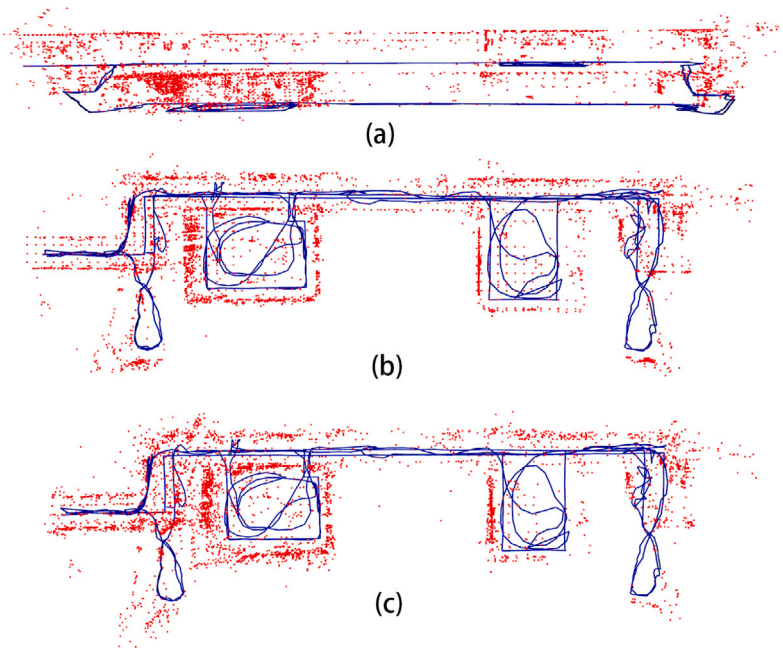

Fig. 18. Point cloud and the estimated trajectory of the proposed algorithm. (a) side view of BA point cloud. (b) top view of BA point cloud. (c) top view of PGO point could. Red points represent feature points reconstructed via visual measurement and blue lines represent estimated trajectory. (The office building).

4.3. Discussion

In this section, we discuss the position and orientation estimation performance of the proposed system from the accuracy and robustness. Firstly, we evaluate the positioning performance of the proposed method in two typical scenarios in Sections 4.2.1 and 4.2.2. The average position error in the office building and the underground parking lot is 0.237 m and 0.227 m, respectively. Secondly, we analyze the robustness of the proposed method at the visual blockage, which is common in harsh real-world scenarios. As mentioned in Section 4.2.3, the proposed method does not show a significant increase in position error even if the vision system is invalid in some sub-area.

Furthermore, we demonstrate the feasibility of the proposed method with experimental results in Section 4.2.4. The visual–inertial bundle adjustment (VIBA) block is computationally expensive, but necessary. The VIBA block not only improve the position accuracy in the navigation frame, but also the relative pose accuracy, which is vital in reconstructing vision maps such as visual feature maps, semantic maps, etc.

5. Conclusion

A foot-mounted visual–inertial indoor POS is proposed in this study. The foot-mounted IMU can provide accurate relative poses when the visual-based SLAM system fails. The camera can detect loop-closures, so as to eliminate error accumulation of the foot PDR. By combining these two systems, the proposed method can provide a globally consistent mapping capability without external dependency.

According to the field experiments described in this paper, this indoor POS solution is practical and reliable for infrastructure-free environments. The proposed system achieves position errors of 0.237 m and 0.227 m in the typical office building and the underground parking lot, respectively. Meanwhile, it can maintain the positioning accuracy of 0.3 m in the mimic visual gaps (i.e., the Ignore case). This result proves that the proposed system could provide acceptable performance even when the vision system crashed for short term. In addition, the necessity of VIBA was verified in the office building case. Using visual–inertial bundle adjustment (VIBA) can reduce the position error from 0.308 m to 0.237 m, and the point cloud quality can be improved effectively.

For future works, we intended to extend the proposed system by adding a hand-held camera that provides more information and achieves higher accuracy by linking to the foot-mounted sensor appropriately. Furthermore, a method to achieve distributed mapping system based on foot-mounted visual–inertial sensors will be considered for wide-area indoor MMS.

