doi:10.20944/preprints202111.0103.v1

Article

An Improved Initialization Method for Monocular Visual-Inertial SLAM

Cheng Jun [1], Zhang Liyan [1,*] and Chen Qihong [1]

[1] School of Automation, Wuhan university of technology, No.122, Luoshi Road, Wuhan, Hubei, 430070, P.R.China

* Correspondence: zlywhut@whut.edu.cn

Abstract: In the aim of improving the positioning accuracy of monocular visual inertial simultaneous localization and mapping (VI-SLAM) system, an improved initialization method with faster convergence is proposed. This approach is classified as three parts: Firstly, in the initial stage, the pure vision measurement model of ORB-SLAM is employed to make all the variables visible. Secondly, the frequency of IMU camera was aligned by IMU preintegration technology. Thirdly, an improved iterative method is put forward for estimating the initial parameters of IMU faster. The estimation of IMU initial parameters is divided into several simpler sub-problems, containing direction refinement gravity estimation, gyroscope deviation estimation, accelerometer bias and scale estimation. The experimental results on the self-built robot platform show that our method can upregulate the initialization convergence speed, simultaneously improve the positioning accuracy of the entire VI-SLAM system.

Keywords: VI-SLAM; Initialization; Localization; optimization

1. Introduction

Visual Simultaneous Localization and Mapping (VSLAM) techniques allow mobile robots [1,2] and VR/AR devices [3,4] to be aware of their surrounding scene, while carries on the self-localization in the unknown environments. The monocular vision inertial system of SLAM containing strap down inertial measurement units (IMU) and monocular vision sensor to provide a low-cost, lightweight, and high-quality solution for most positioning and navigation applications in indoor and outdoor environment. For simultaneous interpreting of multiple sensor measurements from various sensor frames, a process of initial parameters estimation and calibration is essential. The camera only needs to be calibrated once because it does not change over time, and the IMU sensor must be initialized prior to each use. So this paper focuses on the IMU's initial values estimation. The IMU initialization process is designed to evaluate as fast as possible to the initial parameters with the initial IMU biases (gyroscope and accelerometer biases), gravity and scale for the process of later numerical optimization. Once the parameters are triumphantly acquired, inertial measurement can be employed to enhance the robustness and accuracy of the continuous tracking and then find the measurement scale of three-dimensional visual map, which can not be obtained with a pure monocular SLAM system. Currently, the tightly-coupled nonlinear optimization approach for visual-inertial SLAM are widely applied, almost the state-of-the-art frameworks, for instance OKVIS [5], VI-ORBSLAM [6], VI-DSO [7] and VINS-MONO [8] can not have a good performance without an efficient initialization process. Especially, the convergence speed of initial parameters has the significant affect the performance of the whole system.

Generally, the initialization of monocular VI-SLAM system is a fragile but a significant step. The former visual inertia initialization methods can be divided into joint methods together with non-joint methods [9].

The joint visual-inertial initialization approach is introduced through Martinelli at first, which named closed-form solution. However, this research [10] expressed only in theory and then demonstrated by the simulation of general Gaussian motions, the application of MAV is not feasible. So this method is later modified in [11], not only increases the estimation of gyroscope bias, but also this is a successful implementation of actual data from quadrotor MAV. The latest work of [12] put forward a robust and fast initialization approach according to [10,11]. The accuracy is improved through several visual-inertial bundle adjustment (BA), and the robustness of system is enhanced with the addition of consensus and observability tests. As it is tested on the dataset of Euroc [13], it is proved to be consistently initialized with the scale errors is less than five percent. However, those initialization methods have several limitations:

- An ideal hypothesis which all features are tracked in perspective should be contented. However, it can lead to bad solutions under conditions of the spurious tracks.
- Compared with [14], the disjoint visual-inertial initialization method, the accuracy of joint method is lower. To improve it, a lot of frames and tracks are usually added, which lead to computational cost is so high that the real-time performance is unfeasible.
- The method in [12] works only at 20% of trajectory points. If the system requires to be started immediately, this may be a problem in robot use.

The disjoint visual-inertial initialization approach, i.e. loosely couple method, which depends on a very accurate visual measurement model in initial stage. This method is first applied by Mur-Artal and latter adapted in [8,15] with a good performance on the public dataset. In particular, the motion of MAV with metric scale can be recovered with a small error, and the accuracy of positioning is maintained at centimeter level [6]. However, this approach also exists several limitations:

- The process of initial estimation is slow and instable. On account of the inertial parameters are evaluted through solving a set of the linear equations in various steps utilizing the least square method, it requires an excellent iterative strategy makes fast convergence. However, the convergence speed in [6] is not reliable enough for all variables estimation. it can be a problem for many real applications.
- Initialization is fragile. As the method requires to run monocular visual SLAM in advance for finding the accurate inertial parameters. If the visual part gets lost, the inertial system will not be launched immediately.

In summary, there are several initialization methods have been studied for monocular VI-SLAM system. However, few researchers have tried to improve it from the perspective of non-linear optimization. In the current work, an improved initialization approach which is in accordance with the disjoint method is proposed. First, in the initial stage, the pure vision measurement model of ORB-SLAM2 is employed to make all the variables visible. Second, the frequency of IMU camera was aligned by IMU pre-integration technology [16]. Third, the IMU initialization process, which is highlighted in dotted block diagram with red color. It is divided into several simpler sub-problems, containing direction refinement gravity estimation, gyroscope deviation estimation, scale estimation as well as accelerometer deviation. In this work, an improved iterative method is put forward for estimating the initial parameters of IMU faster. The experimental outcomes on real mobile robot demonstrate the excellent performance while our initialization method is integrated into the VI-SLAM system which is on the basis of ORB-SLAM2 skeleton [14,18].

The rest of the current paper is organized as below: We introduce the preparatory work in Section 2. Then the core part of this paper, IMU initialization process, is illustrated in Section 3. The Section 4 introduces the real-time experiment for the mobile robot. Section 5 gives the summaries and the future work.

