sensors

MDPI

Article

A Robust Indoor/Outdoor Navigation Filter Fusing Data from Vision and Magneto-Inertial Measurement Unit

David Caruso [1,*], Alexandre Eudes [2], Martial Sanfourche [2], David Vissière [1] and Guy Le Besnerais [2]

[1] Computer Vision R and D department, Sysnav, 57 rue de Montigny, 27200 Vernon, France; david.vissiere@sysnav.fr
[2] Department of Information Processing and Systems, ONERA, the French Aerospace Lab, Chemin de la Hunière, 91120 Palaiseau, France; alexandres.eudes@onera.fr (A.E.); martial.sanfourche@onera.fr (M.S.); guy.le_besnerais@onera.fr (G.L.B.)
* Correspondence: david.caruso@polytechnique.edu

Received: 31 October 2017; Accepted: 30 November 2017; Published: 4 December 2017

Abstract: Visual-inertial Navigation Systems (VINS) are nowadays used for robotic or augmented reality applications. They aim to compute the motion of the robot or the pedestrian in an environment that is unknown and does not have specific localization infrastructure. Because of the low quality of inertial sensors that can be used reasonably for these two applications, state of the art VINS rely heavily on the visual information to correct at high frequency the drift of inertial sensors integration. These methods struggle when environment does not provide usable visual features, such than in low-light of texture-less areas. In the last few years, some work have been focused on using an array of magnetometers to exploit opportunistic stationary magnetic disturbances available indoor in order to deduce a velocity. This led to Magneto-inertial Dead-reckoning (MI-DR) systems that show interesting performance in their nominal conditions, even if they can be defeated when the local magnetic gradient is too low, for example outdoor. We propose in this work to fuse the information from a monocular camera with the MI-DR technique to increase the robustness of both traditional VINS and MI-DR itself. We use an inverse square root filter inspired by the MSCKF algorithm and describe its structure thoroughly in this paper. We show navigation results on a real dataset captured by a sensor fusing a commercial-grade camera with our custom MIMU (Magneto-inertial Measurment Unit) sensor. The fused estimate demonstrates higher robustness compared to pure VINS estimate, specially in areas where vision is non informative. These results could ultimately increase the working domain of mobile augmented reality systems.

Keywords: Visual-inertial Navigation Systems (VINS); Magneto-inertial Dead reckoning (MI-DR); vision-based navigation; IMU navigation; MSCKF; error-state kalman filter

1. Introduction

1.1. Motivation

Infrastructure-less navigation and positioning in indoor location is a technical prerequisite for numerous industrial and consumer applications: ranging from lone worker safety in industrial facilities, to augmented reality. Still, it remains an open challenge to efficiently and reliably combine embedded sensors to reconstruct a position or a trajectory. In the present work, we address this challenge restricting ourselves to the following sensors: MEMS gyroscopes, accelerometers, magnetometers and a standard industrial vision camera. We motivate this choice by the fact these sensors are cheap and can easily be embedded in a wearable form factor, which makes this combination appealing for pedestrian application. Moreover, VINS (Visual-Inertial Navigation Systems) literature, showed recently tremendous progress in the past few years.

1.2. State of the Art and Contribution

Indeed, if a wide range of embedded visual sensors were previously presented to solve the problem, such as rotating LIDARs [1] or depth sensors [2], much of the recent efforts focused on conventional cameras, as they are cheap, and already present in a wide range of lightweight devices, such as smart-phones. While authors of [3–5] rely on multiple embedded cameras, single cameras solutions have been shown to provide good results. For instance, authors of [6,7] present efficient filtering methods for monocular VINS. Nonetheless, monocular VINS remains highly sensitive to degenerate motions scenarios. For instance, the scale factor is weakly constrained in case of a steady motion and pure rotation motions must be tackled with special care in the estimation process. Moreover, current VINS implementations rely heavily on high frequency visual corrections (10–20 Hz), and break when visual environment is not adequate for more than a few second (bad illumination, presence of smoke, motion blur, etc.).

Compared to the previous literature, we explore a sensor suite alternative to VINS and able to provide a precise and robust navigation information. We combine the vision sensor with a MIMU (Magneto-Inertial Measurement Unit), i.e., an IMU sensor augmented with *an array of* magnetometers. Within a stationary and non-uniform magnetic environment, the magnetic measurements render the body speed observable, which has been shown to significantly improve motion prediction compared to IMU alone [8,9]. This technique, named Magneto-Inertial Dead-Reckoning (MI-DR) hereafter, is perfectly fitted for indoor navigation as human made environment, wall, floor, and furnitures all perturb the magnetic field in a significant way. As a result, the open-loop position error of this method is often around a few percent of the trajectory length in indoor environment (see [10]). MI-DR fails however in places where the gradient of the magnetic field vanishes, (commonly outdoor) and lacks robustness when the magnetic environment is not stationary.

We have already presented various approaches to fuse the information from MIMU sensor with vision sensors for dead-reckoning estimation. In [11], we proposed a semi-tight fusion scheme combining a depth sensor with a MIMU to increase robustness and availability of the position/orientation informations. Yet, we finally turned towards conventional monocular camera, mainly because the limited range of depth sensors makes them unable to improve the MI-DR estimation in large rooms or outdoor. In order to still be able to estimate accurately the scale, we also turned towards a fully tight fusion scheme, that includes estimation of camera pose, current speed, magnetic field and inertial sensor biases. We investigated an optimization-based solution in [12] and a filter-based solution in [13]. Here we present an extension of the work presented in the conference paper [13] with new results and a slightly different implementation of the filter. The used estimator is still inspired by the pure VINS method presented in [14] for the visual measurements and by the magnetic prediction and measurement process of [9] for the MIMU handling process. We will call it MI-MSCKF for Magneto-inertial Multi State Constraint Kalman Filter. As in [13], we choose a square-root implementation which leads to a better overall conditioning of matrices operations involved in the filtering process. The inverse form is also computationally interesting for high-dimensionality measurements [15] (p. 141).

We show on experimental data that the MIMU provides robustness in situations where vision fails, and that, reciprocally, vision allows the system to handle situations where the magnetic gradient is to low for the MIMU to work properly, typically in outdoor situations.

1.3. Paper Organization

After introducing general notations and conventions in Section 2, we describe the dynamic model our filter is built on in Section 3, with a focus on the magnetic prediction equation, less known in visual-inertial literature. This section also presents the model discretization and error model of the sensors. The Section 4 describes the filter design: chosen state and equations for propagation and measurement update steps. The last section (Section 5) presents comparative results of trajectory estimation obtained on real datasets.

2. Notations

2.1. General Conventions

Bold capital letters $\mathbf{X}$ denote matrices or elements of manifold. Parenthesis are used to denote the Cartesian product of two elements $a \in \mathcal{A}, b \in \mathcal{B} \mapsto (a, b) \in \mathcal{A} \times \mathcal{B}$ and brackets for the concatenation of two row vectors. For a vector $\mathbf{x} = [x_1, x_2, x_3]^\top$, $\mathbf{x}_2$ denotes its second component x_2 and $\mathbf{x}_{2:3}$ is the sub-vector $[x_2, x_3]^\top$. For matrices, we define the vectorization operation $\mathrm{Vec}\,()$ so that:

$$\mathrm{Vec}\left(\begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}\right) = \begin{bmatrix} x_{11}, x_{21}, x_{12}, x_{22} \end{bmatrix}^\top. \tag{1}$$

$\mathrm{A} \otimes \mathrm{B}$ will denotes the Kronecker product of matrices A and B. ∂_{A} will be a shorthand for the derivative with respect to the coefficients of $\mathrm{Vec}\,(\mathrm{A})$. The notation $\|\mathbf{x}\|_{\mathbf{\Sigma}}$ is the Mahalanobis norm of invertible covariance $\mathbf{\Sigma}$: $\|\mathbf{x}\|_{\mathbf{\Sigma}} = \mathbf{x}^\top \mathbf{\Sigma}^{-1} \mathbf{x}$. $\mathbf{I}_n$ is the identity matrix of size n and $\mathbf{0}_{n \times m}$ the zero matrix of size $n \times m$. $\mathbf{I}_n$ is the identity matrix of size $n \times n$ or corresponding application. $\mathcal{O}(n)$ is the orthogonal matrix group of size n.

