

# Tightly-Coupled Monocular Visual-Odometric SLAM Using Wheels and a MEMS Gyroscope

MEIXIANG QUAN<sup>1</sup>, SONGHAO PIAO<sup>1</sup>, MINGLANG TAN<sup>2</sup>, AND SHI-SHENG HUANG<sup>2</sup>.

<sup>1</sup>School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China

<sup>2</sup>Hyperception Inc., Beijing 100083, China

Corresponding authors: Songhao Piao (piaosh@hit.edu.cn) and Shi-Sheng Huang (shishenghuang.net@gmail.com)

This work was supported in part by the National Science Foundation of China under Grant 61375081, and in part by the Special Fund Project of Harbin Science and Technology Innovation Talents Research under Grant RC2013XK010002.

**ABSTRACT** In this paper, we present a novel tightly-coupled probabilistic monocular visual-odometric simultaneous localization and mapping (VOSLAM) algorithm using wheels and a MEMS gyroscope, which can provide accurate, robust, and long-term localization for ground robots. Firstly, we present a novel odometer preintegration theory on manifold, it integrates the wheel encoder measurements and gyroscope measurements to a relative motion constraint that is independent of the linearization point, and it carefully addresses the uncertainty propagation and gyroscope bias correction. Based on the preintegrated odometer measurement model, we also introduce the odometer error term and tightly integrate it into the visual optimization framework. Then in order to bootstrap the VOSLAM system, we propose a simple map initialization method. Finally, we present a complete localization mechanism to maximally exploit both sensing cues, which provides different strategies for motion tracking when (1) both measurements are available, (2) visual measurements are not available, and (3) wheel encoders experience slippage, thereby ensuring the accurate and robust motion tracking. The proposed algorithm is evaluated by performing extensive experiments, and the experimental results demonstrate the superiority of the proposed system.

**INDEX TERMS** Motion estimation, sensor fusion, simultaneous localization and mapping

## I. INTRODUCTION

Simultaneous localization and mapping (SLAM) from on-board sensors is a fundamental and key technology for autonomous mobile robot to safely interact within its workspace. SLAM is a technique that builds a globally consistent representation of the environment (i.e. the map) and estimates the state of robot in the map simultaneously. Because SLAM can be used in many practical applications, such as autonomous driving, indoor service robots, and virtual or augmented reality, it has received considerable attention from Robotics and Computer Vision communities.

In this paper, we study the monocular vision-based localization and mapping algorithm for domestic ground robots moving on a plane, such as cleaning robot, nursing robot, and restaurant robot waiter. When localizing the domestic ground robots, planar-motion constraint is often used to improve the localization accuracy. There are two ways to use planar-motion constraint. One is deterministic planar-motion constraint, in which the constraint is not affected by

noise, i.e., the robot is thought to move on a deterministic plane. The other is stochastic planar-motion constraint, in which the constraint is affected by Gaussian noise, thereby taking into account the out-of-plane motion perturbations. Some visual SLAM (VSLAM) system for ground robots have parameterized the pose as SE(2), i.e., deterministic planar-motion constraint, to only consider the deterministic in-plane motion, e.g. in [1]. However in practical environment, ground robots are moving on an approximately flat surface due to the uneven terrains, some objects on the ground, and vibrations of the moving platform. When the robot motion is often out of the constrained planar model, since visual observation is highly coupled with all 6 DoF of 3D pose, the deterministic planar-motion constraint will lead to decrease in visual localization accuracy. Thereby as in [2], we parameterize the pose as SE(3) and add a stochastic planar-motion constraint on the pose, which can explicitly model the approximately in-plane motion of the robot.

For localizing the robot in indoor environment, monocular

visual-inertial setup is the minimum and most commonly used sensor suite due to the complementary characteristics of both sensors. Monocular visual-inertial SLAM (VISLAM) system [3]–[6] can accurately localize the sensor with general 3D motion. However, ground robots interested in this paper are usually constrained to move along straight lines or circular arcs with constant acceleration. As demonstrated in [2], when the robot moves with these specific motions, the monocular VISLAM has additional unobservable directions, e.g. scale, which leads to significant estimation error. Therefore, monocular visual-inertial sensor suite is not able to provide accurate motion tracking for ground robots.

Most ground robots are equipped with wheel encoders. In most cases, wheel encoders provide the reliable inter-frame traveled distance measurements of each wheel at low frequency. Traditional monocular visual-odometric SLAM (VOSLAM) methods [7] [8] just use distance information from wheel encoders to render the scale factor observable, which is due to the fact that the wheel encoders even provide poor orientation estimation for planar trajectory as demonstrated in [8]. Even though the wheel encoders can provide accurate in-plane rotation, when the robot motion is often out of the plane constraint, the encoders can only provide the accurate traveled distance, not the accurate relative transformation. In contrast, the MEMS gyroscope provides accurate inter-frame rotational information at high frequency. Therefore in this paper, we fuse distance measurements from wheel encoders with angular velocity measurements from gyroscope to provide an accurate inter-frame SE(3) relative transformation for both in-plane motion and out-of-plane motion. In the following, we will call the wheel encoders and MEMS gyroscope as odometer. Then by fusing the odometer measurements with visual measurements, we construct a novel tightly-coupled optimization-based monocular VOSLAM system.

The contributions of the paper are as follows. In order to tightly and efficiently integrate the odometer measurements to VSLAM system in the framework of nonlinear optimization, it is important to provide the integrated odometer measurements between the selected keyframes. Motivated by the inertial measurement unit (IMU) preintegration theory proposed in [9], we present a novel odometer preintegration theory on manifold. The preintegration theory integrates the measurements from odometer to a single relative motion constraint that is independent of the change of linearization point, thereby eliminating the repeated computation. Besides, we also derive the corresponding uncertainty propagation and bias correction theory. Then based on the proposed preintegrated odometer measurement model, we formulate a preintegrated odometer factor and tightly integrate it to the visual-odometric optimization framework.

Secondly, since the proposed VOSLAM system requires a good initial value to bootstrap, we present a simple initialization method that builds an initial map of the environment with scale and selectively estimates the initial value of gyroscope bias. Finally, since both visual and odometer measurements

are not always available, we present a complete visual-odometric tracking mechanism to maximally exploit both measurement information. For the situation where both measurements are available, we tightly fuse both measurements to provide the accurate motion tracking. For the situation where visual information is not available, we use odometer measurements to improve the robustness of our system, and we also offer some strategies to render the visual information available as soon as possible. In addition, for the situation where wheel encoders experience slippage, we provide a strategy to detect and compensate for the faulty measurement. In this way, we can track the motion of ground robots accurately and robustly.

In experiments, we carry out extensive experiments to demonstrate the superior accuracy and robustness of our algorithm.

An overview of the proposed algorithm is shown in Fig. 1. The remainder of this article is organized as follows. Section II discusses relevant literature. In Section III, we introduce the preliminaries of the paper. Preintegrated odometer measurements and tightly-coupled visual-odometric nonlinear optimization on manifold are introduced in Section IV. In Section V, the complete monocular VOSLAM system is discussed. The experimental results are shown in Section VI. Finally, Section VII concludes the paper.

## II. RELATED WORK

There are extensive scholarly works on monocular visual odometry (VO) and VSLAM, these works rely on either filtering methods or nonlinear optimization methods. Filtering based approaches achieve higher computational efficiency due to the continuous marginalization of past state. The first real-time monocular VSLAM - MonoSLAM [10] is an extended kalman filter (EKF) based method. The standard way of computing Jacobian by filtering methods leads the system to have incorrect observability, which results in inconsistency and slightly lower accuracy of the system. To solve this problem, the first-estimates Jacobian approach was proposed in [11]. The method computed Jacobian with the first-ever available estimate instead of different linearization points, which makes the system to have correct observability and thereby improves the consistency and accuracy of the system. In addition, the observability-constrained EKF [12] was proposed to explicitly enforce the unobservable directions of the system, hence improving its consistency and accuracy.

On the other hand, nonlinear optimization based approaches can better deal with the nonlinearity of the system due to its ability to re-linearize measurement models, which leads to better accuracy at the expense of high computational cost. The first real-time optimization based monocular VSLAM system is PTAM [13] proposed by Klein and Murray. The method achieved real-time performance by dividing the SLAM system into two parallel threads. In one thread, the system performs bundle adjustment over selected keyframes and constructed map points to obtain accurate map of the environment. In the other parallel thread, the camera pose is

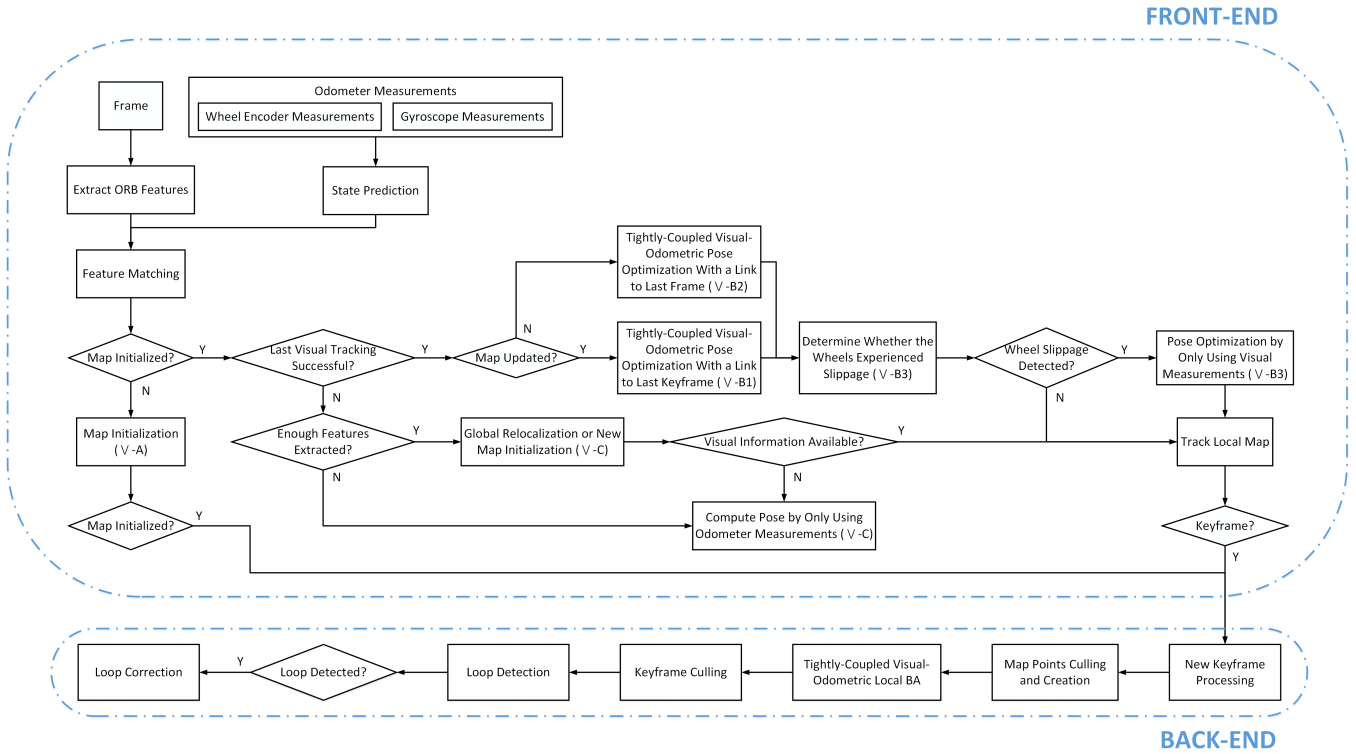


FIGURE 1: A flow chart illustrating the full pipeline of the proposed system.

tracked by minimizing the reprojection error of the features that match the reconstructed map. Based on the work of PTAM, a versatile monocular VSLAM system ORB-SLAM [14] was presented. The system introduced the third loop closing thread to eliminate the accumulated error when revisiting an already reconstructed area, it was achieved by taking advantage of bag-of-words [15] and a 7 degree-of-freedom (dof) pose graph optimization [16].