2. Preliminaries

In the present section, the monocular visual-inertial coordinate frames and visual measurement model are briefly reviewed, then the IMU pre-integration on manifold are described in the follow sections. Our goal is to accurate estimate b_g , g_w , b_a , s , namely, gyroscope bias, gravity, accelerometer bias together with visual scale in the visual-inertial initialization stage, the correlations between camera frame {C} and IMU main frame {B} is defined by scale factor s is considered:

$$\begin{aligned} R_{WB} &= R_{WC} \cdot R_{CB} \\ {}_W P_B &= R_{WC} \cdot {}_C P_B + s \cdot {}_W P_C \end{aligned} \quad (1)$$

in which $\boldsymbol{P}$ and $\boldsymbol{R}$ respectively is the translation and rotation vector.

2.1. Coordinate frames

The transformation between the coordinate frames are shown in figure 1. Since the measurements of inertial and visual odometry are the relative movement. However, the absolute pose is needed in the pre-fixed reference frame. Therefore, it is assumed that the reference frame of our system coincides with the first key frame which is determined by the pure visual SLAM. In this work, the transformation $\boldsymbol{T}_{CB}$ between body frame {B} and camera frame {C} is calibrated in advance. The ${}_E\boldsymbol{G}$ represents the gravity in the inertial frame {E} of earth. The ${}_w\boldsymbol{g}$ represents the gravity in world copedinate system {W}. The first key frame is assumed as reference frame.

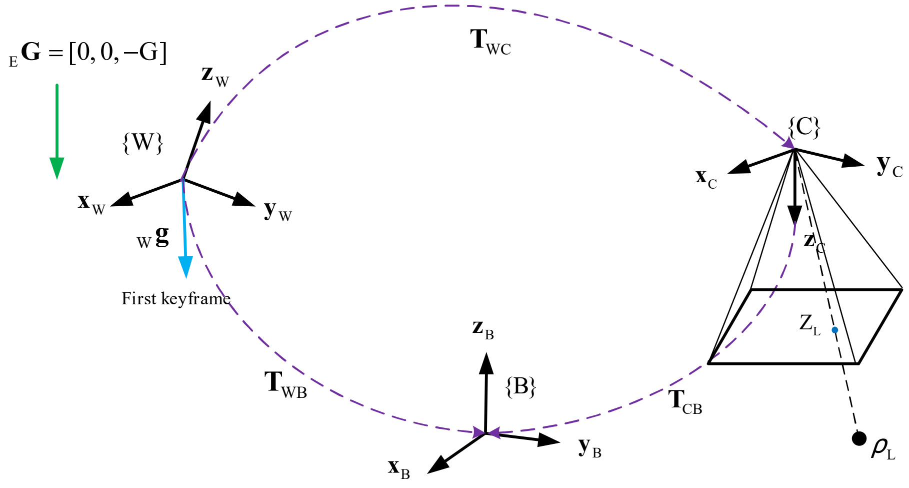


Figure 1. Coordinate frames of monocular VI-SLAM system.

2.2. Visual measurement model

The ORB-SLAM2 [14,18] visual measurement model is adopted for the initial pose estimation. The system consists of the following three parallel threads, i.e., tracing, local mapping along with loop closing. While the tracking together with local mapping threads are applied in the initialization stage. The tracking part is responsible for deciding whether to treat the new frame as a key. After inserting the new key, the associated IMU pre-integration model is calculated between two consecutive key frames. In this work, we adopt the conventional visual model with the visual projection function π: $\mathbb{R}R^3 \Leftrightarrow R\mathbb{R}^2$ which converts 3-dimensional points $X_C = \mathbb{R}^3$ in the camera frame {C} into a 2-dimensional image coordinate $X_C = \mathbb{R}^2$.

$$\pi(X_C) = \begin{bmatrix} f_u \dfrac{x_c}{z_c} + c_u \\ f_v \dfrac{x_c}{z_c} + c_v \end{bmatrix}; \quad X_C = [x_c, y_c, z_c]^T \tag{2}$$

in which c and f respectively is the principal point and focal length.

2.3. IMU Pre-integration

Since the measurements of IMU and camera output with different rates, the IMU pre-integration technology for aligning the frequency of IMU camera is introduced. The concept of IMU pre-integrated is pioneered in [17] and extended in [16] on the manifold space. Assumed that there are two consecutive key frames at time *j* and *i*, the associated IMU position ${}_W P_B$, velocity ${}_W v_B$ and orientation R_{WB} can be calculated through summarizing all of the measurements during this period:

$$\begin{aligned} \boldsymbol{R}_{WB}^{j} &= \boldsymbol{R}_{WB}^{i} \prod_{k=i}^{j-1} Exp((\boldsymbol{w}_B^k - \boldsymbol{b}_g^k - \boldsymbol{\eta}_g^k)\Delta t) \\ {}_W\boldsymbol{v}_B^j &= {}_W\boldsymbol{v}_B^i + {}_W g \Delta t_{ij} + \sum_{k=i}^{j-1} \boldsymbol{R}_{WB}^k (\boldsymbol{a}_B^k - \boldsymbol{b}_a^k - \boldsymbol{\eta}_a^k)\Delta t \\ {}_W\boldsymbol{P}_B^j &= {}_W\boldsymbol{P}_B^i + \sum ({}_W\boldsymbol{v}_B^k \Delta t + \frac{1}{2}\, {}_W\boldsymbol{g}\Delta t^2 + \frac{1}{2}\boldsymbol{R}_{WB}^k (\boldsymbol{a}_B^k - \boldsymbol{b}_a^k - \boldsymbol{\eta}_a^k)\Delta t^2) \end{aligned} \tag{3}$$

in which Δt is the sampling interval of IMU, with $\Delta t_{ij} = (j-i)\Delta t$. The $Exp(.)$ represents an exponential mapping operator that maps Lie algebra $so(3)$ to the Lie group $SO(3)$. When it is assumed that the deviation remains unchanged in the course of pre-integration, and the effect of measurement noise of IMU is ignored, a small correction $\delta b_{(.)}^i$ of the formerly estimated $\bar{b}_{(.)}^i$ could be considered to correct pre-integrated outcomes. We can rewrite the expressions in the equation (3) as below:

$$\begin{aligned} \boldsymbol{R}_{WB}^{j} &= \boldsymbol{R}_{WB}^{i} \Delta \overline{\boldsymbol{R}}_{ij} Exp(J_{\Delta R_{ij}}^g \boldsymbol{\delta} b_g^i) \\ {}_W\boldsymbol{v}_B^j &= {}_W\boldsymbol{v}_B^i + {}_W\boldsymbol{g}\Delta t_{ij} + \boldsymbol{R}_{WB}^i (\Delta v_{ij} + J_{\Delta v_{ij}}^g \boldsymbol{\delta} b_g^i + J_{\Delta v_{ij}}^a \boldsymbol{\delta} b_a^i) \\ {}_W\boldsymbol{P}_B^j &= {}_W\boldsymbol{P}_B^i + {}_W\boldsymbol{v}_B^i \Delta t_{ij} + \frac{1}{2}\, {}_W\boldsymbol{g}\Delta t_{ij}^2 + \boldsymbol{R}_{WB}^i (\Delta \overline{P}_{ij} + J_{\Delta p_{ij}}^g \boldsymbol{\delta} b_g^i + J_{\Delta P_{ij}}^a \boldsymbol{\delta} b_a^i) \end{aligned} \tag{4}$$

among them, the Jacobians $J_{(.)}^g$ and $J_{(.)}^a$ express how the measured value change owing to the change of deviation estimation. The biases $\bar{b}_g^i$ and $\bar{b}_a^i$ remain constant in the course of pre-integration and can be pre-calculated at time i. The specific Jacobians calculation is shown in [16]. Subsequently, the $\Delta \bar{R}_{ij}$, $\Delta \bar{v}_{ij}$ and $\Delta \bar{p}_{ij}$ pre-integration values can be directly calculated from the outputs of IMU between two keyframes, which are independent of the gravity and the states at time *i*:

$$\begin{aligned} \Delta \overline{\boldsymbol{R}}_{ij} &= \prod_{k=i}^{j-1} Exp((w_B^k - \bar{b}_g^i)\Delta t) \\ \Delta \bar{\boldsymbol{v}}_{ij} &= \sum_{k=i}^{j-1} \Delta \overline{\boldsymbol{R}}_{ik} (a_B^k - \bar{b}_a^i) \\ \Delta \overline{\boldsymbol{P}}_{ij} &= \sum_{k=i}^{j-1} (\Delta \bar{\boldsymbol{v}}_{ik} \Delta t + \frac{1}{2} \Delta \overline{\boldsymbol{R}}_{ik} (a_B^k - \bar{b}_a^i)\Delta t^2) \end{aligned} \tag{5}$$

3. IMU initialization

In the present section, the initial IMU parameters are estimated, containing gravity g_w, gyroscope bias b_g, visual scale s and accelerometer bias b_a. With the aim of making all the variables visible, the pure monocular visual SLAM system requires to work for a few seconds and then wait for the several key frames to be formed (Sec. 2.2). The specific process of the estimation of IMU parameters is revealed as below.

doi:10.20944/preprints202111.0103.v1

3.1. Gyroscope bias estimation

From the known direction of two consecutive keyframes, we can estimate the gyro bias. It is assumed that the variation of the deviation is negligible, that is, the bias b_g is a constant value, this constant value minimizes the difference between the relative direction calculated via ORB-SLAM2 and the gyro integral for all pairs of continuous keyframes:

$$\arg\min_{b_g} \sum_{i=1}^{N-1} \| Log(\Delta \boldsymbol{R}_{i,i+1} Exp(\boldsymbol{J}_{\Delta R}^{g} \boldsymbol{b}_g))^T \boldsymbol{R}_{BW}^{i+1} \boldsymbol{R}_{WB}^{i} \|^2 \tag{6}$$

in which N represents the key frames number. $\boldsymbol{R}_{WB}^{(.)} = \boldsymbol{R}_{WC}^{(.)} \cdot \boldsymbol{R}_{CB}$ is calculated from the calibration R_{CB} and orientation $\boldsymbol{R}_{WC}^{(.)}$. $\Delta \boldsymbol{R}_{i,i+1}$ denotes the gyro integration between the two consecutive keyframes. $Exp(.)$ and $J_{\Delta R}^{g}$ respectively represents the exponential mapping $R^3 \to SO3$ together with Jacobian matrix. The analytic Jacobian matrices of similar expression is exhibited in [16].

3.2. Gravity direction estimation

Because the gravity direction possesses a great effect on the acceleration estimation, the direction of gravity must be refined prior to estimating the accelerometer bias, gravity and scale parameters. Particularly, a new constraint, gravity magnitude $G(G \approx 9.8)$, is introduced. As revealed in figure 2. The inertial reference frame is defined as {I} and world frame is defined as {W}, the gravity direction is defined as $\overline{\boldsymbol{g}}_I = \{0,0,1\}$. According to frame {W}, the direction of gravity can be calculated as follows:

$$\overline{\boldsymbol{g}}_w = \boldsymbol{g}_W^* / \| \boldsymbol{g}_W^* \| \tag{7}$$

from the angle θ between two direction vectors, we can calculate rotation R_{WI}:

$$\boldsymbol{R}_{WI} = Exp(\overline{\boldsymbol{v}} \boldsymbol{\theta}) \tag{8}$$

with $\overline{\boldsymbol{v}} = \frac{\overline{\boldsymbol{g}}_I \times \overline{\boldsymbol{g}}_W}{\| \overline{\boldsymbol{g}}_I \times \overline{\boldsymbol{g}}_W \|}, \boldsymbol{\theta} = a\tan 2(\| \overline{\boldsymbol{g}}_I \times \overline{\boldsymbol{g}}_W \|, \overline{\boldsymbol{g}}_I \cdot \overline{\boldsymbol{g}}_W)$, thus the gravity vector can be described as below:

$$\boldsymbol{g}_W = \boldsymbol{R}_{WI} \overline{g}_I G \tag{9}$$

in which R_{WI} can be parametrized, only two angles around axis x and y are used in frame {I}, and the rotation around axis z has no influence in g_W.