2.2. Reserved Symbols

Generally and except stated otherwise we use the following symbols: $\mathbf{p}$ for the translational part of the body pose, $\mathtt{R}$ for its rotational part. $\mathbf{v}$ for the velocity, $\mathbf{b}_a$ and $\mathbf{b}_g$ for inertial sensor bias, $\mathbf{B}$ for the magnetic field, $\nabla\mathbf{B}$ for its 3×3 gradient matrix. $\boldsymbol{\omega}$ is used for the rotational speed and $\mathbf{a}$ for the specific acceleration. 3D landmark position are noted with the letter $\mathbf{l}$ and these observation into an image with the letter $\boldsymbol{o}$. We use a tilde symbol for measured quantities or generally quantities that can be derived from sensor reading. We use an hat for estimated version of a physical quantity. $\mathbf{g}$ symbol is kept for the gravity vector in inertial coordinates. The world coordinates are defined such that the gravity vector writes $\mathbf{g} \simeq [0, 0, -9.81]^\top$. When ambiguous, the reference frame in which a quantity is expressed will be noted in exponent: w stands for the world gravity-aligned reference frame, b for the current body reference frame and c for the camera frame. The Figure 1 summarizes the chosen notations.

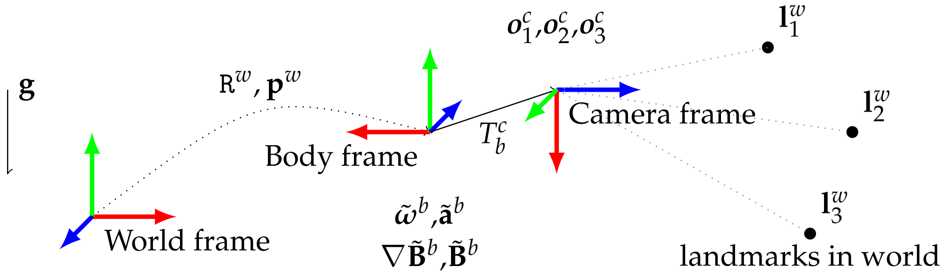

$\boldsymbol{o}_1^c, \boldsymbol{o}_2^c, \boldsymbol{o}_3^c$
$\mathbf{l}_1^w$
$\mathbf{g}$
$\mathtt{R}^w, \mathbf{p}^w$
Camera frame
$\mathbf{l}_2^w$
T_b^c
Body frame
$\mathbf{l}_3^w$
$\tilde{\boldsymbol{\omega}}^b, \tilde{\mathbf{a}}^b$
World frame
$\nabla\tilde{\mathbf{B}}^b, \tilde{\mathbf{B}}^b$
landmarks in world

Figure 1. Reference coordinate frames at play in the problem, with associated typical measurements.

2.3. Rotation Parametrization

For rotations, we use the convention that $\mathtt{R}^w$ transforms a vector from body frame to world frame by left multiplying it. For the sake of clarity of the developments, we represent the attitude of the sensor as a rotation matrix. The Special Orthogonal Group is denoted SO(3) and its associated Lie algebra $\mathfrak{so}(3)$—the set of skew symmetric matrices. Any element of $\mathfrak{so}(3)$ can be identified with a vector of $\mathbb{R}^3$: $[x]_\times \in \mathfrak{so}(3)$ with $x \in \mathbb{R}^3$ and vex so that $vex([x]_\times) = x$. exp and log are the standard exponential map and logarithm on SO(3). As in [16], we use vectorized versions of exp and log:

$$\text{Exp} : \begin{array}{ccc} \mathbb{R}^3 & \to & \text{SO}(3) \\ \boldsymbol{\delta\theta} & \mapsto & \exp([\boldsymbol{\delta\theta}]_\times) \end{array} \tag{2}$$

and Log : SO(3) $\to \mathbb{R}^3$ the inverse function. With these conventions, $\text{Log}(\text{Exp}(x)) = x$.

3. On-Board Sensors and Evolution Model

3.1. Sensing Hardware

The MIMU sensors (see Figure 2) provide raw measurements of biased proper acceleration $\tilde{\mathbf{a}}^b$, biased angular velocity $\tilde{\omega}^b$, magnetic field $\tilde{\mathbf{B}}^b$ and its gradients $\nabla\tilde{\mathbf{B}}^b$. The latter is a 3 × 3 matrix which elements are estimated by finite differences between signals recorded on an array of magnetometers. These sensors are carefully calibrated offline and harmonised in the same spatial coordinate frame with a method similar to [17]. The resulting MIMU coordinate frame, centered around the magnetometer and accelerometer (which are assumed colocalized here), will be used as the *body* frame in all subsequent derivations. We use a global shutter camera modeled as a pinhole camera with instantaneous exposure. In practice, recorded real images are undistorted using intrinsics calibration parameters. Intrinsic parameters (focal length in pixels (f_x, f_y), principal point coordinates (c_x, c_y) and distortion coefficients) are assumed to be known from a preliminary calibration. The pinhole camera projection function π maps a landmark 3D coordinates $\mathbf{l}^c$ expressed in the camera frame to the pixel coordinates of its projection onto the image:

$$\pi : \begin{array}{ccc} \mathbb{R}^3 & \to & \mathbb{R}^2 \\ \mathbf{l}^c & \mapsto & \begin{bmatrix} f_x \frac{\mathbf{l}^c_1}{\mathbf{l}^c_3} + c_x \\ f_y \frac{\mathbf{l}^c_2}{\mathbf{l}^c_3} + c_y \end{bmatrix} \end{array} . \tag{3}$$

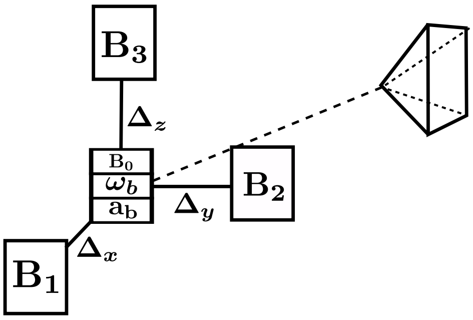


Figure 2. Schematic view of on-board sensors. In addition to accelerometers and gyrometers, the MIMU includes several magnetometers: a central one and at least three peripheral ones in order to compute the full 3 × 3 matrix of magnetic field gradients. The camera is rigidly attached to the MIMU sensor.

The camera is rigidly attached to the MIMU sensor board. The transformation $(\mathrm{R}^{\mathrm{b}}_{\mathrm{c}}, \mathbf{p}^{\mathrm{b}}_{\mathrm{c}})$ between the camera frame and the body frame is assumed to be known. We could alternatively include this transform into the state filter as done in [18] for instance.

Since hardware synchronization was not possible with the components used here, we use a datation approach. In the online estimation process we make the simplifying assumption that images are captured simultaneously with the MIMU sample closest in time. More precisely, the image information at time t_{image_p} is processed using the state estimate at time t_{mimu_k} with k such as:

$$k = \arg\min_{k, t_{mimu_k} < t_{image_p}} |t_{mimu_k} - t_{image_p}|.$$

The temporal error done with this approach is always smaller than the sampling period of the MIMU sensors, which is often below the exposure time of the camera. Besides, this approximation significantly simplifies time management as all measurements are indexed by the MIMU time sampling.

3.2. Evolution Model

In order to estimate the position and orientation of the body coordinate frame in the world frame, one has to track an estimate of its speed and of the magnetic field at the current position of its center. We model the evolution of these quantities with the following differential equations:

$$\dot{\mathtt{R}}^w(t) = \mathtt{R}^w(t)[\boldsymbol{\omega}^b(t)]_\times, \tag{4}$$

$$\dot{\mathbf{p}}^w(t) = \mathtt{R}^w(t)\mathbf{v}^b(t), \tag{5}$$

$$\dot{\mathbf{v}}^b(t) = -[\boldsymbol{\omega}^b(t)]_\times \mathbf{v}^b(t) + \mathtt{R}^{w\mathsf{T}}(t)\mathbf{g}^w + \mathbf{a}^b(t), \tag{6}$$

$$\dot{\mathbf{B}}^b(t) = -[\boldsymbol{\omega}^b(t)]_\times \mathbf{B}^b(t) + \nabla\mathbf{B}^b(t)\mathbf{v}^b(t). \tag{7}$$

This model relies on the following assumptions on the environment:

- **Flat-earth approximation.** We assume that the Est-North-Up (ENU) earth frame at filter initialization is an inertial frame.
- **Stationary magnetic field** in the world frame—although possibly spatially non-uniform, leading to the spatial gradient in (7).