In addition, according to the definition of visual residual models, monocular VSLAM can also be categorized into feature based approaches and direct approaches. The above mentioned methods are all feature based approaches, which is quite mature and able to provide accurate estimate. However, the approaches fail to track in poorly textured environments and need to consume extra computational resources to extract and match features. In contrary, direct approaches work on raw sensor measurements, which makes the methods more efficient and able to exploit the image information where the gradient is small. Therefore, direct methods can outperform feature based methods in low texture environment. DTAM [17], SVO [18], LSD-SLAM [19], and DSO [20] are direct monocular VSLAM systems, which builds the dense or semi-dense map from monocular images in real-time, however its accuracy is still lower than the feature based semi-dense mapping technique [21].

The monocular VSLAM is scale ambiguous and sensitive to motion blur, occlusions, and illumination changes. Therefore, the monocular VSLAM systems are often com-

bined with other odometric sensors, especially IMU sensor, to achieve accurate and robust tracking system. Tightly-coupled VISLAM can also be categorized into filtering based methods and optimization based methods, where visual and inertial measurements are fused from the raw measurement level. Papers [3], [22]–[25] are filtering based monocular VISLAM approaches, which uses the inertial measurements to accurately predict the motion movement between two consecutive frames. An elegant example for filtering based visual-inertial odometry (VIO) system is the MSCKF [3], which can achieve high-precision motion estimation with computational complexity only linear in the number of features, it was achieved by presenting a visual measurement model that excluding point features from the state vector.

OKVIS [4] is an optimization based monocular VISLAM system, which tightly integrates the inertial measurements in the keyframe-based visual-inertial pipeline under the framework of graph optimization. However, in this system, the IMU integration needs to be computed repeatedly whenever the linearization point changes. In order to eliminate this repeated computation, Forster et al. presented an IMU preintegration theory, and tightly integrated the preintegrated IMU factor and visual factor in a fully probabilistic manner in [9]. Later, a new tightly-coupled monocular VISLAM system - ORB-VISLAM [5] was presented. The system can close loop and reuse the previously estimated 3D map, thereby achieving high accuracy and robustness. Recently, another tightly-coupled monocular VISLAM system was proposed in

[6] [26], which provides accurate and robust motion tracking by performing local bundle adjustment (BA) for each frame and its capability to close loop.

There are also several works on the monocular VOSLAM that fuses visual measurements with wheel encoder measurements. In [7] [8], distance information from the odometry measurements was fused with the visual measurements to render the scale factor observable, which was performed in the framework of optimization. In [27], wheel encoder measurements were combined to the system of visual odometry for accurate motion prediction, thereby true scale of the system can be recovered. In addition, the author of [2] proved that for the ground robot that moves along straight lines or circular arcs with constant acceleration, monocular VIO/VISLAM has additional unobservable directions. Therefore a system that integrates the wheel encoder measurements to VIO in a tightly-coupled manner was proposed to render the scale of the system observable.

### III. PRELIMINARIES

We begin by briefly defining the notations used throughout the paper. We employ  $(\cdot)^W$  to denote the world reference frame,  $(\cdot)^{O_k}$ ,  $(\cdot)^{C_k}$  and  $(\cdot)^{B_k}$  to denote the wheel frame, camera frame and inertial frame for the  $k^{th}$  image. In the following, we employ  $\mathbf{R}_{\mathcal{F}_2}^{\mathcal{F}_1} \in \mathbf{SO}(3)$  to represent rotation from frame  $\{\mathcal{F}_2\}$  to  $\{\mathcal{F}_1\}$  and  $\mathbf{p}_{\mathcal{F}_2}^{\mathcal{F}_1} \in \mathbb{R}^3$  to describe the 3D position of frame  $\{\mathcal{F}_2\}$  with respect to the frame  $\{\mathcal{F}_1\}$ . In addition, we use  $\mathbf{I}_{n \times n}$  to denote  $n \times n$  identity matrix and  $\mathbf{0}_{n \times m}$  to denote  $n \times m$  zero matrix.

The rotation and translation between the rigidly mounted wheel and camera sensor are  $\mathbf{R}_O^C \in \mathbf{SO}(3)$  and  $\mathbf{p}_O^C \in \mathbb{R}^3$  respectively, and  $\mathbf{R}_B^O \in \mathbf{SO}(3)$  denotes the rotation from the inertial frame to the wheel frame, these parameters can be obtained from calibration. In addition, the odometer pose for the  $k^{th}$  image is a rigid-body transformation  $\mathbf{T}_W^{O_k} = \begin{bmatrix} \mathbf{R}_W^{O_k} & \mathbf{p}_W^{O_k} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbf{SE}(3)$ , and the 3D position of the  $j^{th}$  map point in the global frame  $\{W\}$  and the camera frame  $\{C_k\}$  are denoted as  $\mathbf{f}_j^W \in \mathbb{R}^3$  and  $\mathbf{f}_j^{C_k} \in \mathbb{R}^3$  respectively.

In order to provide a minimal representation for the rigid-body transformation during the optimization, we use a vector  $\xi \in \mathbb{R}^3$  computed from the Lie algebra of  $\mathbf{SO}(3)$  to represent the over-parameterized rotation matrix  $\mathbf{R}$ . The Lie algebra of  $\mathbf{SO}(3)$  is denoted as  $\mathfrak{so}(3)$ , which is the tangent space of the manifold and coincides with the space of  $3 \times 3$  skew symmetric matrices. The logarithm map associates a rotation matrix  $\mathbf{R} \in \mathbf{SO}(3)$  to a skew symmetric matrix:

$$\xi^\wedge = \log(\mathbf{R}) \quad (1)$$

where  $(\cdot)^\wedge$  operator maps a 3-dimensional vector to a skew symmetric matrix, thus the vector  $\xi$  can be computed by using inverse  $(\cdot)^\vee$  operator:

$$\xi = \text{Log}(\mathbf{R}) = \log(\mathbf{R})^\vee \quad (2)$$

Inversely, the exponential map associates the Lie algebra  $\mathfrak{so}(3)$  to the rotation matrix  $\mathbf{R} \in \mathbf{SO}(3)$ :

$$\mathbf{R} = \text{Exp}(\xi) = \exp(\xi^\wedge) \quad (3)$$

The input of our estimation problem is a stream of measurements from the monocular camera and the odometer. The visual measurement is a set of point features extracted from the captured intensity image  $\mathbf{I}_k : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$  at time-step  $k$ . Such measurement is obtained by camera projection model  $\pi : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ , which projects the  $l^{th}$  map point  $\mathbf{f}_l^{C_k} = (x_c, y_c, z_c)^T \in \mathbb{R}^3$  expressed in the current camera frame onto the image coordinate  $\mathbf{z}_{kl} = (u, v)^T \in \Omega$ :

$$\begin{aligned} \tilde{\mathbf{z}}_{kl} &= \mathbf{z}_{kl} + \sigma_{kl} \\ &= \pi(\mathbf{f}_l^{C_k}) + \sigma_{kl} \end{aligned} \quad (4)$$

where  $\tilde{\mathbf{z}}_{kl}$  is the corresponding feature measurement, and  $\sigma_{kl}$  is the  $2 \times 1$  measurement noise with covariance  $\Sigma_{C_{kl}}$ . The projection function  $\pi$  is determined by the intrinsic parameters of the camera, which is known from calibration.

In addition, the gyroscope of odometer provides the angular velocity measurement  $\tilde{\omega}_k$  at time-step  $k$ , the measurement is assumed to be affected by a slowly time-varying bias  $\mathbf{b}_{gk}$  and a discrete-time zero-mean Gaussian white noise  $\eta_{gd} \sim \mathcal{N}(\mathbf{0}, \Sigma_{gd})$ :

$$\tilde{\omega}_k = \omega_k + \mathbf{b}_{gk} + \eta_{gd} \quad (5)$$

where gyroscope bias  $\mathbf{b}_{gk}$  is modeled as random walk, hence its derivative is Gaussian noise  $\eta_{bg} \sim \mathcal{N}(\mathbf{0}, \Sigma_{bg})$ :

$$\dot{\mathbf{b}}_{gk} = \eta_{bg} \quad (6)$$

The wheel encoders of the odometer measure the traveled distance  $\tilde{\text{Dl}}_k$  and  $\tilde{\text{Dr}}_k$  of the both wheels from time-step  $k-1$  to  $k$  at time-step  $k$ , which is assumed to be affected by a discrete-time zero-mean Gaussian white noise  $\eta_{ed}$  with variance  $\sigma_{ed}$ :

$$\begin{aligned} \tilde{\text{Dl}}_k &= \text{Dl}_k + \eta_{ed} \\ \tilde{\text{Dr}}_k &= \text{Dr}_k + \eta_{ed} \end{aligned} \quad (7)$$

Therefore, the 3D position of frame  $\{O_k\}$  with respect to frame  $\{O_{k-1}\}$  measured by wheel encoders is  $\tilde{\psi}_{O_k}^{O_{k-1}} = \begin{bmatrix} \tilde{\text{Dl}}_k + \tilde{\text{Dr}}_k & 0 & 0 \end{bmatrix}^T$ , which is affected by a Gaussian noise  $\eta_{\psi d} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\psi d})$ :

$$\begin{aligned} \tilde{\psi}_{O_k}^{O_{k-1}} &= \psi_{O_k}^{O_{k-1}} + \eta_{\psi d} \\ &= -\mathbf{R}_W^{O_{k-1}} \mathbf{R}_W^{O_k} \mathbf{p}_W^{O_k} + \mathbf{p}_W^{O_{k-1}} + \eta_{\psi d} \end{aligned} \quad (8)$$

where  $\mathbf{R}_W^{O_{k-1}}$  and  $\mathbf{p}_W^{O_{k-1}}$  constitute the pose of frame  $\{O_{k-1}\}$ , and  $\mathbf{R}_W^{O_k}$  and  $\mathbf{p}_W^{O_k}$  constitute the pose of frame  $\{O_k\}$ . In addition, the covariance of noise  $\eta_{\psi d}$  is computed from the 1-dimensional noise  $\eta_{ed}$  with  $\Sigma_{\psi d} = \frac{\sigma_{ed}}{2} \mathbf{I}_{3 \times 3}$ .

In many cases, the ground robot is moving on an approximately planar surface. The motion on a plane has 3 DoF in

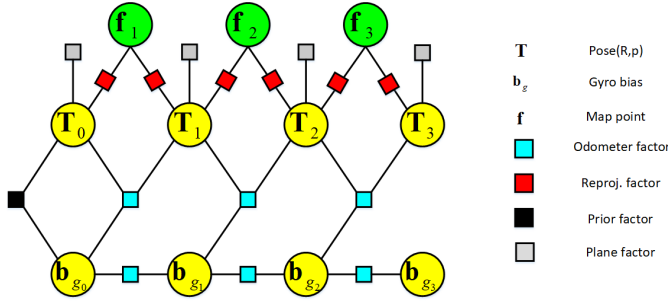


FIGURE 2: Factor graph representing the visual-odometric optimization problem. The states are shown as circles and factors are shown as squares. The blue squares represent the odometer factors and connect to the state of the last keyframe, red squares denote the visual factors corresponding to camera observations, black squares denote prior factors and gray squares denote the plane factors.

contrast to 6DoF of 3D motion. The additional information can improve the accuracy of the system. Therefore for each SE(3) pose, we also provide a planar measurement  $\tilde{\mathbf{p}}_l = [0, 0, 0]^T \in \mathbb{R}^3$  with noise  $\eta_{pl} \sim \mathcal{N}(\mathbf{0}, \Sigma_{pl})$  to constrain its roll, pitch angles and z-axis translation with respect to the physical plane.