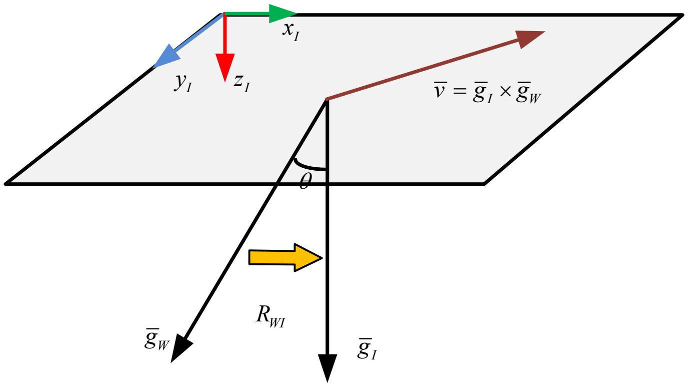


Figure 2. The refinement of gravity direction

3.3. Improved iterative strategy

Since the equation (6) is a classical problem of nonlinear least square. The generally used solution approach is gauss-newton (G-N) algorithm, as revealed in table 1 (algorithm 1), which is adopted in [14]. However, this method has several drawbacks. First, large iteration increment may result in slow convergence. Second, this algorithm requires the H (Hessian matrix) is positive definite and invertible while the actual calculated data may not meet this requirement.

Table 1. Gauss-Newton iterative strategy

```
Algorithm 1
1. Set the initial value $x_0$ ;
2. for the k-th iteration, calculate the Jacobian matrix $J(x_k)$ and the error function $f(x_k)$ ;
3. solve the incremental equation:
       $\boldsymbol{H} \cdot \Delta x_k = \boldsymbol{g}$
4. If the increment $\Delta x_k$ is small enough, iteration stops;
   else $x_{k+1} = x_k + \Delta x_k$, go back to step 2.
```

In this paper, an improved iterative method is proposed for improving the stability of convergence. In particular, an appropriate trust region μ is added to the increment Δx. In the process of each iteration, it is assumed to be effective when the increment Δx is located in the trust region. Otherwise, it is considered to be invalid and the iteration may not be converged. The improved iteration method is displayed in table 2 (algorithm 2).

Table 2. Improved iterative strategy

```
Algorithm 2
1. Set the initial $x_0$ and the radius of trust region $\mu_0$ .
2. Solve the optimal problem:
       $\min_{\Delta x} \frac{1}{2} \| f(x) + \boldsymbol{J}(x)\Delta x \|^2, \quad s.t. \| D\Delta x \|^2 \le \mu$
3. Calculate $\rho$ :
       $\rho = \frac{f(x+\Delta x) - f(x)}{\boldsymbol{J}(x)\Delta x}$
4. Update $\mu$ , if $\rho > 0.75$
           $\mu = 2\mu$
       else if $\rho < 0.25$
           $\mu = 0.5\mu$.
5. If met the iteration termination condition, i.e.
       $\|g\|_\infty \le \eta_1$ or $\|\Delta x\| \le \eta_2 (\| x \| + \eta_2)$
           or $k \ge k_{MAX}$
   then iteration stops;
   if not met, then $x \leftarrow x + \Delta x$, go back to step 2.
```

According to formula in step 2:

$$\min_{\Delta x} \frac{1}{2} \| f(x) + \boldsymbol{J}(x)\Delta x \|^2, \qquad s.t. \| \boldsymbol{D}\Delta x \|^2 \le \mu \tag{10}$$

We add the constraint: $\| \boldsymbol{D}\Delta x \|^2 \le \mu$, where μ and D respectively is the radius of the trust region and scaling matrix. When D is unit matrix I or not (for example, D is diagonal matrix), the trust region is a sphere with radius μ or ellipsoid). To facilitate calculation, Lagrange multiplier is utilized to convert formula (10) into the unconstrained optimization problem:

$$\begin{gathered} \min_{\Delta x} \frac{1}{2} \| f(x) + \boldsymbol{J}(x)\Delta x \|^2, \qquad s.t. \| \boldsymbol{D}\Delta x \|^2 \le \mu \\ \rightarrow \min_{\Delta x} \frac{1}{2} \| f(x) + \boldsymbol{J}(x)\Delta x \|^2 + \frac{\lambda}{2} \| \boldsymbol{D}\Delta x \|^2 \end{gathered} \tag{11}$$

here λ denotes the Lagrange multiplier, through the expansion of formula, a linear equation can be acquired to count the increment:

$$(\boldsymbol{H} + \lambda \boldsymbol{I}) \cdot \Delta x = \text{-}\boldsymbol{g} \tag{12}$$

with $\boldsymbol{H} = \boldsymbol{J}^T \boldsymbol{J}$, $\boldsymbol{g} = \boldsymbol{J}^T \cdot f$ and $\lambda \ge 0$.

where $\boldsymbol{J} = \boldsymbol{J}(x)$ and $f = f(x)$. Formula (12) can be considered as a steepest descent algorithm when λ is small. With the aim of effectively adjusting the range of trust region, the ratio between the approximate model and actual function after each iteration was calculated in step 3, as below:

$$\rho = \frac{f(x+\Delta x) - f(x)}{\boldsymbol{J}(x)\cdot\Delta x} \tag{13}$$

in which the $\{f(x+\Delta x) - f(x)\}$ and $\{J(x).\Delta x\}$ respectively is the actual function together with approximate model. When ρ is close to 1, it indicates that the approximation performance is good. If ρ < the threshold set to be $\rho < 0.25$, it represents that in contrast to approximate reduction, the actual reduction is much smaller, so it is necessary to reduce the trust region radius and set it to $\mu = 0.5\mu$. If ρ is greater than the threshold set to $\rho > 0.75$, it is necessary to expand the trust region radius set to $\mu = 2\mu$.

In step 5, there exist two stop criteria. At first, the stopping criteria of algorithm should meet the following criteria:

$$\|g\|_{\infty} \le \eta_1 \tag{14}$$

here η_1 is the small value, set to $\eta_1 = 10^{-6}$, $\|.\|_{\infty}$ represents an infinite norm.

Secondly, when the increment Δx is too small, we should consider stopping the iteration:

$$\|\Delta x\| \le \eta_2(\|x\| + \eta_2) \tag{15}$$

in which η_2 represents the relative step size, set to $\eta_2 = 10^{-6}$.

Ultimately, we also set up a protection measure to prevent infinite loops that limit the maximum number of the iterations $k_{MAX} = 2000$,when $k \ge k_{MAX}$, the iteration will be forced to stop.