The first assumption is, in practice, often used in VINS literature, in which gyroscopes are not precise enough to measure earth rotational speed (roughly 7×10^{-5} rad/s). However if high-end gyroscopes had been used, it is likely that an estimate based on the simple model (4)–(7) would introduce some error, confusing earth rotational velocity with biases estimates. In our opinion though, even in the case of high-end hardware, it is not clear that the visual pipeline used in this work would be able to provide information reliable enough to estimate biases with sufficient accuracy to be influenced by earth rotational speed. This would be a great challenge to address.

Note also that the world frame differs from the ENU frame defined at filter initialization: they both have their origin at the position of the center of the body frame at initial time, but they can differ by a rotation around the gravity direction, as heading is not observable at initialization.

Equation (7) is the key equation for MI-DR. It relates the evolution of the magnetic field with kinematics quantities and local magnetic gradient. It actually renders the body velocity observable provided the matrix $\nabla\mathbf{B}^b$ is invertible. However, it fails giving useful translational information if the magnetic field is uniform. This happens outdoor, where the magnetic field is uniformly equal to the earth magnetic field or in very large empty rooms. In the latter situation, magnetic gradients can vanish a few meters away from a wall, ceiling or floor.

Also, the stationarity assumption can be challenged in some environments, or punctually if pieces of metal are moving in the vicinity of the magnetometers. While some instationarities can be modeled—as the case of power-line interference [9]—in practice, any useful estimator should also be robust to non modeled effects.

3.3. Model Discretization

The model will be used in a discrete extended Kalman filtering framework described in Section 4 below. We thus discretize it the following way:

$$\mathtt{R}^w_{k+1} = \mathtt{R}^w_k \widetilde{\Delta\mathtt{R}}_{kk+1}, \tag{8}$$

$$\mathbf{p}^w_{k+1} = \mathbf{p}^w_k + \mathtt{R}^w_k \mathbf{v}^b_k \Delta t + \frac{1}{2}\mathbf{g}^w \Delta t^2_{ij} + \mathtt{R}^w_k \widetilde{\Delta\mathbf{p}}_{kk+1}, \tag{9}$$

$$\mathbf{v}^b_{k+1} = \widetilde{\Delta\mathtt{R}}^{\mathsf{T}}_{kk+1}\left(\mathbf{v}^b_k + \mathtt{R}^{w\,\mathsf{T}}_k \mathbf{g}^w \Delta t + \widetilde{\Delta\mathbf{v}}_{kk+1}\right), \tag{10}$$

$$\mathbf{B}^b_{k+1} = \widetilde{\Delta\mathtt{R}}^{\mathsf{T}}_{kk+1}\left(\mathbf{B}^b_k + \widetilde{\boldsymbol{D}}_{\boldsymbol{v};\boldsymbol{kk+1}}\mathbf{v}^b_k + \widetilde{\boldsymbol{D}}_{\mathtt{R};\boldsymbol{kk+1}}\mathtt{R}^{w\,\mathsf{T}}_k \mathbf{g}^w + \widetilde{\boldsymbol{M}}_{\boldsymbol{kk+1}}\right). \tag{11}$$

where we introduced the following notation corresponding to continuous integrals:

$$\widetilde{\Delta\mathtt{R}}_{kk+1} \overset{\mathsf{def}}{=} \Delta\mathtt{R}_k(t_{k+1}), \tag{12}$$

$$\widetilde{\Delta\mathbf{v}}_{kk+1} \overset{\mathsf{def}}{=} \widetilde{\Delta\mathbf{v}}_k(t_{k+1}), \tag{13}$$

$$\widetilde{\Delta\mathbf{p}}_{kk+1} \overset{\mathsf{def}}{=} \int_{t_k}^{t_{k+1}} \widetilde{\Delta\mathbf{v}}_k(\tau)\mathrm{d}\tau, \tag{14}$$

$$\widetilde{\boldsymbol{D}}_{\boldsymbol{v};\boldsymbol{kk+1}} \overset{\mathsf{def}}{=} \int_{t_k}^{t_{k+1}} \Delta\mathtt{R}_k(\tau)\nabla\mathbf{B}^b(\tau)\Delta\mathtt{R}_k(\tau)^{\mathsf{T}}\mathtt{d}\tau, \tag{15}$$

$$\widetilde{\boldsymbol{D}}_{\mathtt{R};\boldsymbol{kk+1}} \overset{\mathsf{def}}{=} \int_{t_k}^{t_{k+1}} \Delta\mathtt{R}_k(\tau)\nabla\mathbf{B}^b(\tau)\Delta\mathtt{R}_k(\tau)^{\mathsf{T}}\left[\tau - t_k\right]\mathtt{d}\tau, \tag{16}$$

$$\widetilde{\boldsymbol{M}}_{\boldsymbol{kk+1}} \overset{\mathsf{def}}{=} \int_{t_k}^{t_{k+1}} \Delta\mathtt{R}_k(\tau)\nabla\mathbf{B}^b(\tau)\Delta\mathtt{R}_k(\tau)^{\mathsf{T}}\Delta\tilde{\mathbf{v}}_k(\tau)\mathtt{d}\tau, \tag{17}$$

$$\text{with } \Delta\mathtt{R}_k(\tau) \overset{\mathsf{def}}{=} \mathbf{I}_3 + \int_{t_k}^{\tau} \Delta\mathtt{R}_k(s)\left[\boldsymbol{\omega}^b(s)\right]_\times \mathtt{d}s \quad \text{and} \quad \widetilde{\Delta\mathbf{v}}_k(\tau) \overset{\mathsf{def}}{=} \int_{t_k}^{\tau} \Delta\mathtt{R}_k(s)\mathbf{a}^b(s)\mathrm{d}s. \tag{18}$$

The notation $\Delta\mathtt{R}_k(\tau)$ is the rotation matrix that transforms point in body frame at time τ to corresponding point in body frame at time t_k, that can be deduced from gyroscopes integration.

These integrals can be computed from unbiased MIMU measurements. We thus estimate the biases of the accelerometers and gyrometers along with previously quantities. These are assumed to follow a first-order Gauss-Markov stochastic evolution:

$$\dot{\mathbf{b}}_g(t) = -\frac{1}{\tau_g}\mathbf{b}_g + \boldsymbol{\jmath}_{bg}, \tag{19}$$

$$\dot{\mathbf{b}}_a(t) = -\frac{1}{\tau_a}\mathbf{b}_a + \boldsymbol{\jmath}_{ba}. \tag{20}$$

where generating noises $\boldsymbol{\jmath}_{bg}$ and $\boldsymbol{\jmath}_{ba}$ satisfy:

$$\mathbb{E}\left(\left[\boldsymbol{\jmath}_{bg}(t), \boldsymbol{\jmath}_{ba}(t)\right]\right) = \mathbf{0}_{6\times 1}, \tag{21}$$

$$\mathbb{E}\left(\left[\boldsymbol{\jmath}_{bg}(t_1), \boldsymbol{\jmath}_{ba}(t_1)\right].\left[\boldsymbol{\jmath}_{bg}(t_2), \boldsymbol{\jmath}_{ba}(t_2)\right]^{\mathsf{T}}\right) = \boldsymbol{\mathcal{W}}_c\delta(t_2 - t_1), \tag{22}$$

$$\boldsymbol{\mathcal{W}}_c = \mathrm{diag}(\sigma^2_{bg;c}\mathbf{I}_3, \sigma^2_{ba;c}\mathbf{I}_3). \tag{23}$$

τ_b, τ_a, $\sigma_{bg;c}$ and $\sigma_{ba;c}$ are expressed in s, s, $\frac{\text{rad}}{s^2}\frac{1}{\sqrt{Hz}}$ and $\frac{\text{m}}{s^3}\frac{1}{\sqrt{Hz}}$ respectively and are characteristics of the IMU. Discretization of the evolution of biases leads to:

$$\mathbf{b}_{g_{k+1}} = -\exp\left(\frac{\Delta t_{ij}}{\tau_g}\right)\mathbf{b}_{g_k} + \boldsymbol{\eta}_{bg} \tag{24}$$

$$\mathbf{b}_{ak+1} = -\exp\left(\frac{\Delta t_{ij}}{\tau_a}\right)\mathbf{b}_{ak} + \boldsymbol{\eta}_{ba} \tag{25}$$

The $\boldsymbol{\eta}_{\mathbf{x}}$ appearing in (24) and (25) will be then modelled as discrete random variables with Gaussian density $\mathcal{N}(0, \boldsymbol{W})$, $\boldsymbol{W}$ being computed as $(t_{k+1} - t_k)\boldsymbol{\mathcal{W}}_c$ [19] (p. 231).