#### IV. TIGHTLY-COUPLED VISUAL-ODOMETRIC NONLINEAR OPTIMIZATION ON MANIFOLD

We use  $\mathcal{K}$  to denote a set of successive keyframes from  $i$  to  $j$ , and  $\mathcal{L}$  to denote all the landmarks visible from the keyframes in  $\mathcal{K}$ . Then the variables to be estimated in the window of keyframes from  $i$  to  $j$  is:

$$\mathcal{X} = \{\mathbf{x}_k, \mathbf{f}_l^W\}_{k \in \mathcal{K}, l \in \mathcal{L}} \quad (9)$$

where  $\mathbf{x}_k = \{\mathbf{T}_k^{O_k}, \mathbf{b}_{g_k}\}$  is the state of the keyframe  $k$ .

We denote the visual measurements of  $\mathcal{L}$  at keyframe  $i$  as  $\mathcal{Z}_{C_i} = \{\tilde{\mathbf{z}}_{il}\}_{l \in \mathcal{L}}$ . In addition, we denote the odometer measurements between two consecutive keyframes  $i$  and  $j$  as  $\mathcal{O}_{ij} = \{\tilde{\omega}_t, \tilde{\mathbf{D}}_t, \tilde{\mathbf{D}}_t\}_{\text{time-step}(i) \leq t \leq \text{time-step}(j)}$ . Therefore, the set of measurements collected for optimizing the state  $\mathcal{X}$  is:

$$\mathcal{Z} = \{\mathcal{Z}_{C_i}, \mathcal{O}_{ij}, \tilde{\mathbf{p}}_l\}_{(i,j) \in \mathcal{K}} \quad (10)$$

##### A. MAXIMUM A POSTERIORI ESTIMATION

The optimum value of state  $\mathcal{X}$  is estimated by solving the following maximum a posteriori (MAP) problem:

$$\mathcal{X}^* = \underset{\mathcal{X}}{\operatorname{argmax}} p(\mathcal{X}|\mathcal{Z}) \quad (11)$$

which means that given the available measurements  $\mathcal{Z}$ , we want to find the best estimate for state  $\mathcal{X}$ . Assuming measurements  $\mathcal{Z}$  are independent, then using Bayes' rule, we can rewrite  $p(\mathcal{X}|\mathcal{Z})$  as:

$$\begin{aligned} p(\mathcal{X}|\mathcal{Z}) &\propto p(\mathcal{X}_0)p(\mathcal{Z}|\mathcal{X}) \\ &= p(\mathcal{X}_0) \prod_{(i,j) \in \mathcal{K}} p(\mathcal{Z}_{C_i}, \mathcal{O}_{ij}, \tilde{\mathbf{p}}_l|\mathcal{X}) \\ &= p(\mathcal{X}_0) \prod_{(i,j) \in \mathcal{K}} p(\mathcal{O}_{ij}|\mathbf{x}_i, \mathbf{x}_j) \prod_{i \in \mathcal{K}} \prod_{l \in \mathcal{L}} p(\tilde{\mathbf{z}}_{il}|\mathbf{x}_i, \mathbf{f}_l^W) \prod_{i \in \mathcal{K}} p(\tilde{\mathbf{p}}_l|\mathbf{x}_i) \end{aligned} \quad (12)$$

The equation can be interpreted as a factor graph. The variables in  $\mathcal{X}$  are corresponding to nodes in the factor graph. The terms  $p(\mathcal{X}_0)$ ,  $p(\mathcal{O}_{ij}|\mathbf{x}_i, \mathbf{x}_j)$ ,  $p(\tilde{\mathbf{z}}_{il}|\mathbf{x}_i, \mathbf{f}_l^W)$ , and  $p(\tilde{\mathbf{p}}_l|\mathbf{x}_i)$  are called factors, which encodes probabilistic constraints between nodes. A factor graph representing the problem is shown in Fig. 2.

The MAP estimate corresponds to the minimum of the negative log-posterior. Under the assumption of zero-mean Gaussian noise, the MAP estimate in (11) can be written as the minimum of the sum of squared residual errors:

$$\begin{aligned} \mathcal{X}^* &= \underset{\mathcal{X}}{\operatorname{argmin}} -\log p(\mathcal{X}|\mathcal{Z}) \\ &= \underset{\mathcal{X}}{\operatorname{argmin}} \|\mathbf{r}_0\|_{\Sigma_0}^2 + \sum_{(i,j) \in \mathcal{K}} \rho\left(\|\mathbf{r}_{\mathcal{O}_{ij}}\|_{\Sigma_{\mathcal{O}_{ij}}}^2\right) \\ &\quad + \sum_{i \in \mathcal{K}} \sum_{l \in \mathcal{L}} \rho\left(\|\mathbf{r}_{\mathcal{C}_{il}}\|_{\Sigma_{\mathcal{C}_{il}}}^2\right) + \sum_{i \in \mathcal{K}} \rho\left(\|\mathbf{r}_{\mathbf{p}_l}\|_{\Sigma_{\mathbf{p}_l}}^2\right) \end{aligned} \quad (13)$$

where  $\mathbf{r}_0$ ,  $\mathbf{r}_{\mathcal{O}_{ij}}$ ,  $\mathbf{r}_{\mathcal{C}_{il}}$ , and  $\mathbf{r}_{\mathbf{p}_l}$  are the prior error, odometer error, reprojection error, and plane error respectively, as well as  $\Sigma_0$ ,  $\Sigma_{\mathcal{O}_{ij}}$ ,  $\Sigma_{\mathcal{C}_{il}}$ , and  $\Sigma_{\mathbf{p}_l}$  are the corresponding covariance matrices, and  $\rho$  is the Huber robust cost function. In the following subsections, we provide detailed expressions for these residual errors.

##### B. PREINTEGRATED ODOMETER MEASUREMENT

In this section, we derive the preintegrated odometer measurement between two consecutive keyframes  $i$  and  $j$  by assuming the gyroscope bias of keyframe  $i$  is known. Firstly, we define the rotation increment  $\Delta \mathbf{R}_{ij}$  and position increment  $\Delta \mathbf{p}_{ij}$  in frame  $\{\mathcal{O}_i\}$  as:

$$\begin{aligned} \Delta \mathbf{R}_{ij} &= \prod_{k=i}^{j-1} \mathbf{R}_B^O \operatorname{Exp}\left((\tilde{\omega}_k - \mathbf{b}_{g_k} - \eta_{gd})\Delta t\right) \mathbf{R}_B^{O^T} \\ \Delta \mathbf{p}_{ij} &= \sum_{k=i+1}^j \Delta \mathbf{R}_{ik-1}(\tilde{\psi}_{\mathcal{O}_k}^{O_{k-1}} - \eta_{\psi d}) \end{aligned} \quad (14)$$

where  $\mathbf{b}_{g_k} = \mathbf{b}_{g_i} + \sum_{n=i}^{k-1} \eta_{b_g} \Delta t$ , which is obtained by integrating (6). Because the gyroscope bias at keyframe  $i$  is known, the mean of  $\mathbf{b}_{g_k}$  is  $\mathbf{b}_{g_i}$  and  $\delta \mathbf{b}_{g_{ik}} = \sum_{n=i}^{k-1} \eta_{b_g} \Delta t$  is the bias noise. Bias noise  $\delta \mathbf{b}_{g_{ik}}$  is a zero-mean Gaussian noise, since it is a linear combination of zero-mean Gaussian noise  $\eta_{b_g}$ . Then by using the first-order approximation and dropping higher-order noise terms, we split each increment in



(14) to preintegrated measurement and its noise. For rotation, we have:

$$\begin{aligned}\Delta \mathbf{R}_{ij} &= \prod_{k=i}^{j-1} \text{Exp} \left( \mathbf{R}_B^O(\tilde{\omega}_k - \mathbf{b}_{g_i} - \delta \mathbf{b}_{g_{ik}} - \boldsymbol{\eta}_{gd}) \Delta t \right) \\ &\approx \prod_{k=i}^{j-1} \left[ \text{Exp} \left( \mathbf{R}_B^O(\tilde{\omega}_k - \mathbf{b}_{g_i}) \Delta t \right) \text{Exp} \left( -\mathbf{J}_{r_k} \mathbf{R}_B^O(\delta \mathbf{b}_{g_{ik}} + \boldsymbol{\eta}_{gd}) \Delta t \right) \right] \\ &= \Delta \tilde{\mathbf{R}}_{ij} \prod_{k=i}^{j-1} \text{Exp} \left( -\Delta \tilde{\mathbf{R}}_{k+1j}^T \mathbf{J}_{r_k} \mathbf{R}_B^O(\delta \mathbf{b}_{g_{ik}} + \boldsymbol{\eta}_{gd}) \Delta t \right) \\ &= \Delta \tilde{\mathbf{R}}_{ij} \text{Exp}(\delta \phi_{ij})\end{aligned}\quad (15)$$

where  $\mathbf{J}_{r_k} = \mathbf{J}_r(\mathbf{R}_B^O(\tilde{\omega}_k - \mathbf{b}_{g_i}) \Delta t)$  is the right Jacobian of SO(3). Therefore, we obtain the preintegrated rotation measurement:

$$\Delta \tilde{\mathbf{R}}_{ij} = \prod_{k=i}^{j-1} \text{Exp} \left( \mathbf{R}_B^O(\tilde{\omega}_k - \mathbf{b}_{g_i}) \Delta t \right) \quad (16)$$

For position, we have:

$$\begin{aligned}\Delta \mathbf{p}_{ij} &\approx \sum_{k=i+1}^j \left[ \Delta \tilde{\mathbf{R}}_{ik-1} (\mathbf{I}_{3 \times 3} + \delta \phi_{ik-1}^\wedge) (\tilde{\psi}_{O_k}^{O_{k-1}} - \boldsymbol{\eta}_{\psi d}) \right] \\ &= \sum_{k=i+1}^j \left[ \Delta \tilde{\mathbf{R}}_{ik-1} (\mathbf{I}_{3 \times 3} + \delta \phi_{ik-1}^\wedge) \tilde{\psi}_{O_k}^{O_{k-1}} - \Delta \tilde{\mathbf{R}}_{ik-1} \boldsymbol{\eta}_{\psi d} \right] \\ &= \Delta \tilde{\mathbf{p}}_{ij} + \sum_{k=i+1}^j \left[ -\Delta \tilde{\mathbf{R}}_{ik-1} \tilde{\psi}_{O_k}^{O_{k-1}^\wedge} \delta \phi_{ik-1} - \Delta \tilde{\mathbf{R}}_{ik-1} \boldsymbol{\eta}_{\psi d} \right] \\ &= \Delta \tilde{\mathbf{p}}_{ij} + \delta \mathbf{p}_{ij}\end{aligned}\quad (17)$$

Therefore, we obtain the preintegrated position measurement:

$$\Delta \tilde{\mathbf{p}}_{ij} = \sum_{k=i+1}^j \Delta \tilde{\mathbf{R}}_{ik-1} \tilde{\psi}_{O_k}^{O_{k-1}} \quad (18)$$

### C. NOISE PROPAGATION

As defined in Section IV-B, the bias noise is:

$$\delta \mathbf{b}_{g_{ij}} = \sum_{k=i}^{j-1} \boldsymbol{\eta}_{b_g} \Delta t \quad (19)$$

Then obtained from (15), the rotation noise is:

$$\delta \phi_{ij} = \sum_{k=i}^{j-1} \left[ -\Delta \tilde{\mathbf{R}}_{k+1j}^T \mathbf{J}_{r_k} \mathbf{R}_B^O(\delta \mathbf{b}_{g_{ik}} + \boldsymbol{\eta}_{gd}) \Delta t \right] \quad (20)$$

The rotation noise term  $\delta \phi_{ij}$  is zero-mean and Gaussian, since it is a linear combination of zero-mean white Gaussian noise  $\boldsymbol{\eta}_{gd}$  and  $\delta \mathbf{b}_{g_{ij}}$ .