3.4. Accelerometer bias and scale estimation

In accordance with the former sections (section 3.1, 3.2 and 3.3). Once the accurate gravity vector and gyro bias is acquired, the equation (4) is applied for the pre-integration of positions and velocities, rotate the measurement of acceleration correctly to compensate for the gyro deviation. Subsequently, in consideration of the effect resulted from the accelerometer deviation, the R_{WI} is also adjusted, which can be described via a two degree of the freedom disturbance $\delta\theta$, the equation (9) can be rewritten as below:

$$\begin{aligned} \boldsymbol{g}_W &= \boldsymbol{R}_{WI}Exp(\delta\theta)\overline{g}_I G \approx \boldsymbol{R}_{WI}\overline{g}_I G + \boldsymbol{R}_{WI}(\delta\theta)^{\wedge}\overline{g}_I G \\ &= \boldsymbol{R}_{WI}\overline{g}_I G - \boldsymbol{R}_{WI}(\overline{g}_I)^{\wedge} G\delta\theta \end{aligned} \tag{16}$$

With $\delta\theta = [\delta\theta_{xy}^T, 0]^T, \delta\theta_{xy} = [\delta\theta_x, \delta\theta_y]^T$.

Therefore, containing the influence of the accelerometer bias, we can get:

$$\begin{aligned} s\,{}_W\boldsymbol{p}_C^{i+1} = s\,{}_W\boldsymbol{p}_C^{i} + {}_W\boldsymbol{v}_B^{i}\Delta t_{i,i+1} - \frac{1}{2}\boldsymbol{R}_{WI}(\overline{g}_I)\times G\Delta t_{i,i+1}^2 \delta\theta \\ + \boldsymbol{R}_{WB}^{\mathrm{i}}(\Delta p_{i,i+1} + J_{\Delta p}^{a} b_a) + (\boldsymbol{R}_{WC}^{i} - \boldsymbol{R}_{WC}^{i+1})\,{}_C\boldsymbol{p}_B + \frac{1}{2}\boldsymbol{R}_{WI}\overline{g}_I G\Delta t_{i,i+1}^2 \end{aligned} \tag{17}$$

In consideration of the constraints among the three consecutive keyframes, the velocities can be eliminated and the linear relationship is get as follows:

$$[\lambda(i) \quad \varphi(i) \quad \zeta(i)]\begin{bmatrix} s \\ \delta\theta_{xy} \\ b_a \end{bmatrix} = \psi(i) \tag{18}$$

here $\lambda_{(i)}$, $\varphi(i)$, $\zeta(i)$, and $\psi(i)$ are calculated as below:

doi:10.20944/preprints202111.0103.v1

$$\begin{aligned}
\lambda(i) &= ({}_W\boldsymbol{p}_C^2 - {}_W\boldsymbol{p}_C^1)\Delta t_{23} - ({}_W\boldsymbol{p}_C^3 - {}_W\boldsymbol{p}_C^2)\Delta t_{12} \\
\varphi(\mathrm{i}) &= \left[\frac{1}{2}\boldsymbol{R}_{WI}(\overline{g}_I)\times G(\Delta t_{12}^2\Delta t_{23} + \Delta t_{23}^2\Delta t_{12})\right]_{(:,1:2)} \\
\zeta(\mathrm{i}) &= \boldsymbol{R}_{WB}^2 J_{\Delta p23}^a \Delta t_{12} + \boldsymbol{R}_{WB}^1 J_{\Delta v23}^a \Delta t_{12}\Delta t_{23} - \boldsymbol{R}_{WB}^1 J_{\Delta p12}^a \Delta t_{23} \\
\psi(\mathrm{i}) &= (\boldsymbol{R}_{WC}^2 - \boldsymbol{R}_{WC}^1)\,{}_C\boldsymbol{p}_B\Delta t_{23} - (\boldsymbol{R}_{WC}^3 - \boldsymbol{R}_{WC}^2)\,{}_C\boldsymbol{p}_B\Delta t_{12} \\
&\quad + \boldsymbol{R}_{WB}^2\Delta p_{23}\Delta t_{12} + \boldsymbol{R}_{WB}^1\Delta v_{12}\Delta t_{12}\Delta t_{23} - \boldsymbol{R}_{WB}^1\Delta p_{12}\Delta t_{23} + \frac{1}{2}\boldsymbol{R}_{WI}\,\overline{g}_I G\Delta t_{ij}^2
\end{aligned} \tag{19}$$

in which $[]_{(:,1:2)}$ denotes the top two columns of matrix. By superimposing all the correlations between three consecutive key frames (18), the linear system can generate the following equations $A_{3(N-2)\times 6}X_{6\times 1} = B_{3(N-2)\times 1}$, which can be solved through the method of singular value decomposition (SVD). In this condition, it is composed of six unknown variables and 3(N - 2) equations, and at least four key frames are required to solve the system.

4. Experiments

The initialization method is applied in unknown indoor environment with self-build mobile robot platform. The platform structure is exhibited in figure 3, the major components include a low cost VI-camera (MYNT S1030-IR-120), a NVIDIA Jetson TX2, a Xsens MTI-300 and two 12V DC batteries for power supply.

The key parameters of MYNT S1030-IR-120 are shown in table 3, it communicates with NVIDIA Jetson TX2 through USB 3.0 interface. In terms of the Xsens MTI-300, it outputs the high frequency measurements of accelerometers and gyroscopes. In this work, we treat it as reference system through the post-processing operation. As the low-cost equipment is used to collect datasets, the frequency of IMU sensor is set to 150Hz, while the frequency of camera is set to 10Hz. All of the experiments are implemented by utilizing the computer with i7-9700 CPU (8 cores @3.00GHz) and 16GB RAM in the Ubuntu 18.04+Melodic operating system. The internal and external parameters of the IMU and camera are calibrated via Kalibr tool [19] in advance, and the camera-IMU external parameters are assumed as constant values.