CRediT authorship contribution statement

Xiaoji Niu: Conceptualization, Writing – review & editing, Supervision. **Yan Wang:** Methodology, Software, Validation, Writing – original draft. **Jian Kuang:** Conceptualization, Methodology, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Karam, G. Vosselman, M. Peter, S. Hosseinyalamdary, V. Lehtola, Design, calibration, and evaluation of a backpack indoor mobile mapping system, Remote Sens. 11 (8) (2019) 905, http://dx.doi.org/10.3390/rs11080905.

[2] J. Liu, K. Gao, W. Guo, J. Cui, C. Guo, Role, path, and vision of 5G+BDS/GNSS, Satell. Navig. 1 (1) (2020) 23, http://dx.doi.org/10.1186/s43020-020-00024-w.

[3] Z. Yang, C. Wu, Y. Liu, Locating in fingerprint space: wireless indoor localization with little human intervention, in: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking - Mobicom '12, Istanbul, Turkey, 2012, p. 269. http://dx.doi.org/10.1145/2348543.2348578.

[4] K. Wang, et al., Learning to improve WLAN indoor positioning accuracy based on DBSCAN-KRF algorithm from RSS fingerprint data, IEEE Access 7 (2019) 72308–72315, http://dx.doi.org/10.1109/ACCESS.2019.2919329.

[5] M. Zhou, Y. Lin, N. Zhao, Q. Jiang, X. Yang, Z. Tian, Indoor WLAN intelligent target intrusion sensing using ray-aided generative adversarial network, IEEE Trans. Emerg. Top. Comput. Intell. 4 (2020) 61–73, http://dx.doi.org/10.1109/TETCI.2019.2892748.

[6] J. Röbesaat, P. Zhang, M. Abdelaal, O. Theel, An improved BLE indoor localization with Kalman-based fusion: An experimental study, Sensors 17 (5) (2017) 5, http://dx.doi.org/10.3390/s17050951.

[7] B. Zhou, Z. Gu, W. Ma, X. Liu, Integrated BLE and PDR indoor localization for geo-visualization mobile augmented reality, in: 2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV), 2020, pp. 1347–1353, http://dx.doi.org/10.1109/ICARCV50220.2020.9305324.

[8] A. Sato, M. Nakajima, N. Kohtake, Rapid BLE beacon localization with range-only EKF-SLAM using beacon interval constraint, in: 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN). Presented at the 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2019, pp. 1–8, http://dx.doi.org/10.1109/IPIN.2019.8911778.

[9] P. Yang, W. Wu, M. Moniri, C.C. Chibelushi, SLAM algorithm for 2D object trajectory tracking based on RFID passive tags, in: 2008 IEEE International Conference on RFID. Presented at the 2008 IEEE International Conference on RFID, 2008, pp. 165–172, http://dx.doi.org/10.1109/RFID.2008.4519349.

[10] Y. Wang, X. Li, An improved robust EKF algorithm based on sigma points for UWB and foot-mounted IMU fusion positioning, J. Spat. Sci. 66 (2) (2021) 329–350, http://dx.doi.org/10.1080/14498596.2019.1632754.

[11] V. Barral, C.J. Escudero, J.A. García-Naya, R. Maneiro-Catoira, NLOS identification and mitigation using low-cost UWB devices, Sensors 19 (16) (2019) 16, http://dx.doi.org/10.3390/s19163464.

[12] K. Yu, K. Wen, Y. Li, S. Zhang, K. Zhang, A novel NLOS mitigation algorithm for UWB localization in harsh indoor environments, IEEE Trans. Veh. Technol. 68 (1) (2019) 686–699, http://dx.doi.org/10.1109/TVT.2018.2883810.

[13] M. Kwak, C. Hamm, S. Park, T.T. Kwon, Magnetic field based indoor localization system: A crowdsourcing approach, in: 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, Italy, 2019, pp. 1–8, http://dx.doi.org/10.1109/IPIN.2019.8911795.

[14] J. Kuang, X. Niu, X. Chen, Robust pedestrian dead reckoning based on MEMS-IMU for smartphones, Sensors 18 (5) (2018) 5, http://dx.doi.org/10.3390/s18051391.

[15] R. Mur-Artal, J.D. Tardos, ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras, IEEE Trans. Robot. 33 (5) (2017) 1255–1262, http://dx.doi.org/10.1109/TRO.2017.2705103.