The presented models discretization exhibits integrals that all have to be computed numerically in order to build the filter on. Normally, the choice of the numerical integration method depends on the required accuracy. In the current implementation though, we take a very simple approach: we assume that the biases, acceleration and magnetic field gradient—the last two are in world frame—are constant between two MIMU sample and use the following identity:

$$\widetilde{\Delta\mathrm{R}}_{kk+1} \simeq \mathrm{Exp}\left(\left(\tilde{\boldsymbol{\omega}}_k^b - \mathbf{b}_{g_k}\right)\Delta t_k\right)\,, \tag{26}$$

$$\widetilde{\Delta\mathbf{v}}_{kk+1} \simeq \left(\tilde{\mathbf{a}}_k^b - \mathbf{b}_{ak}\right)\Delta t_k\,, \tag{27}$$

$$\widetilde{\Delta\mathbf{p}}_{kk+1} \simeq \frac{1}{2}\left(\tilde{\mathbf{a}}_k^b - \mathbf{b}_{ak}\right)\Delta t_k^2\,, \tag{28}$$

$$\widetilde{\boldsymbol{D}}_{v;\boldsymbol{kk+1}} \simeq \nabla\tilde{\mathbf{B}}_k^b\Delta t_k\,, \tag{29}$$

$$\widetilde{\boldsymbol{D}}_{\mathrm{R};\boldsymbol{kk+1}} \simeq \frac{1}{2}\nabla\tilde{\mathbf{B}}_k^b\Delta t_k^2\,, \tag{30}$$

$$\widetilde{\boldsymbol{M}}_{\boldsymbol{kk+1}} \simeq \nabla\tilde{\mathbf{B}}_k^b\frac{1}{2}\left(\tilde{\mathbf{a}}_k^b - \mathbf{b}_{ak}\right)\Delta t_k^2. \tag{31}$$

The error induced by this simple integration scheme is limited by the relatively high frequency of MIMU sample (325 Hz).

3.4. Sensors Error Model

Sensors errors propagate through the discrete model when used for filtering. We assume that the noisy sensor reading at time k (the *input vector* of the filter) can be written:

$$\tilde{\mathbf{u}}_k = \underbrace{\begin{bmatrix} \tilde{\mathbf{a}}_k^b \\ \tilde{\boldsymbol{\omega}}_k^b \\ \mathrm{Vec}\left(\nabla\tilde{\mathbf{B}}_k^b\right) \end{bmatrix}}_{\text{measurement}} = \underbrace{\begin{bmatrix} \mathbf{a}_k \\ \boldsymbol{\omega}_k \\ \mathrm{Vec}\left(\nabla\mathbf{B}_k\right) \end{bmatrix}}_{\text{real values}} + \begin{bmatrix} \mathbf{I}_3 & 0_3 & 0_{3\times5} \\ 0_3 & \mathbf{I}_3 & 0_{3\times5} \\ 0_3 & 0_3 & \mathcal{P}_{\nabla\mathbf{B}} \end{bmatrix} \underbrace{\begin{bmatrix} \boldsymbol{\eta}_{\mathbf{a}_k^b} \\ \boldsymbol{\eta}_{\boldsymbol{\omega}_k^b} \\ \boldsymbol{\eta}_{\nabla\mathbf{B}_k^b} \end{bmatrix}}_{\text{input noise } \delta\mathbf{u}_k\in\mathbb{R}^{11}} \tag{32}$$

The noise $\delta\mathbf{u}_k$ is assumed to be gaussian $\delta\mathbf{u}_k \propto \mathcal{N}(0, \boldsymbol{\Sigma}_{\mathbf{u}})$ and is a sensor characteristic. We use here:

$$\boldsymbol{\Sigma}_{\mathbf{u}} = \begin{bmatrix} \sigma_\omega^2\mathbf{I}_3 & 0_{3\times3} & 0_{3\times5} \\ 0_{3\times3} & \sigma_{\mathbf{a}}^2\mathbf{I}_3 & 0_{3\times5} \\ 0_{5\times3} & 0_{5\times3} & \boldsymbol{\Sigma}_{\nabla\mathbf{B}} \end{bmatrix} \text{ with } \sigma_\omega \text{ in } \frac{rad}{s},\ \sigma_{\mathbf{a}} \text{ in } \frac{m}{s^2} \text{ and } \boldsymbol{\Sigma}_{\nabla\mathbf{B}} \text{ in } \frac{\text{Gauss}}{m} \in \mathbb{R}^{5\times5}. \tag{33}$$

Note that $\mathcal{P}_{\nabla\mathbf{B}}$ reflects that we explicitly exploit the symmetry in the magnetic field from Maxwell equation. They implies that the gradient should be a symmetric matrix and of zero trace and thus has

5 degrees of freedom, instead of the nine coefficient of the full matrix. $\mathcal{P}_{\nabla \mathbf{B}} \in \mathbb{R}^{9\times 5}$ is thus the matrix of the application generating the vectorized gradient matrix from minimal gradient coordinates:

$$\begin{array}{ccc} \mathbb{R}^5 & \rightarrow & \mathbb{R}^9 \\ \begin{pmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \end{pmatrix} & \mapsto & \text{Vec}\left(\begin{pmatrix} g_1 & g_2 & g_3 \\ g_2 & g_4 & g_5 \\ g_3 & g_5 & -g_1 - g_4 \end{pmatrix}\right) \end{array} \tag{34}$$

4. Tight Fusion Filter

This section describe thoroughly the MI-MSCKF filter that is used in the experimental Section 5. Section 4.1 describes the parametrization of the state of the filter. Section 4.2 is dedicated to the propagation step while Section 4.3 details the magnetic and visual measurement equations.

4.1. State and Error State

We define the state space at time index k, $\mathcal{X}_k$ as a manifold which is compound of :

- the *keyframe poses* state space $\mathcal{K}_k$, which elements are the poses of a set of N past frames at time indexes $\{i_1, ...i_N\}$ not necessarily temporally successive but close in time. With poses written $\xi_i = (\mathtt{R}_i^w, \mathbf{p}_i^w)$, this part of the state have the following form:

$$\left(\xi_{i_1}, \cdots, \xi_{i_N}\right) \in \mathcal{K}_k = \left(\mathrm{SO}(3) \times \mathbb{R}^3\right)^N \tag{35}$$

- the *current mimu state* $\mathcal{S}$ space which elements have the following form:

$$\mathbf{s}_k = \left(\xi_k, \mathbf{v}_k^b, \mathbf{B}_k^b, \mathbf{b}_{ak}, \mathbf{b}_{gk}\right)^{\mathsf{T}} \in \mathrm{SO}(3) \times \mathbb{R}^3 \times \mathbb{R}^{12} \tag{36}$$

A complete state space element at time index k is thus noted:

$$\mathbf{X}_k = \left(\xi_{i_1}, \cdots, \xi_{i_N}, \mathbf{s}_k\right) \in \mathcal{X}_k = \mathcal{K}_k \times \mathcal{S} \tag{37}$$

We use the Lie group structure of this manifold and its tangent space to (i) define the error tracked by the filter and (ii) define the Jacobian of the measurement process. This derives from the fact that a perturbations around an element can be expressed as an element of its Lie algebra. We use the $\boxplus$ operator symbol, so that $\mathbf{X}_k \boxplus \delta\mathbf{X}_k$ computes a new state in $\mathcal{X}_k$ from a tangent perturbation $\delta\mathbf{X}_k$ around $\mathbf{X}_k$. We define it as regular addition operation for all components of the state except for pose states where we use:

$$\xi \boxplus \delta\xi = \left(\mathrm{Exp}\left(\delta\xi_{4:6}\right)\mathtt{R}, \mathbf{p} + \delta\xi_{1:3}\right) \tag{38}$$

Similarly we define the reciprocal operator $\boxminus$ as the binary operator giving the perturbation element between two states of $\mathcal{X}$. It is defined as regular minus operation except for pose states for which it is:

$$\xi_2 \boxminus \xi_1 = \left[\mathrm{Log}(\mathtt{R}_2\mathtt{R}_1^{-1})^T, \mathbf{p}_2^{\mathsf{T}} - \mathbf{p}_1^{\mathsf{T}}\right]^{\mathsf{T}} \tag{39}$$

We here define the error state as the application $\boxminus$ operator between the true state and the estimated state, noted hereafter with an hat. It is thus an element of the tangent space at the current estimate.

$$\epsilon_k \overset{\text{def}}{=} \mathbf{X}_k \boxminus \hat{\mathbf{X}}_k \Leftrightarrow \mathbf{X}_k = \hat{\mathbf{X}}_k \boxplus \epsilon_k \tag{40}$$

Note that this implies a parametrization of the rotation error in *world* frame, and is different from our previous work [13] where rotation error was parametrized in the body frame.