Furthermore, from (17), we obtain the position noise:

$$\delta \mathbf{p}_{ij} = \sum_{k=i+1}^j \left[ -\Delta \tilde{\mathbf{R}}_{ik-1} \tilde{\psi}_{O_k}^{O_{k-1}^\wedge} \delta \phi_{ik-1} - \Delta \tilde{\mathbf{R}}_{ik-1} \boldsymbol{\eta}_{\psi d} \right] \quad (21)$$

The position noise  $\delta \mathbf{p}_{ij}$  is also zero-mean Gaussian noise, because it is a linear combination of the noise  $\boldsymbol{\eta}_{\psi d}$  and the rotation noise  $\delta \phi_{ij}$ .

We write (19), (20) and (21) in iterative form, then the noise propagation can be written in matrix form as:

$$\begin{bmatrix} \delta \phi_{ik+1} \\ \delta \mathbf{p}_{ik+1} \\ \delta \mathbf{b}_{g_{ik+1}} \end{bmatrix} = \begin{bmatrix} \Delta \tilde{\mathbf{R}}_{kk+1}^T & \mathbf{0}_{3 \times 3} & -\mathbf{J}_{r_k} \mathbf{R}_B^O \Delta t \\ -\Delta \tilde{\mathbf{R}}_{ik}^T \tilde{\psi}_{O_{k+1}}^\wedge & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \begin{bmatrix} \delta \phi_{ik} \\ \delta \mathbf{p}_{ik} \\ \delta \mathbf{b}_{g_{ik}} \end{bmatrix} + \begin{bmatrix} -\mathbf{J}_{r_k} \mathbf{R}_B^O \Delta t & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -\Delta \tilde{\mathbf{R}}_{ik} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \Delta t \end{bmatrix} \begin{bmatrix} \boldsymbol{\eta}_{gd} \\ \boldsymbol{\eta}_{\psi d} \\ \boldsymbol{\eta}_{b_g} \end{bmatrix} \quad (22)$$

or more simply:

$$\mathbf{n}_{ik+1} = \mathbf{A} \mathbf{n}_{ik} + \mathbf{B} \boldsymbol{\eta} \quad (23)$$

Given the linear model (23) and the covariance  $\boldsymbol{\Sigma}_\eta \in \mathbb{R}^{9 \times 9}$  of the odometer measurements noise  $\boldsymbol{\eta}$ , it is possible to compute the covariance of the odometer preintegration noise iteratively:

$$\boldsymbol{\Sigma}_{O_{ik+1}} = \mathbf{A} \boldsymbol{\Sigma}_{O_{ik}} \mathbf{A}^T + \mathbf{B} \boldsymbol{\Sigma}_\eta \mathbf{B}^T \quad (24)$$

with initial condition  $\boldsymbol{\Sigma}_{O_{i1}} = \mathbf{0}_{9 \times 9}$ .

Therefore, we can fully characterize the preintegrated odometer measurements noise as:

$$\mathbf{n}_{ij} = \begin{bmatrix} \delta \phi_{ij}^T & \delta \mathbf{p}_{ij}^T & \delta \mathbf{b}_{g_{ij}}^T \end{bmatrix}^T \sim \mathcal{N}(\mathbf{0}_{9 \times 1}, \boldsymbol{\Sigma}_{O_{ij}}) \quad (25)$$

### D. BIAS UPDATE

In the previous section, we assumed that the gyroscope bias  $\mathbf{b}_{g_i}$  is fixed. Given the bias change  $\mathbf{b}_{g_i} \leftarrow \bar{\mathbf{b}}_{g_i} + \delta \mathbf{b}_{g_i}$ , we can update the preintegrated measurements by using the first-order approximation. For preintegrated rotation measurement:

$$\begin{aligned}\Delta \tilde{\mathbf{R}}_{ij}(\mathbf{b}_{g_i}) &= \prod_{k=i}^{j-1} \text{Exp} \left( \mathbf{R}_B^O(\tilde{\omega}_k - \bar{\mathbf{b}}_{g_i} - \delta \mathbf{b}_{g_i}) \Delta t \right) \\ &\approx \prod_{k=i}^{j-1} \left[ \text{Exp} \left( \mathbf{R}_B^O(\tilde{\omega}_k - \bar{\mathbf{b}}_{g_i}) \Delta t \right) \text{Exp} \left( -\mathbf{J}_{r_k} \mathbf{R}_B^O \delta \mathbf{b}_{g_i} \Delta t \right) \right] \\ &= \Delta \tilde{\mathbf{R}}_{ij}(\bar{\mathbf{b}}_{g_i}) \prod_{k=i}^{j-1} \text{Exp} \left( -\Delta \tilde{\mathbf{R}}_{k+1j}^T \mathbf{J}_{r_k} \mathbf{R}_B^O \delta \mathbf{b}_{g_i} \Delta t \right) \\ &= \Delta \tilde{\mathbf{R}}_{ij}(\bar{\mathbf{b}}_{g_i}) \text{Exp} \left( \frac{\partial \Delta \tilde{\mathbf{R}}_{ij}}{\partial \mathbf{b}_{g_i}} \delta \mathbf{b}_{g_i} \right)\end{aligned}\quad (26)$$

where  $\frac{\partial \Delta \bar{\mathbf{R}}_{ij}}{\partial \mathbf{b}_{g_i}} = \sum_{k=i}^{j-1} -\Delta \tilde{\mathbf{R}}_{k+1,j}^T \mathbf{J}_{r_k} \mathbf{R}_B^O \Delta t$ . For preintegrated position measurement:

$$\begin{aligned} \Delta \tilde{\mathbf{p}}_{ij}(\mathbf{b}_{g_i}) &= \sum_{k=i+1}^j \Delta \tilde{\mathbf{R}}_{ik-1}(\bar{\mathbf{b}}_{g_i}) \text{Exp} \left( \frac{\partial \Delta \bar{\mathbf{R}}_{ik-1}}{\partial \mathbf{b}_{g_i}} \delta \mathbf{b}_{g_i} \right) \tilde{\psi}_{O_k}^{O_{k-1}} \\ &\approx \sum_{k=i+1}^j \Delta \tilde{\mathbf{R}}_{ik-1}(\bar{\mathbf{b}}_{g_i}) \left( \mathbf{I}_{3 \times 3} + \left( \frac{\partial \Delta \bar{\mathbf{R}}_{ik-1}}{\partial \mathbf{b}_{g_i}} \delta \mathbf{b}_{g_i} \right)^\wedge \right) \tilde{\psi}_{O_k}^{O_{k-1}} \\ &= \Delta \tilde{\mathbf{p}}_{ij}(\bar{\mathbf{b}}_{g_i}) - \sum_{k=i+1}^j \Delta \tilde{\mathbf{R}}_{ik-1}(\bar{\mathbf{b}}_{g_i}) \tilde{\psi}_{O_k}^{O_{k-1}^\wedge} \frac{\partial \Delta \bar{\mathbf{R}}_{ik-1}}{\partial \mathbf{b}_{g_i}} \delta \mathbf{b}_{g_i} \\ &= \Delta \tilde{\mathbf{p}}_{ij}(\bar{\mathbf{b}}_{g_i}) + \frac{\partial \Delta \tilde{\mathbf{p}}_{ij}}{\partial \mathbf{b}_{g_i}} \delta \mathbf{b}_{g_i} \end{aligned} \quad (27)$$

$$\text{where } \frac{\partial \Delta \tilde{\mathbf{p}}_{ij}}{\partial \mathbf{b}_{g_i}} = \sum_{k=i+1}^j -\Delta \tilde{\mathbf{R}}_{ik-1}(\bar{\mathbf{b}}_{g_i}) \tilde{\psi}_{O_k}^{O_{k-1}^\wedge} \frac{\partial \Delta \bar{\mathbf{R}}_{ik-1}}{\partial \mathbf{b}_{g_i}}.$$

### E. PREINTEGRATED ODOMETER FACTOR

From the geometric relation between two consecutive keyframes  $i$  and  $j$ , we get our preintegrated odometer measurement model as:

$$\begin{aligned} \mathbf{R}_W^{O_i} \mathbf{R}_W^{O_j T} &= \Delta \tilde{\mathbf{R}}_{ij}(\mathbf{b}_{g_i}) \text{Exp}(\delta \phi_{ij}) \\ -\mathbf{R}_W^{O_i} \mathbf{R}_W^{O_j T} \mathbf{p}_W^{O_j} + \mathbf{p}_W^{O_i} &= \Delta \tilde{\mathbf{p}}_{ij}(\mathbf{b}_{g_i}) + \delta \mathbf{p}_{ij} \end{aligned} \quad (28)$$

Therefore, the preintegrated odometer residual  $\mathbf{r}_{O_{ij}} = [\mathbf{r}_{\Delta \mathbf{R}_{ij}}^T, \mathbf{r}_{\Delta \mathbf{p}_{ij}}^T, \mathbf{r}_{\Delta \mathbf{b}_{g_{ij}}}^T]^T \in \mathbb{R}^9$  can be defined as:

$$\begin{aligned} \mathbf{r}_{\Delta \mathbf{R}_{ij}} &= \text{Log} \left( \left( \Delta \tilde{\mathbf{R}}_{ij}(\bar{\mathbf{b}}_{g_i}) \text{Exp} \left( \frac{\partial \Delta \bar{\mathbf{R}}_{ij}}{\partial \mathbf{b}_{g_i}} \delta \mathbf{b}_{g_i} \right) \right)^T \mathbf{R}_W^{O_i} \mathbf{R}_W^{O_j T} \right) \\ \mathbf{r}_{\Delta \mathbf{p}_{ij}} &= -\mathbf{R}_W^{O_i} \mathbf{R}_W^{O_j T} \mathbf{p}_W^{O_j} + \mathbf{p}_W^{O_i} - \left( \Delta \tilde{\mathbf{p}}_{ij}(\bar{\mathbf{b}}_{g_i}) + \frac{\partial \Delta \tilde{\mathbf{p}}_{ij}}{\partial \mathbf{b}_{g_i}} \delta \mathbf{b}_{g_i} \right) \\ \mathbf{r}_{\Delta \mathbf{b}_{g_{ij}}} &= \mathbf{b}_{g_j} - \mathbf{b}_{g_i} \end{aligned} \quad (29)$$

### F. VISUAL FACTOR

Through the measurement model in (4), the  $l^{\text{th}}$  map point expressed in the world reference frame  $\{W\}$  can be projected onto the image plane of the  $i^{\text{th}}$  keyframe as:

$$\mathbf{z}_{il} = \pi(\mathbf{R}_O^C \mathbf{R}_W^{O_i} * \mathbf{f}_l^W + \mathbf{R}_O^C \mathbf{p}_W^{O_i} + \mathbf{p}_O^C) \quad (30)$$

Therefore, the reprojection error  $\mathbf{r}_{C_{il}} \in \mathbb{R}^2$  for the  $l^{\text{th}}$  map point seen by the  $i^{\text{th}}$  keyframe is:

$$\mathbf{r}_{C_{il}} = \mathbf{z}_{il} - \tilde{\mathbf{z}}_{il} \quad (31)$$

### G. PLANE FACTOR

The x-y plane of the first wheel frame  $\{O_1\}$  coincides with the physical plane. Therefore, we express the plane factor  $\mathbf{r}_{\text{pl}_k} \in \mathbb{R}^3$  as:

$$\mathbf{r}_{\text{pl}_k} = \begin{bmatrix} [\mathbf{e}_1 \ \mathbf{e}_2]^T \mathbf{R}_W^{O_k} \mathbf{R}_W^{O_1 T} \mathbf{e}_3 \\ \mathbf{e}_3^T (-\mathbf{R}_W^{O_1} \mathbf{R}_W^{O_k T} \mathbf{p}_W^{O_k} + \mathbf{p}_W^{O_1}) \end{bmatrix} - \tilde{\mathbf{p}}_k \quad (32)$$

where  $\mathbf{e}_1 = [1 \ 0 \ 0]^T$ ,  $\mathbf{e}_2 = [0 \ 1 \ 0]^T$  and  $\mathbf{e}_3 = [0 \ 0 \ 1]^T$ .