[16] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, G. Huang, OpenVINS: A research platform for visual-inertial estimation, in: 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 2020, pp. 4666–4672, http://dx.doi.org/10.1109/ICRA40945.2020.9196524.

[17] A.I. Mourikis, S.I. Roumeliotis, A multi-state constraint kalman filter for vision-aided inertial navigation, in: Proceedings 2007 IEEE International Conference on Robotics and Automation, Rome, Italy, 2007, pp. 3565–3572. http://dx.doi.org/10.1109/ROBOT.2007.364024.

[18] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, D. Rus, LIO-SAM: Tightly-coupled lidar inertial odometry via smoothing and mapping, 2020, arXiv:2007.00258 [cs]. Accessed: Jun. 28, 2021. [Online]. Available: http://arxiv.org/abs/2007.00258.

[19] L. Chang, X. Niu, T. Liu, J. Tang, C. Qian, GNSS/INS/LiDAR-SLAM integrated navigation system based on graph optimization, Remote Sens. 11 (9) (2019) 1009, http://dx.doi.org/10.3390/rs11091009.

[20] T. Shan, B. Englot, LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 4758–4765, http://dx.doi.org/10.1109/IROS.2018.8594299.

[21] A.R. Jiménez, F. Seco, F. Zampella, J.C. Prieto, J. Guevara, PDR with a foot-mounted IMU and ramp detection, Sensors 11 (10) (2011) 9393–9410, http://dx.doi.org/10.3390/s111009393.

[22] W. Liu, Y. Zhang, X. Yang, S. Xing, Pedestrian navigation using inertial sensors and altitude error correction, Sensor Rev. 35 (1) (2015) 68–75, http://dx.doi.org/10.1108/SR-01-2014-612.

[23] T. Qin, P. Li, S. Shen, VINS-mono: A robust and versatile monocular visual-inertial state estimator, IEEE Trans. Robot. 34 (4) (2018) 1004–1020, http://dx.doi.org/10.1109/TRO.2018.2853729.

[24] X. Niu, T. Liu, J. Kuang, Y. Li, A novel position and orientation system for pedestrian indoor mobile mapping system, IEEE Sens. J. (2020) 1, http://dx.doi.org/10.1109/JSEN.2020.3017235.

[25] T. Schneider, et al., Maplab: An open framework for research in visual-inertial mapping and localization, IEEE Robot. Autom. Lett. 3 (3) (2018) 1418–1425, http://dx.doi.org/10.1109/LRA.2018.2800113.

[26] C. Campos, R. Elvira, J.J.G. Rodríguez, J.M.M. Montiel, J.D. Tardós, ORB-SLAM3: An accurate open-source library for visual, visual–Inertial, and multimap SLAM, IEEE Trans. Robot. (2021) 1–17, http://dx.doi.org/10.1109/TRO.2021.3075644.

[27] I. Skog, P. Handel, J.O. Nilsson, J. Rantakokko, Zero-velocity detection—An algorithm evaluation, IEEE Trans. Biomed. Eng. 57 (11) (2010) 2657–2666, http://dx.doi.org/10.1109/TBME.2010.2060723.

[28] D. DeTone, T. Malisiewicz, A. Rabinovich, SuperPoint: Self-supervised interest point detection and description, 2018, arXiv:1712.07629 [cs]. Accessed: Oct. 13, 2020. [Online]. Available: http://arxiv.org/abs/1712.07629.

[29] P.-E. Sarlin, D. DeTone, T. Malisiewicz, A. Rabinovich, SuperGlue: Learning feature matching with graph neural networks, 2020, arXiv:1911.11763 [cs]. Accessed: Oct. 13, 2020. [Online]. Available: http://arxiv.org/abs/1911.11763.

[30] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, J. Sivic, NetVLAD: CNN Architecture for Weakly Supervised Place Recognition, p. 11.

[31] S.I. Roumeliotis, J.W. Burdick, Stochastic cloning: a generalized framework for processing relative state measurements, in: Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292), Vol. 2, Washington, DC, USA, 2002, pp. 1788–1795. http://dx.doi.org/10.1109/ROBOT.2002.1014801.

[32] C. Forster, L. Carlone, F. Dellaert, D. Scaramuzza, On-manifold preintegration for real-time visual-inertial odometry, 2016, http://dx.doi.org/10.1109/TRO.2016.2597321, arXiv:1512.02363 [cs].