The filtering process propagates the estimated mean, along with an estimate of uncertainty. This uncertainty is represented as a Gaussian density on the error state $\epsilon_k \propto \mathcal{N}(0, \mathbf{P}_k)$ in order to take advantage of the Lie group structure defined previously, i.e., a minimal parametrization and a locally euclidean structure in tangent space.

For numerical reasons, the covariance $\mathbf{P}_k$ will be tracked by the filter in an square-root information form, such that we have the relationship $\mathbf{P}_k = (\hat{\boldsymbol{\mathcal{S}}}_k^\top \hat{\boldsymbol{\mathcal{S}}}_k)^{-1}$ with $\hat{\boldsymbol{\mathcal{S}}}_k$ an upper triangular matrix. The next section describes how this quantity evolves through the different steps of the filter: propagation and update.

4.2. Propagation/Augmentation/Marginalization

At the time of arrival of a new IMU data $k+1$, $\mathbf{X}_{k|k}$ and $\hat{\boldsymbol{\mathcal{S}}}_{k|k}$ are propagated. This is summarized by Figure 3 and splitted in three steps. First, the mimu state $\boldsymbol{s}_k$ is propagated with discretized model (Section 4.2.1), then the full state is augmented with the resulting new mimu state $\boldsymbol{s}_{k+1}$ (Section 4.2.2). Finally, some part of this augmented state are marginalized before the update step (Section 4.2.3).

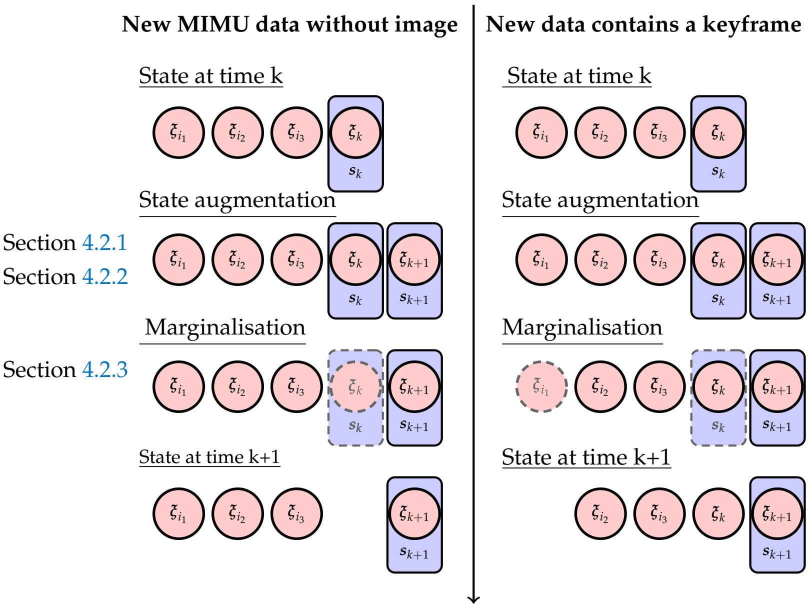


Figure 3. Illustration of state augmentation and marginalization across time as described in Section 4.2.

4.2.1. Propagation

Keyframe poses states are estimation of physical quantities blocked at fixed instant in time, their error do not evolve with time and thus they are propagated with an identity function. Besides, the current mimu state error is propagated according to

$$\boldsymbol{s}_{k+1} = \mathbf{f}_{\text{mimu}}(\boldsymbol{s}_k, \tilde{\mathbf{u}}_k, \boldsymbol{\eta}_k) \tag{41}$$

where the *discrete mimu process function* $\mathbf{f}_{\text{mimu}}$ summarizes (8)–(11) and (24)–(25).

The mimu state error is increased from the three sources of uncertainty: the stochastic model of biases, the measurement noise $\delta\mathbf{u}_k$ on the input vector and the uncertainty of the previous estimate:

$$\begin{aligned}\epsilon_{\text{mimu},k+1} &= \mathbf{f}_{\text{mimu}}(\hat{\mathbf{s}}_k \boxplus \epsilon_k^{\text{mimu}}, \tilde{\mathbf{u}}_k - \delta\mathbf{u}_k, \boldsymbol{\eta}_k) \boxminus \mathbf{f}_{\text{mimu}}(\hat{\mathbf{s}}_k, \tilde{\mathbf{u}}_k, 0) && (42)\\ &\simeq \boldsymbol{\Phi}_{k+1,k}^{\text{mimu}}\epsilon_k^{\text{mimu}} + \mathbf{G}_{k+1,k}^{\text{mimu}}\delta\mathbf{u}_k + \mathbf{C}_{k+1,k}^{\text{mimu}}\boldsymbol{\eta}_k + O\left(\epsilon_k^{\text{mimu}}, \delta\mathbf{u}_k, \boldsymbol{\eta}_k\right) && (43)\end{aligned}$$

The expression of matrices $\boldsymbol{\Phi}_{k+1,k}^{\text{mimu}}$, $\mathbf{G}_{k+1,k}^{\text{mimu}}$, $\mathbf{C}_{k+1,k}^{\text{mimu}}$ are derived by 1st order development of (42):

$$\boldsymbol{\Phi}_{k+1,k}^{\text{mimu}} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \boldsymbol{\Phi}_{k+1,k\,\mathrm{R}\mathbf{b}_g}^{\text{mimu}} & \mathbf{0}_3 \\ \left[-\mathrm{R}_k(\mathbf{v}_k^b\Delta t + \widetilde{\Delta\mathbf{p}}_{kk+1})\right]_\times & \mathbf{I}_3 & \mathrm{R}_k\Delta t & \mathbf{0}_3 & \boldsymbol{\Phi}_{k+1,k\,\mathbf{pb}_g}^{\text{mimu}} & \boldsymbol{\Phi}_{k+1,k\,\mathbf{pb}_g}^{\text{mimu}} \\ \mathrm{R}_{k+1}^{\mathsf{T}}\left[\mathbf{g}^w\right]_\times \Delta t & \mathbf{0}_3 & \Delta\mathrm{R} & \mathbf{0}_3 & \boldsymbol{\Phi}_{k+1,k\,\mathbf{vb}_g}^{\text{mimu}} & \boldsymbol{\Phi}_{k+1,k\,\mathbf{vb}_a}^{\text{mimu}} \\ \Delta\mathrm{R}^{\mathsf{T}}\widetilde{\boldsymbol{D}}_{\mathrm{R};\boldsymbol{kk+1}}\left[\mathbf{g}^w\right]_\times & \mathbf{0}_3 & \Delta\mathrm{R}^{\mathsf{T}}\widetilde{\boldsymbol{D}}_{\boldsymbol{v;kk+1}} & \Delta\mathrm{R}^{\mathsf{T}} & \boldsymbol{\Phi}_{k+1,k\,\mathbf{Bb}_g}^{\text{mimu}} & \boldsymbol{\Phi}_{k+1,k\,\mathbf{Bb}_a}^{\text{mimu}} \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\exp\left(\frac{\Delta t}{\tau_g}\right) & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\exp\left(\frac{\Delta t}{\tau_a}\right) \end{bmatrix} \tag{44}$$