The first two elements in (32) is the planar rotational constraint, it means that rotating  $\mathbf{e}_3$  vector in frame  $\{O_1\}$  to frame  $\{O_k\}$ , the result should also be  $\mathbf{e}_3$ . The result is equal to  $\mathbf{e}_3$  if and only if the roll and pitch angles between frames  $\{O_1\}$  and  $\{O_k\}$  are all zero. Thereby constraining the first two elements in the residual to zero corresponds to constraining the roll and pitch angles between two frames to zero. The third element in (32) is the planar translational constraint, which means that z-axis translation between frames  $\{O_1\}$  and  $\{O_k\}$  should be zero. In addition, the covariance of the planar measurement  $\Sigma_{\text{pl}}$  is set as  $\text{diag}(0.0012, 0.0012, 0.0004)$  in this paper to allow  $2^\circ$  deviation for roll and pitch angles and 0.02m deviation for z-axis translation.

## V. MONOCULAR VOSLAM SYSTEM

Our monocular VOSLAM system is inspired by the ORB-SLAM [14] and ORB-VISLAM [5] methods. Fig. 1 shows an overview of our system. In this section, we detail the main changes of our VOSLAM system with respect to the referenced system.

### A. MAP INITIALIZATION

The map initialization is in charge of estimating two initial values. Firstly, we estimate the initial value for gyroscope bias  $\mathbf{b}_g$ , this calculation is not necessary. However computing its initial value can achieve better accuracy as shown in the seventh and eighth columns of Table 2, therefore we compute the initial  $\mathbf{b}_g$  in this paper. Then we construct the initial map with scale.

The steps of map initialization are as follows. Firstly, we search for feature matches between the current frame  $k$  and reference frame  $r$ . If there are sufficient feature matches, we add this frame to the local window of gyroscope bias initialization and perform the next step, else we set the current frame as reference frame and clear the local window. The second step is to check the parallax of each correspondence and pick out a set of feature matches  $\mathcal{F}$  that have sufficient parallax. When the size of  $\mathcal{F}$  is greater than a threshold, we perform the next step. The next steps are different depending on whether to estimate the initial value for  $\mathbf{b}_g$ .

If we estimate the initial value for gyroscope bias, in third step, we firstly compute the up-to-scale relative transformation  $\mathbf{T}_{C_k}^{C_r}$  between reference frame  $r$  and current frame  $k$  by using the Five-point algorithm [28], and triangulate feature matches  $\mathcal{F}$ . Then if the size of successfully triangulated map points is larger than a threshold, we compute the pose  $\mathbf{T}_{C_i}^{C_r}$  of all frames in the local window by using the perspective-n-point (PnP) method [29], and perform a pure visual global BA to optimize the pose of all frames in the local window. If the size of successfully triangulated map points is still larger than a threshold after the optimization, we perform the next step. In step four, we compute the preintegrated odometer measurement  $\Delta \tilde{\mathbf{R}}_{ri}$  for all the frames in the local window of gyroscope bias initialization, and combine it with

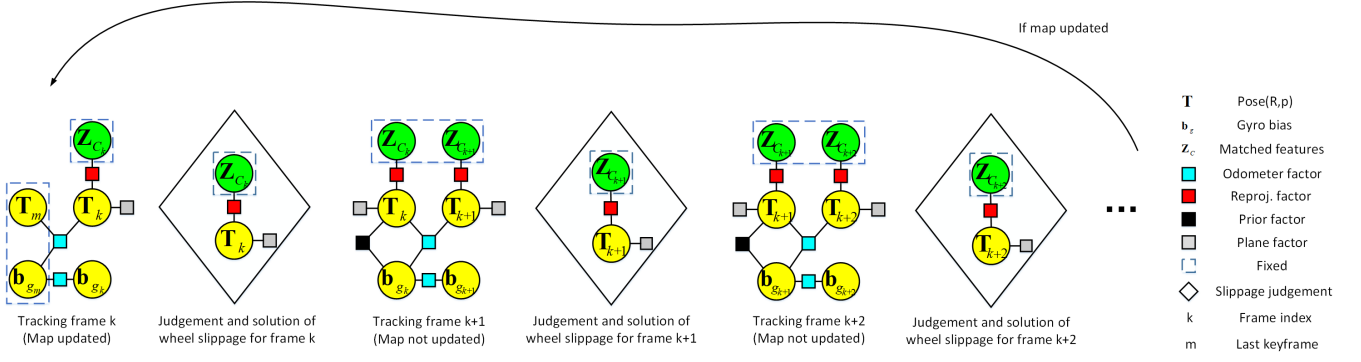


FIGURE 3: Evolution of the factor graph in motion tracking when the last visual tracking is successful. If map is updated, we optimize the state of frame  $k$  by connecting an odometer factor to last keyframe  $m$ . If map is not changed, state of both last frame  $k-1$  and current frame  $k$  are jointly optimized by linking an odometer factor between them and adding a prior factor to last frame  $k-1$ . The prior for last frame  $k-1$  is obtained from the last optimization. At the end of each joint optimization, through the optimized result we determine whether the wheel slippage has occurred. If wheel slippage is detected, the pose of frame  $k$  is re-optimized using factor graph in the diamond.

the computed relative rotation  $\mathbf{R}_{O_i}^{O_r} = \mathbf{R}_O^C \mathbf{R}_{C_i}^{C_r} \mathbf{R}_O^C$  from the last step to estimate the initial gyroscope bias:

$$\arg \min_{\mathbf{b}_g} \sum_{i=r+1}^k \left\| \text{Log} \left( \left( \Delta \tilde{\mathbf{R}}_{ri} \text{Exp} \left( \frac{\partial \Delta \tilde{\mathbf{R}}_{ri}}{\partial \mathbf{b}_g} \mathbf{b}_g \right) \right)^T \mathbf{R}_{O_i}^{O_r} \right) \right\| \quad (33)$$

which is derived from (26) (28). Next, based on the estimated gyroscope bias  $\mathbf{b}_g$ , we re-compute the odometer preintegration terms  $\Delta \tilde{\mathbf{R}}_{rk}$  and  $\Delta \tilde{\mathbf{p}}_{rk}$ , and reset the pose of current frame as  $\mathbf{T}_W^{O_k} = \Delta \tilde{\mathbf{T}}_{rk}^{-1} \mathbf{T}_W^{O_r}$ , where the pose  $\mathbf{T}_W^{O_r}$  of reference frame can be set to arbitrary value, in this paper we set it to the identity. Based on the re-computed pose, we re-triangulate the matched features  $\mathcal{F}$ . Then if the size of the successfully created map points is greater than a threshold, we set the reference frame and current frame as keyframes.

If we do not estimate the initial value for  $\mathbf{b}_g$ , in third step, we use the odometer measurements to compute the relative transformation  $\Delta \tilde{\mathbf{T}}_{rk}$  between reference frame  $r$  and current frame  $k$ , and set the pose of current frame as  $\mathbf{T}_W^{O_k} = \Delta \tilde{\mathbf{T}}_{rk}^{-1} \mathbf{T}_W^{O_r}$ , where the pose  $\mathbf{T}_W^{O_r}$  of reference frame can be set to arbitrary value, in this paper we set it to the identity. Then we triangulate the matched features  $\mathcal{F}$ . If the size of the successfully created map points is greater than a threshold, we set reference frame and current frame as keyframes.

The created two keyframes and the constructed initial map points constitute the initial map. Finally, a global BA that minimizes all the reprojection errors, odometer errors, and plane errors contained in the initial map is performed to refine the initial map.

## B. TRACKING WHEN LAST VISUAL TRACKING IS SUCCESSFUL

For tracking the motion of current frame when the last visual tracking is successful, we firstly use the odometer measure-

ments to predict the initial pose of current frame. Based on the initial pose, we match the keypoints extracted from the current frame to the map points seen by last frame. Then we use all the available visual and odometer measurements to optimize the pose of current frame. The confidence of the last state estimate is different according to whether the map is updated in back-end, thus we adopt different strategies to track the motion of current frame depending on whether the map is updated. When the map is updated in back-end, the state estimate of last keyframe optimized by local BA or pose graph optimization has high confidence, thus we fix the optimized state of last keyframe to optimize the pose of current frame as described in Section V-B1. Then when the map is not updated in back-end, the last state optimized by motion tracking is not as confident as the state optimized in back-end, thus instead of fixing last state estimate, we add a prior to the state of last frame to optimize the pose of current frame as described in Section V-B2. Finally, according to the optimized result, we use strategy described in Section V-B3 to detect and solve the wheel slippage. The tracking mechanism is illustrated graphically in Fig. 3.

### 1) Tracking when Map is Updated

When tracking is performed just after the map is updated in the back-end, we firstly compute the preintegrated odometer measurement between current frame  $k$  and last keyframe  $m$ . Then the preintegrated measurement  $\Delta \mathbf{R}_{O_k}^{O_m}$  and  $\Delta \mathbf{p}_{O_k}^{O_m}$  are combined with the optimized pose of last keyframe to predict the initial pose of the current frame. Finally, we optimize the state of the current frame  $k$  by minimizing the following energy function:

$$\gamma = \{\mathbf{x}_k\}$$

$$\gamma^* = \arg \min_{\gamma} \left( \sum_{l \in \mathcal{Z}_{C_k}} \|\mathbf{r}_{C_{kl}}\|_{\Sigma_{C_{kl}}}^2 + \|\mathbf{r}_{O_{mk}}\|_{\Sigma_{O_{mk}}}^2 + \|\mathbf{r}_{Pl_k}\|_{\Sigma_{Pl}}^2 \right) \quad (34)$$



After the optimization, the resulting estimation and Hessian matrix are served as a prior for next optimization.

## 2) Tracking when Map is not Updated

When the map is not changed in the back-end, we compute the preintegrated odometer measurement between current frame  $k$  and last frame  $k-1$ , and predict the initial pose of current frame  $k$  by integrating the preintegrated measurement  $\Delta \mathbf{R}_{O_k}^{O_{k-1}}$  and  $\Delta \mathbf{p}_{O_k}^{O_{k-1}}$  to the pose of last frame. Then, we optimize the pose of current frame  $k$  by performing the non-linear optimization that minimizing the following objective function:

$$\begin{aligned} \gamma &= \{\mathbf{x}_{k-1}, \mathbf{x}_k\} \\ \gamma^* &= \underset{\gamma}{\operatorname{argmin}} \left( \sum_{l \in \mathcal{Z}_{C_{k-1}}} \|\mathbf{r}_{C_{k-1}l}\|_{\Sigma_{C_{k-1}l}}^2 + \sum_{n \in \mathcal{Z}_{C_k}} \|\mathbf{r}_{C_{kn}}\|_{\Sigma_{C_{kn}}}^2 \right. \\ &\quad \left. + \|\mathbf{r}_{O_{k-1}k}\|_{\Sigma_{O_{k-1}k}}^2 + \|\mathbf{r}_{0_{k-1}}\|_{\Sigma_{0_{k-1}}}^2 + \|\mathbf{r}_{p_{k-1}}\|_{\Sigma_{p_{k-1}}}^2 + \|\mathbf{r}_{p_{1k}}\|_{\Sigma_{p_{1k}}}^2 \right) \end{aligned} \quad (35)$$

where the residual  $\mathbf{r}_{0_{k-1}} = \{\mathbf{r}_{\mathbf{R}_{k-1}}^T, \mathbf{r}_{\mathbf{p}_{k-1}}^T, \mathbf{r}_{\mathbf{b}_{g_{k-1}}}^T\}^T \in \mathbb{R}^9$  is a prior error term of last frame:

$$\begin{aligned} \mathbf{r}_{\mathbf{R}_{k-1}} &= \operatorname{Log} \left( \tilde{\mathbf{R}}_W^{O_{k-1}T} \mathbf{R}_W^{O_{k-1}} \right) \\ \mathbf{r}_{\mathbf{p}_{k-1}} &= \mathbf{p}_\omega^{O_{k-1}} - \tilde{\mathbf{p}}_\omega^{O_{k-1}} \\ \mathbf{r}_{\mathbf{b}_{g_{k-1}}} &= \mathbf{b}_{g_{k-1}} - \tilde{\mathbf{b}}_{g_{k-1}} \end{aligned} \quad (36)$$

where  $\tilde{\mathbf{R}}_W^{O_{k-1}}$ ,  $\tilde{\mathbf{p}}_\omega^{O_{k-1}}$ ,  $\tilde{\mathbf{b}}_{g_{k-1}}$  and  $\Sigma_{0_{k-1}}$  are the estimated states and resulting covariance matrix from last pose optimization. The optimized result is also served as a prior for next optimization.