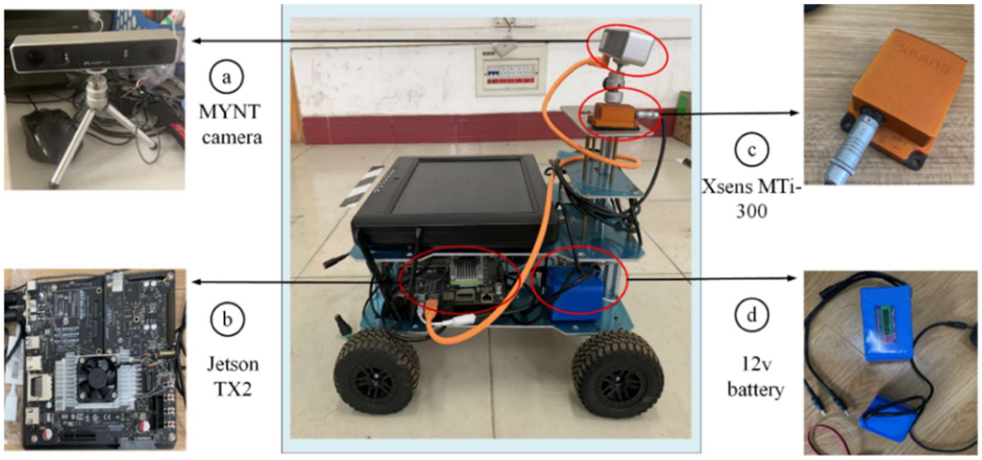


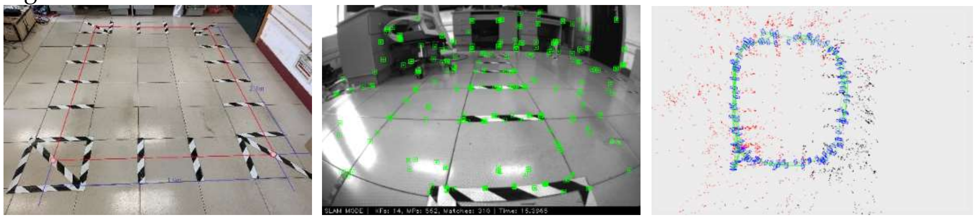
Figure 3. Mobile Robot Platform. (a) MYNT binocular camera with two global shutter cameras. (b) NVIDIA Jetson TX2 as an onboard computation resource, (c) Xsens MTi-300 as reference system. (d) Batteries with 12V DC power.

Table 3. Parameters of MYNT camera

Version	S1030-IR-120
Size	165x31.5x31.23mm
Weight	184g
Frames per Second	10-60FPS
Resolution	752*480;376*240
FHD	6.0*6.0um
Base line	120.0mm
Focal length	2.1mm
Power dissipation	1~2.7W @ 5v DC
IMU frequency	100-500Hz
Exposure mode	Global shutter
Measuring Depth	0.8-5m+
Interface	USB 3.0

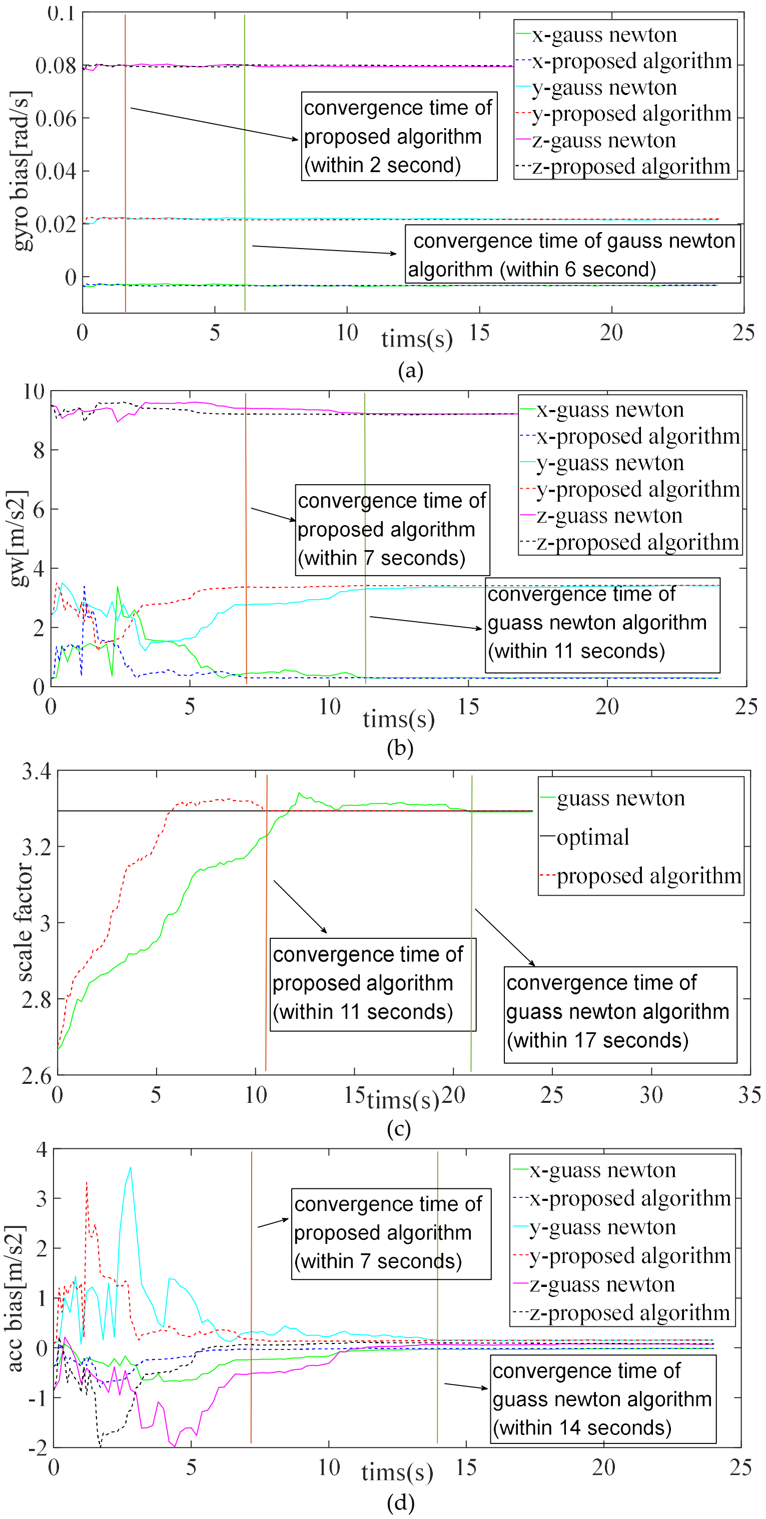
4.1. Evaluation of the initial estimation