$$\mathbf{G}_{k+1,k}^{\text{mimu}} = \begin{bmatrix} -\boldsymbol{\Phi}_{k+1,k\,\mathrm{R}\mathbf{b}_g}^{\text{mimu}} & 0 & 0 \\ -\boldsymbol{\Phi}_{k+1,k\,\mathbf{pb}_g}^{\text{mimu}} & -\boldsymbol{\Phi}_{k+1,k\,\mathbf{pb}_a}^{\text{mimu}} & 0 \\ -\boldsymbol{\Phi}_{k+1,k\,\mathbf{vb}_g}^{\text{mimu}} & -\boldsymbol{\Phi}_{k+1,k\,\mathbf{vb}_a}^{\text{mimu}} & 0 \\ -\boldsymbol{\Phi}_{k+1,k\,\mathbf{Bb}_g}^{\text{mimu}} & -\boldsymbol{\Phi}_{k+1,k\,\mathbf{Bb}_a}^{\text{mimu}} & \mathbf{G}_{k+1,k\,\mathbf{B}\nabla\mathbf{B}}^{\text{mimu}} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{C}_{k+1,k}^{\text{mimu}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \mathbf{I}_3 & 0 \\ 0 & \mathbf{I}_3 \end{bmatrix} \tag{45}$$

$$\begin{aligned}\boldsymbol{\Phi}_{k+1,k\,\mathrm{R}\mathbf{b}_g}^{\text{mimu}} &= -\mathrm{R}_k\Delta t && (46)\\ \boldsymbol{\Phi}_{k+1,k\,\mathbf{vb}_g}^{\text{mimu}} &= -\Delta t\left[\mathbf{v}_{k+1}\right]_\times + \partial_{\mathbf{b}_{g_k}}\left(\widetilde{\Delta\mathbf{v}}_{kk+1}\right) && (47)\\ \boldsymbol{\Phi}_{k+1,k\,\mathbf{Bb}_g}^{\text{mimu}} &= -\Delta t\left[\mathbf{B}_{k+1}\right]_\times && (48)\\ &\quad + \Delta\mathrm{R}^{\mathsf{T}}\left(\mathbf{v}_k^{\mathsf{T}} \otimes \mathbf{I}_3\right)\partial_{\mathbf{b}_g}\left(\mathrm{Vec}\left(\widetilde{\boldsymbol{D}}_{\boldsymbol{v;kk+1}}\right)\right) \\ &\quad + \Delta\mathrm{R}^{\mathsf{T}}\left(\mathbf{g}^{\mathsf{T}}\mathrm{R}_k \otimes \mathbf{I}_3\right)\partial_{\mathbf{b}_g}\left(\mathrm{Vec}\left(\widetilde{\boldsymbol{D}}_{\mathrm{R};\boldsymbol{kk+1}}\right)\right) \\ &\quad + \Delta\mathrm{R}^{\mathsf{T}}\partial_{\mathbf{b}_g}\left(\widetilde{\boldsymbol{M}}_{\boldsymbol{kk+1}}\right) && (49)\\ \boldsymbol{\Phi}_{k+1,k\,\mathbf{pb}_g}^{\text{mimu}} &= \mathrm{R}_k\partial_{\mathbf{b}_{g_k}}\left(\widetilde{\Delta\mathbf{p}}_{kk+1}\right) && (50)\\ \boldsymbol{\Phi}_{k+1,k\,\mathbf{vb}_a}^{\text{mimu}} &= \Delta\mathrm{R}^{\mathsf{T}}\partial_{\mathbf{b}_{ak}}\left(\widetilde{\Delta\mathbf{v}}_{kk+1}\right) && (51)\\ \boldsymbol{\Phi}_{k+1,k\,\mathbf{pb}_a}^{\text{mimu}} &= \mathrm{R}_k\partial_{\mathbf{b}_{ak}}\left(\widetilde{\Delta\mathbf{p}}_{kk+1}\right) && (52)\\ \boldsymbol{\Phi}_{k+1,k\,\mathbf{Bb}_a}^{\text{mimu}} &= \Delta\mathrm{R}^{\mathsf{T}}\partial_{\mathbf{b}_{ak}}\left(\widetilde{\boldsymbol{M}}_{\boldsymbol{kk+1}}\right) && (53)\\ \mathbf{G}_{k+1,k\,\mathbf{B}\nabla\mathbf{B}}^{\text{mimu}} &= \Delta\mathrm{R}^{\mathsf{T}}\left(\mathbf{v}_k^{\mathsf{T}} \otimes \mathbf{I}_3\right)\partial_{\nabla\mathbf{B}}\left(\mathrm{Vec}\left(\widetilde{\boldsymbol{D}}_{\boldsymbol{v;kk+1}}\right)\right) \\ &\quad + \Delta\mathrm{R}^{\mathsf{T}}\left(\mathbf{g}^{\mathsf{T}}\mathrm{R}_k \otimes \mathbf{I}_3\right)\partial_{\nabla\mathbf{B}}\left(\mathrm{Vec}\left(\widetilde{\boldsymbol{D}}_{\mathrm{R};\boldsymbol{kk+1}}\right)\right)\mathcal{P}_{\nabla\mathbf{B}} + \Delta\mathrm{R}^{\mathsf{T}}\partial_{\nabla\mathbf{B}}\left(\widetilde{\boldsymbol{M}}_{\boldsymbol{kk+1}}\right)\mathcal{P}_{\nabla\mathbf{B}} && (54)\end{aligned}$$

where $\mathcal{P}_{\nabla\mathbf{B}}$ is defined as in (34). Note that we wrote here the transition and noise matrices as a function of the integrals (12)–(17), independently of the way the are computed, so that these expressions are still valid if one choose to compute the integrals with a more sophisticated scheme. Derivative of these integrals with respect to the biases are also required: these quantities should be computed simultaneously with the integrals. In our implementation, they are computed easily with the approximations made in (26)–(31).

4.2.2. State Augmentation

When MIMU sample $k+1$ arrives, the mimu propagation function is used to augment the state with the new current mimu state $\hat{\mathbf{s}}_{k+1} = \mathbf{f}_{\text{mimu}}(\hat{\mathbf{s}}_k, \tilde{\mathbf{u}}_k, 0)$, leading to the augmented state $\hat{\mathbf{X}}^{\oplus}_{k+1|k}$. The error state square root information matrix is augmented accordingly to [14], $\hat{\boldsymbol{\mathcal{S}}}^{\oplus}_{k+1|k}$ using the Jacobian derived in previous subsection.

$$\hat{\mathbf{X}}^{\oplus}_{k+1|k} \overset{\text{def}}{=} \left(\mathbf{X}_{k|k}, \mathbf{s}_{k+1}\right) \tag{55}$$

$$\hat{\boldsymbol{\mathcal{S}}}^{\oplus}_{k+1|k} \overset{\text{def}}{=} \begin{bmatrix} \hat{\boldsymbol{\mathcal{S}}}_{k|k} & 0 \\ \boldsymbol{V_1} & \boldsymbol{V_2} \end{bmatrix} \tag{56}$$

with

$$\boldsymbol{V_1} = [\mathbf{0}_{18\times 6}, \cdots, \mathbf{0}_{18\times 6}, \boldsymbol{Q}_{k+1}^{-\frac{1}{2}} \boldsymbol{\Phi}^{\text{mimu}}_{k+1,k}{}_{k+1,k}] \tag{57}$$

$$\boldsymbol{V_2} = \boldsymbol{Q}_{k+1}^{-\frac{1}{2}} \tag{58}$$

and the discrete model noise at $\boldsymbol{Q}_k$ time k :

$$\boldsymbol{Q}_{k+1} = \left(\mathbf{G}^{\text{mimu}}_{k+1,k} \mathbf{C}^{\text{mimu}}_{k+1,k}\right) \begin{pmatrix} \boldsymbol{\Sigma}_{\mathbf{u}} & 0 \\ 0 & \boldsymbol{W} \end{pmatrix} \begin{pmatrix} \mathbf{G}^{\text{mimu}^T}_{k+1,k} \\ \mathbf{C}^{\text{mimu}^T}_{k+1,k} \end{pmatrix}. \tag{59}$$

4.2.3. Marginalization of Old State

Then, in order to bound the size of $\hat{\mathbf{X}}_{k+1|k}$, some part of it are marginalized. The state elements to be marginalized depend on the type of data available at the current timestamps as depicted in Figure 3. If only MIMU data (without an new keyframe image attached) are arriving at time k, $\mathbf{s}_k$ is marginalized. If MIMU data and a new keyframe image are available at time k, the oldest keyframe pose is marginalized together with the non pose element of $\mathbf{s}_k$. Note that since the image frame rate is well below MIMU frequency, the first case happens more often than the second case.