## 3) Detecting and Solving Wheel Slippage

Wheel encoders are an ambivalent sensor, it provides reliable traveled distance measurements of each wheel at most of the time, but it can also deliver very faulty data when the robot experiences slippage. If we perform visual-odometric joint optimization using this kind of faulty data, in order to simultaneously satisfy the constraints of both odometer measurements with slippage and visual measurements, the optimization will lead to a false estimate. Therefore, we provide a strategy to detect and solve this case. We think the current frame  $k$  experienced a slippage if the above visual-odometric optimization (34) (35) makes more than half of the originally matched features become outliers. Once the wheel slippage is detected, we set slippage flag to current frame and reset the initial pose of current frame  $k$  as the pose of last frame  $k-1$ . Then we re-match the features of the current frame to the map points seen by last frame. Finally, the state of current frame is optimized by only using those matched features:

$$\begin{aligned} \gamma &= \{\mathbf{x}_k\} \\ \gamma^* &= \underset{\gamma}{\operatorname{argmin}} \left( \sum_{l \in \mathcal{Z}_{C_k}} \|\mathbf{r}_{C_{kl}}\|_{\Sigma_{C_{kl}}}^2 + \|\mathbf{r}_{p_{1k}}\|_{\Sigma_{p_{1k}}}^2 \right) \end{aligned} \quad (37)$$

In this way, we can detect and solve the wheel slippage efficiently. Therefore the system can provide accurate pose estimation all the time. After the optimization, the resulting estimate and Hessian matrix of current frame are served as a prior for next optimization.

## C. TRACKING WHEN LAST VISUAL TRACKING IS LOST

If visual tracking information is not available for the motion tracking of current frame, we can only use odometer measurements to compute the pose of the frame. So in order to obtain more accurate pose estimate, we should make the visual information available as early as possible.

Supposing the last visual tracking is lost, then one of the three cases will happen for the current frame: (1) the robot revisits to an already reconstructed area; (2) the robot visits to a new environment where exists sufficient features; (3) the visual features are still unavailable wherever the robot is. For these different situations, we perform different strategies to estimate the pose of the current frame. For case 1, a global relocalization method as done in [14], i.e. using DBOW [15] and PnP algorithm [29], is performed to compute the pose of the current frame and render the visual information available. For case 2, using the pose estimated from the solution of case 3, we re-perform the map initialization procedure in Section V-A to construct new map that is connected to the previous map, therefore making the visual information available. For case 3, we use the odometer measurements to compute the pose of current frame, which improves the robustness of the system to visual loss.

When enough features are extracted from the current frame, we firstly think the robot may returned to an already reconstructed environment and perform the global relocalization method (solution for case 1). However, if the relocalization has continuously failed for the 20 frames with enough features, we think the robot entered into a new environment and then construct the new map as solution for case 2. We deem the visual information becomes available for motion tracking of current frame when the pose of current frame computed from the above two cases is supported by enough matched features. So if the pose is not supported by enough matched features or fewer features are extracted from the current frame, we think the visual information is still unavailable for motion tracking of the current frame and set the pose of current frame according to the solution for case 3.

## D. TRACK LOCAL MAP AND KEYFRAME SELECTION

When the current visual tracking is successful, we match the keypoints in current frame to the local map for constructing compact covisibility graph, which can greatly improve the accuracy of the system. Then we insert a keyframe to back-end when the following criteria are satisfied: (1) current frame tracks less than 90% features than last keyframe; (2) Local BA is finished in back-end. These criteria ensure a good visual tracking of the system.

### E. BACK-END

The back-end includes the local mapping thread and the loop closing thread as in paper [14] [5]. The local mapping thread aims to update the covisibility graph, construct the new map points of the map and optimize the local map. When new keyframe  $k$  is inserted to local mapping thread, we make small change in local BA with respect to paper [5]. Visual-odometric local BA minimizing the cost function (13) is performed to optimize the last  $N$  keyframes in local window and all the map points seen by those  $N$  keyframes. One thing to note is that the odometer constraint linking to the last keyframe is only constructed for those keyframes without the slippage flag. The loop closing thread is in charge of eliminating the accumulated drift when returning to an already reconstructed area, it is implemented in the same way as paper [5].

### VI. EXPERIMENTS

In the following, we perform a number of experiments to evaluate the proposed VOSLAM system both qualitatively and quantitatively. Firstly in Section VI-A and Section VI-B, we perform the qualitative and quantitative analysis to show the accuracy of our system. Then the validity of the proposed strategy for detecting and solving the wheel slippage is demonstrated in Section VI-C. Finally in Section VI-D, we test the robustness of our algorithm to visual loss. The experiments are performed on a laptop with Intel Core i5 2.2GHz CPU and an 8GB RAM. The corresponding videos are available at: <https://youtu.be/EaDTC92hQpc>.

#### A. ALGORITHM EVALUATION ON THE DS DATASET

The DS dataset is provided by the author of [2], it is the dataset 1 in [2]. The dataset is recorded by a Pioneer 3 DX robot with a Project Tango, it provides  $640 \times 480$  grayscale images at 30 Hz, the inertial measurements at 100Hz, and wheel-encoder measurements at 10 Hz. In addition, the dataset also provides the ground truth that is computed from the batch least squares offline using all the available visual, inertial, and wheel measurements.

Qualitative comparison of the estimated trajectories by our method without and with loop closure is shown in Fig. 4. The estimated trajectories and the ground truth are aligned in closed form by using the method of Horn [30]. We can qualitatively compare the estimated trajectories with the result provided in figure 6 of the state-of-the-art visual-inertial-odometric SLAM system [2]. Firstly, by comparing the estimated trajectory from our algorithm without loop closure to the result in [2], we can know that our algorithm without loop closure produces more accurate trajectory estimate than method [2], which is achieved by 1) tightly fusing the visual and odometer measurements in optimization framework; 2) performing complete visual-odometric tracking strategies; 3) performing local BA that contains many covisibility information and inter-frame odometric constraints. Then by further comparison, it is clear that our algorithm with loop closure achieves better accuracy than our algorithm without loop

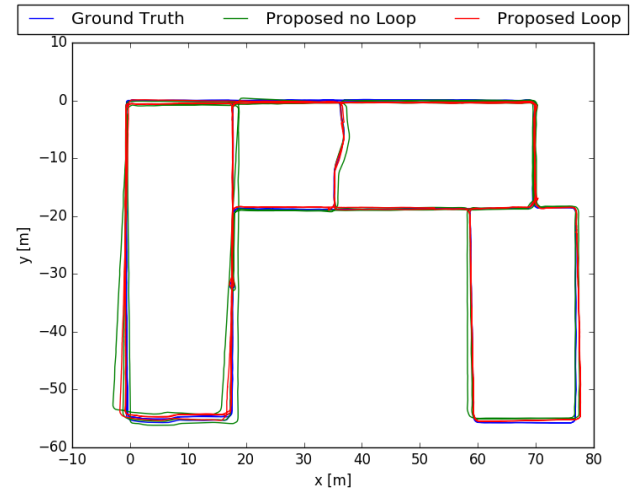


FIGURE 4: Comparison between the estimated trajectories and the ground truth.

closure, and thereby certainly achieves better accuracy than method [2], which is achieved by eliminating the accumulated error when returning to an already mapped area. Quantitatively, the sequence is 1080m long, and the positioning Root Mean Square Error (RMSE) of our algorithm without and with loop closure is 1.001m and 0.606m respectively, it is 0.093% and 0.056% of the total traveled distance with a comparison to 0.25% of the approach [2].

We also performed this sequence on the state-of-the-art and open-source monocular VSLAM system ORB-SLAM [14] and monocular visual-inertial SLAM system VINS-MONO [6] [26] for comparison. ORB-SLAM is the base of our system, which successfully ran the sequence and reached the translation RMSE of 2.787m, the translation RMSE is computed by scaling the estimated trajectory to match the scale of the ground truth. From the result, we can conclude that compared with the original monocular ORB-SLAM system, the proposed monocular VOSLAM system can not only recover the scale of the environment, but also achieve better accuracy. These improvements are achieved by 1) good scale initialization thanks to the reliable inter-frame odometer measurements; 2) tightly fusing the odometer factor with the visual factor of ORB-SLAM; 3) complete visual-odometric tracking strategies that maximally exploit both measurements. However, VINS-MONO method fails to run through the sequence. After the visual-inertial alignment succeeds, the trajectory starts to drift in a short period of time, which is due to the special motion of the dataset. In most cases, the motion of the dataset is along the straight lines with constant velocity, VISLAM has additional unobservable directions (i.e. the scale of the environment and direction of the local gravity) in the special motion, which makes the VINS-MONO unable to provide good initial value for the dataset. Whereas, initialization is the most fragile step for tightly-coupled monocular VISLAM, therefore in the dataset, the

bad initialization leads to the drift of the estimated trajectory.

### B. ALGORITHM EVALUATION IN THE RAWSEEDS DATASET

We also evaluate our algorithm on four indoor sequences of the Rawseeds dataset. The dataset provides  $320 \times 240$  images that are recorded by a forward-looking camera at the frequency of 30Hz, and provides inertial measurements and wheel-encoder measurements at 125Hz and 50Hz respectively. Besides, ground truth trajectory is also provided for all the sequences. Since there are frame losses at the latter part of each sequence, we only use the reliable first 15000 frames of each sequence for algorithm comparison.

TABLE 1: Translation RMSE of the estimated trajectories from ORB-SLAM, VINS-MONO, and our method on the rawseeds dataset.

Sequence	ORB-SLAM(m)	VINS-MONO(m)	our method(m)
2009-02-25a	X	6.825	0.661
2009-02-25b	X	3.074	1.418
2009-02-26a	X	4.137	1.558
2009-02-27a	X	9.032	2.057

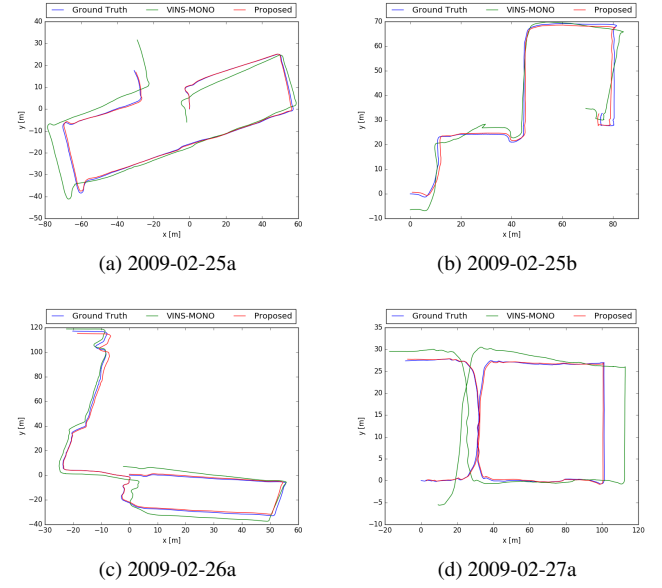


FIGURE 6: Comparison of the estimated trajectories from VINS-MONO and our algorithm with the ground truth on rawseeds dataset.