Our initialization method is first integrated into VI-SLAM system. In order to evaluate the algorithms fairly, the original algorithm with gauss-newton algorithm (Mur-Artal et al., 2017) and the proposed algorithms are detected on a same data set which the mobile robot is controlled to perform several close-loop movements in indoor environment. Beside, we only utilized left camera image to test the performance of monocular VI-SLAM system. Figure 4(a)-(c) shows the example image frame from the laboratory dataset. The comparison results of the initial parameter estimation are shown in figure 5(a)-(d), it can be known that all estimated variables, containing gravity, gyro deviation, scale factor and accelerometer deviation are converged to the stable values within 2 to 11 seconds by using the proposed algorithm (dotted lines), while the gauss newton algorithm (solid lines) is converged within 6 to 17 second. In particular, as exhibited in figure. 5 (a), within 2 seconds, the gyro bias in x, y, z directions converges to - 0.019, 0.023, and 0.081. It is well demonstrated that the iterative method acquired better performance. In figures 5 (b) and 5 (d), the characteristic curves of accelerometer deviation and gravity oscillate seriously within five seconds. This is owing to the mobile robot platform does not show enough excitation to the sensor kit in the slight disturbance and stationary stages, making it difficult to distinguish between gravity vector and accelerometer bias, but the proposed algorithm still has a good performance in convergence speed. In figure 5(c), the visual scale factor is converged 10 seconds later, and gauss newton algorithm is converged after 17 second. In general, in the convergence speed, the algorithm is faster than Gauss Newton algorithm.



(a) (b) (c)

Figure 4. Laboratory scenes and experimental results. (a) the laboratory scene with about 2.4m long and 1.6m wide in indoor environment. (b) the feature-based front end of system, which the ORB feature locations shown in green color. (c) the trajectory of key frames with 3D point cloud map.



(a)

(b)

(c)

(d)

Figure 5. Time varying characteristic curves for initial estimations. (a) the gyroscope bias estimation, "gyro" denotes "gyroscope". (b) the gravity estimation, "gw" denotes "gravity". (c) the estimation of visual scale factor. (d) the estimation of accelerometer bias, "acc" denotes "accelerometer". The dotted line denotes the algorithm utilizing improved iterative method, and the solid lines is the outcomes of algorithm based on guass newton which is employed in VI-ORBSLAM system. The convergence time are represented via red and green vertical lines, respectively.

doi:10.20944/preprints202111.0103.v1

4.2. Evaluation of the tracking accuracy

In the present section, the property of this algorithm on the VI-SLAM system accuracy was assessed. Similar to the public dataset experiment, when our algorithm is tested on the self-collected dataset, the visual-inertial odometry is utilized as attitude and position feedback. The trajectories are aligned with the reference trajectory, i.e. the measurements of Xsens MTI-300. As exhibited in figure 6, the dotted line denotes the ground truth trajectories, and the green line and red line respectively represent the trajectories of the gauss-newton based algorithm (i.e. VI-ORBSLAM) and our proposed algorithm, respectively. It can be known that the trajectories can be tracked completely by them, but the two algorithms have different degrees of deviation. Due to the improved initialization process, the trajectory of ours is closer to the ground truth compared with VI-ORBSLAM.

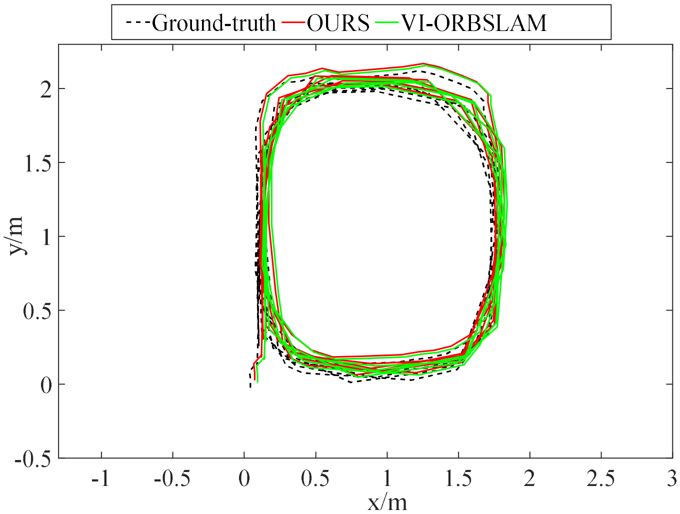


Figure 6. Trajectories comparison with VI-ORBSLAM. The 2D trajectories, VI-ORBSLAM (green line), Our system (red line) and ground-truth (dotted line). VI-ORBSLAM adopts gauss newton algorithm, while our system adopts the proposed algorithm.

The quantitative evaluation results are obtained through the calculation of equation (20) and (21). Which the RMSE errors are calculated as follows:

1) RMSE error of position

$$\begin{cases} \boldsymbol{RMSE}_{pos\_x} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\overline{x}_{pos(k)} - x_{pos(k)})^2} \\ \boldsymbol{RMSE}_{pos\_y} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\overline{y}_{pos(k)} - y_{pos(k)})^2} \\ \boldsymbol{RMSE}_{pos\_z} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\overline{z}_{pos(k)} - z_{pos(k)})^2} \end{cases} \tag{20}$$

where, $(\overline{x}_{pos(k)}, \overline{y}_{pos(k)}, \overline{z}_{pos(k)})$ denote the estimation of position with x, y, z axis, $(x_{pos(k)}, y_{pos(k)}, z_{pos(k)})$ denote the true position with x, y and z axis, respectively.

2) RMSE errors of orientation

$$\begin{cases} \boldsymbol{RMSE}_{ori\_x} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\overline{x}_{ori(k)} - x_{ori(k)})^2} \\ \boldsymbol{RMSE}_{ori\_y} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\overline{y}_{ori(k)} - y_{ori(k)})^2} \\ \boldsymbol{RMSE}_{ori\_z} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(\overline{z}_{ori(k)} - z_{ori(k)})^2} \end{cases} \tag{21}$$

where, $(\overline{x}_{ori(k)}, \overline{y}_{ori(k)}, \overline{z}_{ori(k)})$ denote the estimation of orientation with x, y, z axis, $(x_{ori(k)}, y_{ori(k)}, z_{ori(k)})$ denote the true orientation with x, y and z axis, respectively.