Within the square root information form, marginalization is done similarly to [14]. With $\boldsymbol{\Pi}_k$ being the matrix permutation putting the *to marginalize* error states at the beginning, a QR decomposition of a square root information matrix of the permuted augmented error state vector writes

$$\hat{\boldsymbol{\mathcal{S}}}^{\oplus}_{k+1|k} \boldsymbol{\Pi}_k = \mathbf{O}_p \boldsymbol{R}_p, \quad \begin{matrix} \mathbf{O}_p \in \mathcal{O}(n) \\ \boldsymbol{R}_p \in \mathbb{R}^n \end{matrix} \tag{60}$$

$$= \mathbf{O}_p \begin{bmatrix} * & * \\ 0 & \hat{\boldsymbol{\mathcal{S}}}_{k+1|k} \end{bmatrix} \tag{61}$$

We obtain the predicted state $\hat{\mathbf{X}}_{k+1|k}$ removing marginalized states, and its—upper triangular—square-root information matrix $\hat{\boldsymbol{\mathcal{S}}}_{k+1|k}$. If no measurement update has to be performed at current time step they can be used directly as $\mathbf{X}_{k+1|k+1}$ and $\hat{\boldsymbol{\mathcal{S}}}_{k+1|k+1}$ for the next propagation.

Note: The marginalization of a joint Gaussian distribution with a square-root information form is not often demonstrated in book or lecture but can be deduced from the full information form $\boldsymbol{\Lambda}_{\text{joint}}$:

$$\text{if } \boldsymbol{\Lambda}_{\text{joint}} = \begin{bmatrix} \boldsymbol{\Lambda}_M & \boldsymbol{\Lambda}_{MR} \\ \boldsymbol{\Lambda}^{\mathsf{T}}_{MR} & \boldsymbol{\Lambda}_R \end{bmatrix} = \hat{\boldsymbol{\mathcal{S}}}^{\mathsf{T}}_{\text{joint}} \hat{\boldsymbol{\mathcal{S}}}_{\text{joint}} = \begin{bmatrix} \hat{\boldsymbol{\mathcal{S}}}^{\mathsf{T}}_M \hat{\boldsymbol{\mathcal{S}}}_M & \hat{\boldsymbol{\mathcal{S}}}^{\mathsf{T}}_M \hat{\boldsymbol{\mathcal{S}}}_{MR} \\ \hat{\boldsymbol{\mathcal{S}}}^{\mathsf{T}}_{MR} \hat{\boldsymbol{\mathcal{S}}}_M & \hat{\boldsymbol{\mathcal{S}}}^{\mathsf{T}}_R \hat{\boldsymbol{\mathcal{S}}}_R + \hat{\boldsymbol{\mathcal{S}}}^{\mathsf{T}}_{MR} \hat{\boldsymbol{\mathcal{S}}}_{MR} \end{bmatrix} \text{ with } \hat{\boldsymbol{\mathcal{S}}}_{\text{joint}} = \begin{bmatrix} \hat{\boldsymbol{\mathcal{S}}}_M & \hat{\boldsymbol{\mathcal{S}}}_{MR} \\ 0 & \hat{\boldsymbol{\mathcal{S}}}_R \end{bmatrix} \tag{62}$$

then the square root information matrix resulting from marginalization of the the M variables is: $\hat{\boldsymbol{\mathcal{S}}}_R^{\text{marg}} = \hat{\boldsymbol{\mathcal{S}}}_R$. This is deduced by calculus from the usually demonstrated fact that: $\boldsymbol{\Lambda}_R^{\text{marg}} = \boldsymbol{\Lambda}_R - \boldsymbol{\Lambda}_{MR}\boldsymbol{\Lambda}_M^{-1}\boldsymbol{\Lambda}_{RM}^{\mathsf{T}}$.

4.3. Measurement Update

The filter processes two kinds of measurement: (i) the magnetic one that compares the magnetic field measured at the current timestamp with the magnetic field predicted by the filter; (ii) the visual measurement equation for features for which tracking has just ended.

We first briefly recall the update process of the inverse square root filter on a manifold. Let us suppose that some measurement occurs that can be modeled as:

$$\mathbf{h}(\mathbf{X}_{k+1}) = \tilde{\mathbf{h}}_{k+1} + \boldsymbol{\eta}_{\mathbf{h}} \in \mathbb{R}^{n_m},\ \boldsymbol{\eta}_{\mathbf{h}} \sim \mathcal{N}(0, \Sigma_{\mathbf{h}}) \tag{63}$$

with n_m the dimension of the measurement vector. Writing the dimension of the predicted state as n_s, the measurement error $\mathbf{z}_{k+1} = \mathbf{h}(\hat{\mathbf{X}}_{k+1|k}) - \tilde{\mathbf{h}}_{k+1}$ and $\boldsymbol{H}$ the jacobian of the application:

$$\begin{array}{ccl} \mathbb{R}^{n_s} & \to & \mathbb{R}^{n_m} \\ \delta\boldsymbol{X} & \mapsto & \mathbf{h}(\hat{\mathbf{X}}_{k+1|k} \boxplus \delta\boldsymbol{X}) \end{array}, \tag{64}$$

the update step finds the tangent correction $\delta\mathbf{X}^*$ that minimizes the following linearized cost:

$$\begin{aligned} \mathcal{C}(\delta\mathbf{X}) &= \|\delta\mathbf{X}\|^2_{\mathbf{P_{k+1|k}}} + \|\boldsymbol{H}\delta\mathbf{X} - \mathbf{z}_{k+1}\|^2_{\boldsymbol{\Sigma}_h} \\ &= \left\|\hat{\boldsymbol{\mathcal{S}}}_{k+1|k}.\delta\mathbf{X}\right\|^2 + \left\|\boldsymbol{\Sigma}_{\mathbf{h}}^{-\frac{1}{2}}\left(\boldsymbol{H}\delta\mathbf{X} - \mathbf{z}_{k+1}\right)\right\|^2 \\ &= \left\|\begin{bmatrix} \hat{\boldsymbol{\mathcal{S}}}_{k+1|k} \\ \boldsymbol{\Sigma}_{\mathbf{h}}^{-\frac{1}{2}}\boldsymbol{H} \end{bmatrix}\delta\mathbf{X} - \begin{bmatrix} 0 \\ \boldsymbol{\Sigma}_{\mathbf{h}}^{-\frac{1}{2}}\mathbf{z}_{k+1} \end{bmatrix}\right\|^2. \end{aligned} \tag{65}$$

The optimum point can be obtained by of a thin QR decomposition:

$$\begin{bmatrix} \hat{\boldsymbol{\mathcal{S}}}_{k+1|k} \\ \boldsymbol{\Sigma}_h^{-\frac{1}{2}}\boldsymbol{H} \end{bmatrix} = \mathbf{O}_u\boldsymbol{R}_u,\quad \begin{array}{l} \mathbf{O}_u \in \boldsymbol{\mathcal{O}}(n_s + n_m) \\ \boldsymbol{R}_u \in \mathbb{R}^{n_s + n_m} \end{array} \tag{66}$$

$$\mathcal{C}(\delta\mathbf{X}) = \left\|\boldsymbol{R}_u\delta\mathbf{X} - \mathbf{O}_u^{\mathsf{T}}\begin{bmatrix} 0 \\ \boldsymbol{\Sigma}_{\mathbf{h}}^{-\frac{1}{2}}\mathbf{z}_{k+1} \end{bmatrix}\right\|^2, \tag{67}$$

$\boldsymbol{R}_u$ being upper triangular, $\delta\mathbf{X}^*$ is efficiently computed by back-substitution. This optimal correction is finally applied to the predicted state with the retraction operator:

$$\mathbf{X}_{k+1|k+1} = \hat{\mathbf{X}}_{k+1|k} \boxplus \delta\mathbf{X}^* \tag{68}$$

$$\hat{\boldsymbol{\mathcal{S}}}_{k+1|k} = \boldsymbol{R}_u \mathrm{J}_r^{-1}. \tag{69}$$

With J_r being the Jacobian at $e = 0$ of:

$$\begin{aligned} &\mathbb{R}^{n_s} \to \mathbb{R}^{n_s} \\ &\quad e \mapsto (\hat{\mathbf{X}}_{k+1|k} \boxplus (\delta\mathbf{X}^* + e)) \boxminus (\hat{\mathbf{X}}_{k+1|k} \boxplus \delta\mathbf{X}^*). \end{aligned} \tag{70}$$

Intuitively, this Jacobian transforms the square-root information matrix from the tangent space of predicted state to the tangent space of the updated state. Note that, in the current parametrization choice (cf. (38)), this Jacobian is the identity matrix; this was not the case in our previous work [13] where the rotation error was expressed in body frame.