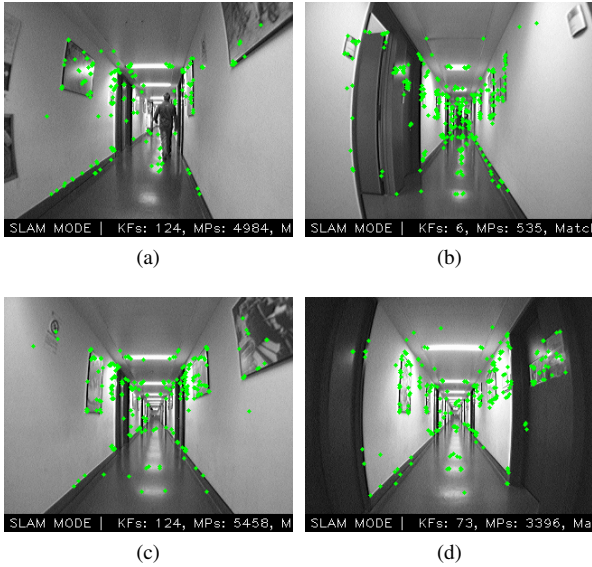


FIGURE 5: Tracked features in sample images of the rawseeds dataset.

We firstly compare the proposed algorithm with the monocular ORB-SLAM and VINS-MONO algorithm. In this section, we switch off the loop closure to perform fair comparison between the use of visual-odometer factor graph and the use of visual-inertial factor graph. A comparison of the translation RMSE for the estimated trajectories from different methods is shown in Table 1. X means that the corresponding method fails to run in the sequence. From the result, we can find that ORB-SLAM fails to run in the four sequences, which is because the sequences are relatively low-textured. In these sequences, most point features extracted

and matched are on the lines and white walls as shown in Fig. 5, therefore faulty data association leads to motion tracking failure. However, VINS-MONO can track the motion successfully in the four sequences. The sequences do not go as straight as the ds dataset, so the system can provide the good initial value that enables the system to successfully track the motion of subsequent frames. Qualitative comparison between VINS-MONO and our method is shown in Fig. 6. From the above qualitative and quantitative comparison, we can know that compared to VINS-MONO that uses the visual and inertial measurements, our method using visual and odometer measurements can achieve better accuracy. It is due to the observability of the two systems under the special motion interested in this paper, which is discussed above. Therefore, we can conclude that our system that tightly fuses the visual and odometer measurements is more suitable for estimating the motion of the ground robots.

Furthermore, we also compare the performance of the proposed factor graph with the factor graphs that arbitrarily combine the visual measurements with the wheel encoder, gyroscope, and planar measurements. In this comparison experiments, for fairly comparing each factor graph, we also switch off the loop closure. In the following, we abbreviate the proposed SE(3) odometer factor as O, SE(2) wheel encoder factor as W, SO(3) gyroscope factor as G, deterministic planar constraint as D, stochastic planar constraint as S, bg with computed initial value as C, bg with initial value of zero as N. The factor graphs used for comparing with our factor graph are respectively containing 1) WD, 2) WS, 3) WGDC, 4) WGSC, 5) ODC, 6) OSN, and our factor graph contains OSC. Qualitative and Quantitative comparison between the

TABLE 2: Translation RMSE of difference approaches on the rawseeds dataset.

Sequence	WD(m)	WS(m)	WGDC(m)	WGSC(m)	ODC(m)	OSN(m)	OSC(m)
2009-02-25a	2.343	0.779	1.718	0.793	0.910	1.145	0.661
2009-02-25b	1.691	2.235	1.879	1.615	1.603	2.293	1.418
2009-02-26a	3.323	1.831	3.527	1.790	1.731	1.945	1.558
2009-02-27a	2.806	2.676	2.587	2.445	2.408	2.583	2.057

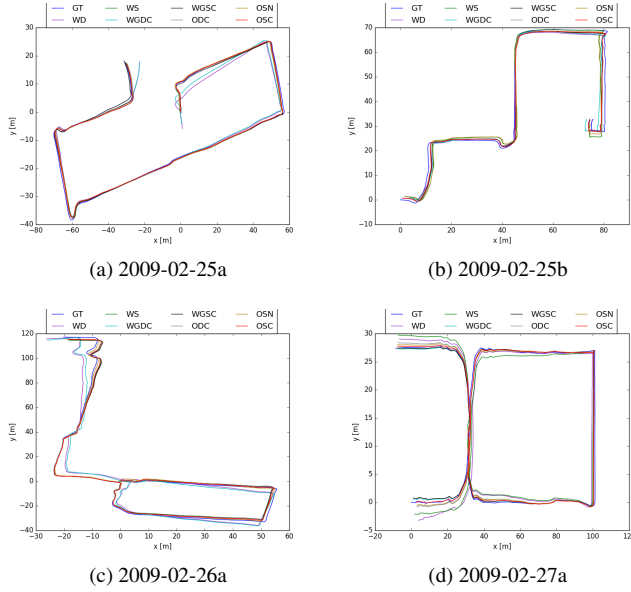


FIGURE 7: Comparison of the estimated trajectories from different methods on rawseeds dataset.

six factor graphs and our method are shown in Fig. 7 and Table 2.

From the result, we can know that our algorithm using the proposed OSC factor graph achieves best accuracy. Compared with the use of factor graphs that respectively combine W, WGC, and OC with deterministic planar constraint D, the use of factor graphs that respectively combine W, WGC, and OC with stochastic planar constraint S improves the translation RMSE of 26%, 32%, and 14% respectively. The approximately in-plane motion of ground robots caused by uneven terrains and vibrations of the moving platform is better modeled by stochastic planar constraint, thus the system using the stochastic planar constraint achieves better accuracy than using the deterministic one. Then by comparing the second and third columns of the table to the sixth and eighth columns of the table, we can know that combining visual factor with the proposed SE(3) odometer factor provides better accuracy than combining visual factor with the commonly used SE(2) wheel encoder factor. Besides, when combining the wheel encoder measurements and gyroscope measurements with the visual measurements, the factor graph using the proposed SE(3) odometer factor provides better accuracy than the factor graph using separate wheel encoder factor and gyroscope factor, which is demonstrated in the fifth and eighth columns

of the table. The advantage of the proposed SE(3) odometer factor is that compared to using the wheel encoder measurements and gyroscope measurements separately, tightly fusing these measurements can provide more accurate inter-frame rotational and translational constraint for both in-plane and out-of-plane motion. Finally by comparing the seventh and eighth columns of the table, we can know that computing initial value for gyroscope bias can achieve better accuracy, it is since computing initial gyroscope bias can provide better rotational and thereby better translational constraint. The superiority of the proposed factor graph for localizing the ground robot will be more obvious when the motion of robot is often out of the plane constraint.

### C. DEMONSTRATION OF ROBUSTNESS TO WHEEL SLIPPAGE

In the following experiments, we use data recorded from a DIY robot with a OV7251 camera mounted on it to look upward for visual sensing. The sensor suite provides the  $640 \times 480$  grayscale images at the frequency of 30Hz, the wheel-odometer and gyroscope measurements at 50 Hz. Since there is no ground truth available, we just perform qualitative analysis.

We firstly let the ground robot to walk normally, then hold the robot to make it static but the wheel is spinning, and finally let it to normally walk once again. The estimated results in some critical moments are shown in Fig. 8 and Fig. 9. Fig. 8a is the captured image at the first critical moment when the platform start to experience wheel slippage, and the trajectories estimated by our method and the odometer from the beginning to this moment are shown in Fig. 9a. We can see that both methods can accurately estimate the position of the sensor suite under normal motion. The image and the estimated trajectory obtained at second moment when wheel slippage is over are given in Fig. 8b and Fig. 9b. As evident, the images captured at first and second critical moments are almost the same. Although the odometer provides far away poses for these two moments due to the wheel slippage, our method still gives the very close poses for these two moments. Thus the validity of the proposed strategy for detecting and solving wheel slippage can be proved. The reconstructed 3D map for the sequence are shown in Fig. 8c, the map is globally consistent, which is achieved by effectively solving the problem of wheel slippage.

Strategy for detecting and solving the wheel slippage can also be used to solve situations where the platform is moved artificially. For validation, we perform the experiment as follows. The sensor suite walks normally at first, then the





FIGURE 8: Captured images when the platform begins to experience wheel slippage and wheel slippage is over, and the finally reconstructed 3D map at the end of the sequence.

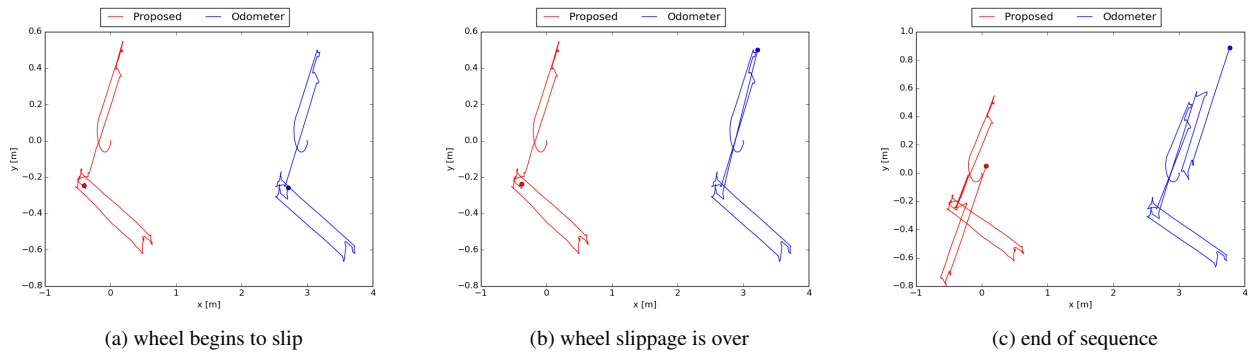


FIGURE 9: The estimated trajectories from beginning to some critical moments.



FIGURE 10: Captured images when the platform begins to be moved and the platform has been moved to another location, and the finally reconstructed 3D map at the end of the sequence.

wheels turn normally, however the platform is moved to another location artificially, and finally it normally walks once again. The test results for the situation are shown in Fig. 10 and Fig. 11. Before the sensor is moved away, the estimated trajectories from both our method and the odometer are close to each other as shown in Fig. 11a. Fig. 10a and Fig. 10b are the captured images at first moment when platform starts to move and at second moment when the platform has been moved to another location. As shown in Fig. 11b, for the artificial movement, the proposed method provides precise

motion tracking with comparison to the faulty estimation of the odometer. Thereby, the effectiveness of the proposed strategy for detecting and solving faulty wheel measurements is demonstrated again.

#### D. DEMONSTRATION OF ROBUSTNESS TO VISUAL LOSS

The robustness of our system to visual loss is tested in two sequences, sequence 1 includes the case 1 and case 3 described in Section V-C and the sequence 2 includes the



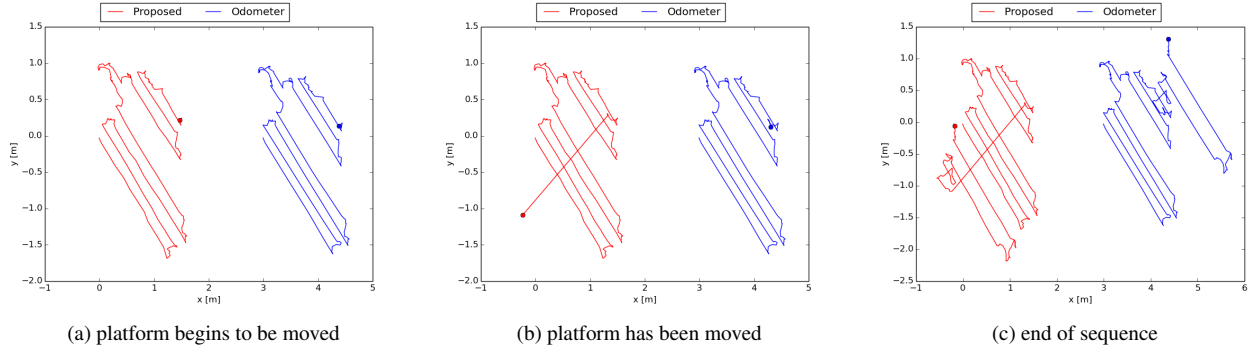


FIGURE 11: The estimated trajectories from beginning to some critical moments.