As shown in table 4, the reported value is the median after 10 times of each test. Bold type represents the optimal result. The RMSE errors of position in terms of the VI-ORB-SLAM and our proposed algorithm are [0.150, 0.125, 0.133] (m) and [0.091, 0.115, 0.123] (m). Which the position accuracy is increased by 39.3%, 8%, 7.5% along x axis, y axis, and z axi, respectively. The RMSE errors of orientation are [1.356, 1.165, 1.987] and [1.032, 1.134, 1.857] (°). Which the orientation accuracy is increased by 23.9%, 2.7%, 6.5% along x axis, y axis, and z axis, respectively. Obviously, the improvement of position and orientation accuracy in x axis is most significant. It also well confirms that the proposed initialization method pocesses a positive role on the positioning accuracy of the monocular VI-SLAM system.

Table 4. Quantitative RMSE evaluation results of original algorithm and our algorithm.

	VI-ORBSLAM		OURS	
	Pos(m)	Ori (°)	Pos(m)	Ori(°)
X	0.150	1.356	**0.091**	**1.032**
Y	0.125	1.165	**0.115**	**1.134**
Z	0.133	1.987	**0.123**	**1.857**

5. Conclusions and future work

In the present work, we put forward a new initialization algorithm for monocular VI-SLAM system from the perspective of non-linear optimization. Thanks to an improved iterative strategy, our initialization procedure provides high quality initial seeds which contain gravity vector, gyroscope bias, visual scale as well as accelerometer biases. Besides, a real world dataset collected by self-build mobile robots to validate the proposal. The results demonstrate that this algorithm has excellent property in system positioning accuracy and initial parameter convergence speed than the gauss-newton based algorithm (VI-ORBSLAM) in indoor environment. A limitation of this strategy is the camera-IMU external parameters are assumed as constant values. Actually, the external parameters have uncertain influence on the initialization results. In the future works, we will make additional online estimation of external parameters in initialization stage to improve the property of system.

6. Patents

Supplementary Materials: Not applicable.

Author Contributions: Conceptualization, Zhang Liyan.; methodology and writing—original draft preparation Cheng Jun.; supervision, Chen Qihong. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported partially through the National Key Research and Development Program of China (2019YFB1504703).

Data Availability Statement: The data presented in this study are available on request from the corresponding author

Conflicts of Interest: The authors declare that there exist no conflict of interests in publishing this paper.

References

1. Lin, Y., Gao, F., Qin, T., et al. (2018) "Autonomous aerial navigation using monocular visual-inertial fusion." Journal of Field Robotics. Vol. 35, No. 1, pp. 23-51.
2. Bloesch, M., Omari, S., Hutter, M., and Siegwart, R. (2015) "Robust visual inertial odometry using a direct EKF-based approach." IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, pp. 298–304.
3. Taragay, O., Supun, S., and Rakesh, K. (2012) "Multi-sensor navigation algorithm using monocular camera, IMU and GPS for large scale augmented reality." Proc. IEEE Int. Symp. Mixed Augmented Reality, pp. 71–80.
4. Li, P., Qin, T., Hu, B., Zhu, F. and Shen, S. (2017) "Monocular visual-inertial state estimation for mobile augmented reality." IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 11–21.
5. Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R. and Furgale, P. (2015), "Keyframe-based visual-inertial odometry using nonlinear optimization", The International Journal of Robotics Research, Vol. 34, No. 3, pp. 314-334.
6. Murartal, R. and Tardos, J. D. (2017), "Visual-Inertial Monocular SLAM With Map Reuse", IEEE Robotics and Automation Letters, Vol. 2, No. 2, pp. 796-803.
7. Von Stumberg, L., Usenko, V., and Cremers, D. (2018), "Direct Sparse Visual-Inertial Odometry Using Dynamic Marginalization," IEEE International Conference on Robotics and Automation, Brisbane, pp. 2510-2517.
8. Qin, T., Li, P. and Shen, S. (2018), "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator". IEEE Transactions on Robotics, Vol. 34, No. 4, pp. 1004-1020.
9. Campos, C., Montiel, J. M., and Tardos, J. D. (2020). "Inertial-Only Optimization for Visual-Inertial Initialization", arXiv: Robotics.
10. Martinelli, A. (2014). "Closed-form solution of visual-inertial structure from motion." International Journal of Computer Vision, vol. 106, no. 2, pp. 138–152.
11. Kaiser, J., Martinelli, A., Fontana, F., and Scaramuzza, D. (2016). "Simultaneous state initialization and gyroscope bias calibration in visual inertial aided navigation," IEEE Robotics and Automation Letters, vol. 2, no. 1, pp. 18–25.
12. Campos, C, Montiel, J. M. M., and Tardos, Juan D. (2019). "Fast and Robust Initialization for Visual-Inertial SLAM." IEEE International Conference on Robotics and Automation, Montreal, pp. 1288-1294.
13. Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., et al. (2016). "The EuRoC micro aerial vehicle datasets." The International Journal of Robotics Research, vol. 35, no. 10, pp.1157–1163.
14. Murartal, R. and Tardos, J. D. (2017), "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras". IEEE Transactions on Robotics, Vol. 33, No. 5, pp. 1255-1262.
15. Qin, T, and Shen, S.(2017),"Robust initialization of monocular visual-inertial estimation on aerial robots." IEEE/RSJ International Conference on Intelligent Robots & Systems IEEE, Vancouver, pp. 4225-4232.
16. Forster, C., Carlone, L., Dellaert, F. and Scaramuzza, D.(2015), "IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation", robotics science and systems, Rome, pp. 1-20.
17. Lupton, T. and Sukkarieh, S.(2012), "Visual-Inertial-Aided Navigation for High-Dynamic Motion in Built Environments Without Initial Conditions," IEEE Transactions on Robotics, Vol. 28, No. 1, pp. 61-76.
18. Murartal, R, Montiel, J. M. M., and Tardos, J. D. (2015). "ORB-SLAM: A Versatile and Accurate Monocular SLAM System." IEEE Transactions on Robotics, vol. 31, no.5, pp. 1147-1163.
19. Rehder, J., Nikolic, J., Schneider, T., Hinzmann, T., Siegwart, R. (2016). "Extending kalibr: Calibrating the extrinsics of multiple IMUs and of individual axes." IEEE International Conference on Robotics and Automation, Stockholm, pp. 4304-4311.