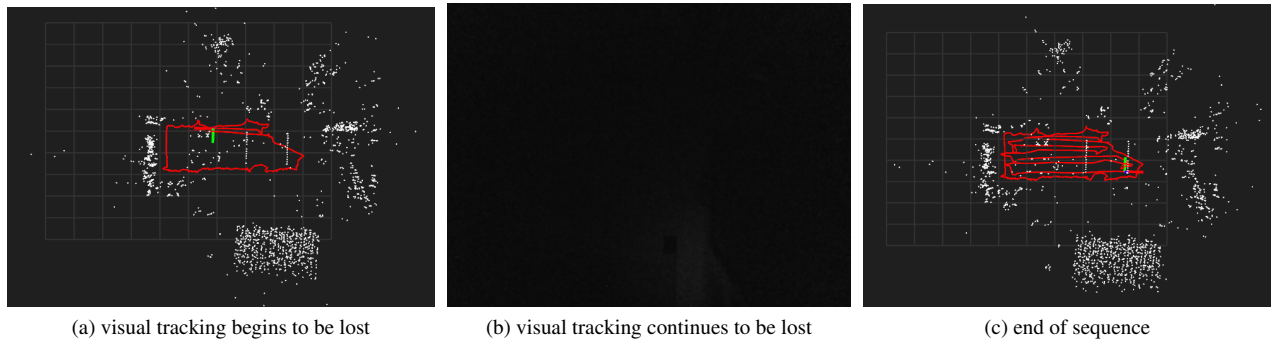


FIGURE 12: Reconstructed 3D map when the visual tracking begins to be lost and at the end of the sequence, and captured image when visual tracking is lost.

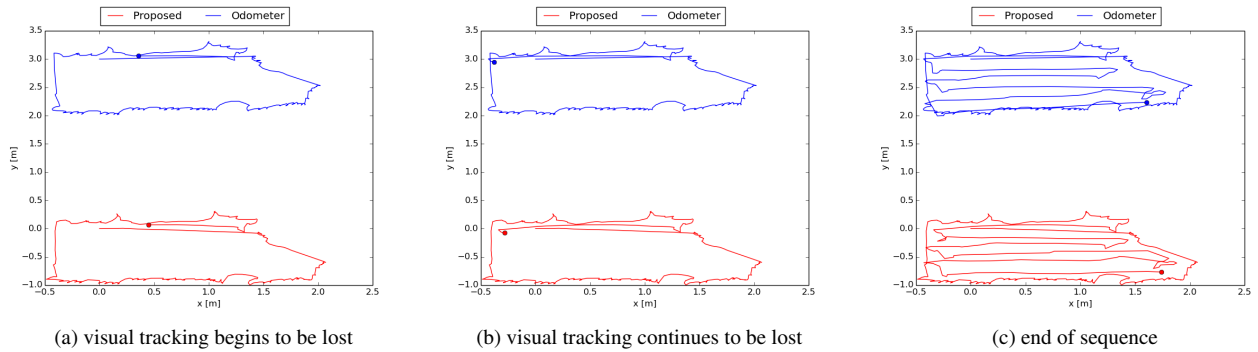


FIGURE 13: The estimated trajectories from beginning to some critical moments.

case 2 and case 3 described in Section V-C. Firstly, we use sequence 1 to test the proposed solution for the case 1 and case 3, the estimated results in some critical moments are shown in Fig. 12 and Fig. 13. The robot firstly moves on areas where enough visual information is available to build a map of the environment shown in Fig. 12a. Then we turn off the lights to make the visual information unavailable. The motion of robot is continuously computed in the period of visual loss as shown in Fig. 13b, which is achieved by using the odometer measurements as solution to case 3. Finally, we turn on the lights to make the robot revisiting an already

reconstructed area. As shown in accompanying video, at that moment, the visual information is recovered by performing the global relocalization. The reconstructed map at the end of sequence is shown in Fig. 12c, it is globally consistent without closing the loop. Therefore, we can validate the effectiveness of the proposed solution for case 1 and case 3.

Secondly, we use sequence 2 to test the proposed solution for case 2, the test results are shown in Fig. 14. The ground robot firstly moves on areas where enough visual information is available to build the map of the environment shown in Fig. 14a. Then the robot goes to the low texture environment,

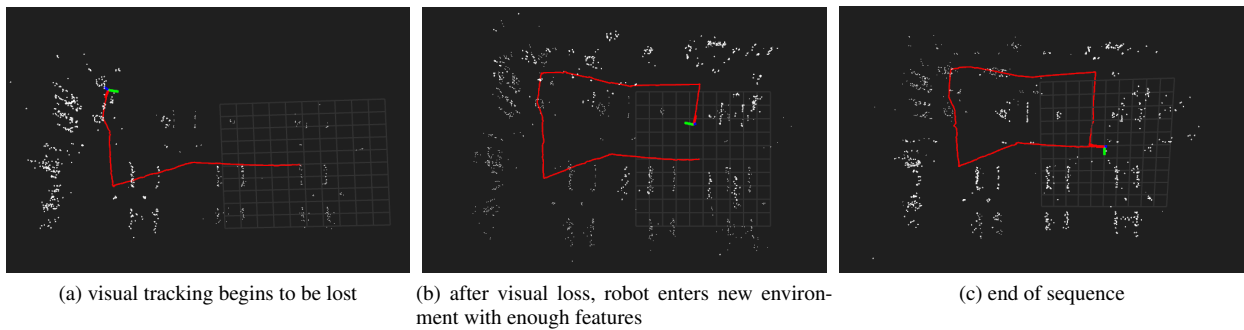


FIGURE 14: Reconstructed 3D map when visual tracking begins to be lost, when after visual loss the robot enters new environment with enough features, and at the end of the sequence.

and later enters new environment where enough features are available. From Fig. 14b, we can know that new map is created when enough feature points exist in new environment, however the new map is not consistent with the previously reconstructed map. Finally, the robot returns to an already mapped area, which triggers loop closure to eliminate the accumulated error, thereby constructing the globally consistent map as shown in Fig. 14c. From the experiment, we can conclude that our system can not only tightly fuse both measurements to ensure the accuracy of the system, but also can improve the robustness of the system to visual loss by using stable measurements from the odometer.

## VII. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a tightly-coupled monocular VOSLAM system. The whole system is bootstrapped by the proposed map initialization method. Then when both visual and odometer measurements are available, the system tightly integrates the proposed odometer factor and visual factor in the optimization framework to ensure the accuracy of the system. The system can also detect and solve the faulty information from wheel encoders to avoid the false estimates caused by wrong measurements. Then when the visual information is not available, the system tries to recover the visual information as soon as possible and improves the robustness of the system to visual loss by using the reliable odometer measurements. By carrying out thorough experiments, we have demonstrated that our system can provide accurate, robust, and long-term localization for the wheeled robots moving on a plane.

In future work, we aim to exploit line features to improve the performance of our algorithm in environments where only fewer point features are available. In addition, we will add full IMU measurements to our system for improving the accuracy of system and dealing with the situation where both visual and wheel measurements cannot provide valid measurements for localization.

## ACKNOWLEDGMENT

We thank the author of [2] for providing the dataset to evaluate our algorithm in Section VI-A.

## REFERENCES

- [1] H. Lategahn, A. Geiger, and B. Kitt. Visual slam for autonomous ground vehicles. In *IEEE International Conference on Robotics and Automation*, pages 1732–1737, May 2011.
- [2] K. J. Wu, C. X. Guo, G. Georgiou, and S. I. Roumeliotis. Vins on wheels. In *IEEE International Conference on Robotics and Automation*, pages 5155–5162, May 2017.
- [3] A. I. Mourikis and S. I. Roumeliotis. A multi-state constraint kalman filter for vision-aided inertial navigation. In *IEEE International Conference on Robotics and Automation*, pages 3565–3572, April 2007.
- [4] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart. Keyframe-based visual-inertial odometry using nonlinear optimization. *International Journal of Robotics Research*, 34(3):314–334, 2014.
- [5] R. Mur-Artal and J. D. Tardos. Visual-inertial monocular slam with map reuse. *IEEE Robotics and Automation Letters*, 2(2):796–803, 2017.
- [6] P. Li, T. Qin, B. Hu, F. Zhu, and S. Shen. Monocular visual-inertial state estimation for mobile augmented reality. In *IEEE International Symposium on Mixed and Augmented Reality*, pages 11–21, Oct 2017.
- [7] J. Michot, A. Bartoli, and F. Gaspard. Bi-objective bundle adjustment with application to multi-sensor slam. In *International Symposium on 3D Data Processing, Visualization and Transmission*, pages 1–8, 2010.
- [8] A. Eudes, M. Lhuillier, S. Naudet-Collette, and M. Dhome. Fast odometry integration in local bundle adjustment-based visual slam. In *International Conference on Pattern Recognition*, pages 290–293, Aug 2010.
- [9] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza. Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. In *Robotics: Science and Systems*, Rome, Italy, July 2015.
- [10] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.
- [11] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis. A first-estimates jacobian ekf for improving slam consistency. In *International Symposium on Experimental Robotics*, pages 373–382, Athens, Greece, July 2008.
- [12] J. A. Hesch and S. I. Roumeliotis. Consistency analysis and improvement for single-camera localization. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 15–22, June 2012.
- [13] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 225–234, Nov 2007.
- [14] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.
- [15] D. Galvez-Lopez and J. D. Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5):1188–1197, 2012.

- [16] H. Strasdat, J. M. M. Montiel, and A. J. Davison. Scale drift-aware large scale monocular slam. In *Robotics: Science and Systems*, Zaragoza, Spain, June 2010.
- [17] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *IEEE International Conference on Computer Vision*, pages 2320–2327, Nov 2011.
- [18] C. Forster, M. Pizzoli, and D. Scaramuzza. Svo: Fast semi-direct monocular visual odometry. In *IEEE International Conference on Robotics and Automation*, pages 15–22, June 2014.
- [19] J. Engel, T. Schops, and D. Cremers. Lsd-slam: Large-scale direct monocular slam. In *European Conference on Computer Vision*, pages 834–849, Sep 2014.
- [20] J. Engel, V. Koltun, and D. Cremers. Direct sparse odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3):611–625, 2018.
- [21] R. Mur-Artal and J. D. Tardos. Probabilistic semi-dense mapping from highly accurate feature-based monocular slam. In *Robotics: Science and Systems*, Rome, Italy, July 2015.
- [22] P. Pinies, T. Lupton, S. Sukkariéh, and J. D. Tardos. Inertial aiding of inverse depth slam using a monocular camera. In *IEEE International Conference on Robotics and Automation*, pages 2797–2802, April 2007.
- [23] M. Kleinert and S. Schleith. Inertial aided monocular slam for gps-denied navigation. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 20–25, Sep 2010.
- [24] E. Jones and S. Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *International Journal of Robotics Research*, 30(4):407–430, 2011.
- [25] M. Li and A. I. Mourikis. High-precision, consistent ekf-based visual-inertial odometry. *International Journal of Robotics Research*, 32(6):690–711, 2013.
- [26] T. Qin, P. Li, and S. Shen. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018.
- [27] J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel. 1-point ransac for ekf filtering. application to real-time structure from motion and visual odometry. *Journal of Field Robotics*, 27(5):609–631, 2010.
- [28] D. Nister. An efficient solution to the five-point relative pose problem. *IEEE transactions on pattern analysis and machine intelligence*, 26(6):756–770, 2004.
- [29] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnnp: An accurate o(n) solution to the pnp problem. *International Journal of Computer Vision*, 81(2):155–166, 2009.
- [30] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.

...