4.3.1. Magnetic Measurement Update

The magnetic measurement update is the simplest and uses at MIMU frequency the direct magnetic field measurement which writes:

$$h_{\mathbf{B}}(\mathbf{X}_k) = \boldsymbol{P}_{\mathbf{B}_b}\mathbf{X}_k \tag{71}$$

$$\Sigma_{h_{\mathbf{B}}} = \sigma_{\mathbf{B}}^2 \mathbf{I}_3 \tag{72}$$

with $\boldsymbol{P}_{\mathbf{B}_b}$ the projection operator from the state space to the coordinates of the state corresponding to $\mathbf{B}_b$. $\sigma_{\mathbf{B}}$ is the noise of the magnetometers reading.

4.3.2. Opportunistic Feature Tracks Measurement Update

We use the feature tracks in the same way as proposed by [20]. We derive here the equation for completeness.

When a feature track ends or when the frame in which the feature was detected is about to be marginalized, we process the entire feature track as a measurement constraining the pose of each in-state frame where the feature was detected.

The predicted reprojection of feature i in frame at time t is written:

$$\mathbf{pr}_t^i(\boldsymbol{\xi}_t, \mathbf{l}_i^w) = \pi\left(\mathrm{R}_{\mathrm{c}}^{\mathrm{b}\,\mathsf{T}}\left(\mathrm{R}_t^{w\,\mathsf{T}}\left(\mathbf{l}_i^w - \mathbf{p}_t^w\right) - \mathbf{p}_{\mathrm{c}}^{\mathrm{b}}\right)\right) \in \mathbb{R}^2 \tag{73}$$

with $\mathbf{l}_i^w \in \mathbb{R}^3$ the 3D-position of the features i in inertial coordinates and $\pi : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ the projection function of the camera. We recall that $(\mathrm{R}_{\mathrm{c}}^{\mathrm{b}}, \mathbf{p}_{\mathrm{c}}^{\mathrm{b}})$ is the known transform between the body frame and the camera frame. $\mathbf{l}_i^w$ is computed by a fast triangulation function from known in-state poses and the measured observation, noted $\boldsymbol{o}_{it}$.

By stacking all these reprojections, we can write the non linear measurement function:

$$\mathbf{h}_{fi}(\mathbf{X}, \mathbf{l}_i^w) = \begin{bmatrix} \cdot \\ \mathbf{pr}_t^i(\boldsymbol{\xi}_t, \mathbf{l}_i^w) \\ \cdot \end{bmatrix} = \begin{bmatrix} \cdot \\ \boldsymbol{o}_{it} \\ \cdot \end{bmatrix} + \boldsymbol{\eta}_{fi}\,, \tag{74}$$

with $\boldsymbol{\eta}_{fi}$ is assumed to be an additive Gaussian white noise : $\boldsymbol{\eta}_{fi} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_C)$, $\boldsymbol{\Sigma}_C = \sigma_c I_{2m}$. We note subsequently $\boldsymbol{o}_i$ the vector resulting from stacking of 2-vector observation $\boldsymbol{o}_{it}$.

The attentive reader may note that this measurement equation is not of the form of (63), because $\mathbf{l}_i^w$ is not part of our state: for this reason we can not directly use (65) and (67). Worse, the computed $\mathbf{l}^w$ estimate is correlated with the current state error, so that we can not just use it as a fixed constant. The solution used here aims at expressing a projection of the residual that depends only on the poses, up to the first order.

We start by linearizing the residual then proceeds the minimization in two steps to extract the used measurement equation. Linearization of $\mathbf{r}_i : \mathbf{X}, \mathbf{l}_i^w \mapsto \left(\mathbf{h}_{fi}(\mathbf{X}, \mathbf{l}_i^w) - \boldsymbol{o}_{\mathrm{i}}\right)$ yields:

$$\begin{aligned} &\mathbf{r}_i(\mathbf{X}_{k+1|k} \boxplus \delta\boldsymbol{X}, \mathbf{l}_i^w + \delta\mathbf{l}_i^w) \\ &\quad \simeq \boldsymbol{F}_i\delta\boldsymbol{X} + \boldsymbol{E}_f\delta\mathbf{l}_i^w + \left(\mathbf{h}_{fi}(\mathbf{X}_{k+1|k}, \mathbf{l}_i^w) - \boldsymbol{o}_i\right) \\ &\quad = \boldsymbol{F}_k\delta\boldsymbol{X} + \boldsymbol{E}_f\delta\mathbf{l}_i^w - \delta\boldsymbol{o}_i \end{aligned} \tag{75}$$

$\delta \boldsymbol{o}_i = \mathbf{h}_{fi}(\mathbf{X}_{k+1|k}, \mathbf{l}_i^w) - \boldsymbol{o}_i$ is the $2m$-vector of predicted residual error. $\boldsymbol{E}_f$ is a $2m \times 3$ matrix of rank 3 ; m denoting the number of observations for the feature. The rank of $\boldsymbol{E}_f$ is guaranteed during the triangulation. Its QR decomposition writes:

$$\boldsymbol{E}_f = [\mathbf{O}_{E1}, \mathbf{O}_{E0}] \begin{bmatrix} \boldsymbol{R}_{E1} \\ \mathbf{0}_{2m-3\times 2m} \end{bmatrix} \tag{76}$$

$$\mathbf{O}_{E1} \in \mathbb{R}^{2m\times 3}, \mathbf{O}_{E0} \in \mathbb{R}^{2m\times 2m-3}, \boldsymbol{R}_{E1} \in GL_3(\mathbb{R}); \tag{77}$$

Properties of square orthogonal matrices on the L_2 norm of vectors allow to split the cost function into two terms, one depending only of the current predicted state vector.

$$\begin{aligned} \min_{\delta\boldsymbol{X}, \delta\mathbf{l}_i^w} \|\mathbf{r}_i\|^2_{\boldsymbol{\Sigma}_C} &= \min_{\delta\boldsymbol{X}, \delta\mathbf{l}_i^w} \left\| \begin{bmatrix} \mathbf{O}_{E1}^\top \\ \mathbf{O}_{E0}^\top \end{bmatrix} \mathbf{r_i} \right\|^2_{\boldsymbol{\Sigma}_C} \\ &= \min_{\delta\boldsymbol{X}, \delta\mathbf{l}_i^w} \left\| \mathbf{O}_{E1}^\top \boldsymbol{F}_i \delta\boldsymbol{X} + \boldsymbol{R}_{E1} \delta\mathbf{l}_i^w - \mathbf{O}_{E1}^\top \delta\boldsymbol{o}_i \right\|^2_{\boldsymbol{\Sigma}_C} \\ &+ \min_{\delta\boldsymbol{X}} \left\| \boldsymbol{Q}_{E0}^\top \boldsymbol{F}_i \delta\boldsymbol{X} - \mathbf{O}_{E0}^\top \delta\boldsymbol{o}_i \right\|^2_{\boldsymbol{\Sigma}_C} \end{aligned} \tag{78}$$

As $\boldsymbol{R}_{E1}$ is invertible, the first term of the quantity to be minimized can be reduced to zero for all $\delta\boldsymbol{X}$. Minimization reduces thus to:

$$\min_{\delta\boldsymbol{X}} \left\| \mathbf{O}_{E0}^\top \boldsymbol{F}_i \delta\boldsymbol{X} - \mathbf{O}_{E0}^\top \delta\boldsymbol{o}_i \right\|^2_{\boldsymbol{\Sigma}_C} \tag{79}$$

Finally, the previous linearized residual is used in the cost function (65), i.e., we use for the $\boldsymbol{H}$ matrix, the $\mathbf{z}$ error vector and the covariance of measurement $\Sigma_{\mathbf{h}}$ the quantities:

$$\boldsymbol{H}_f = \mathbf{O}_{E0}^\top \boldsymbol{F}_i, \qquad \mathbf{z}_f = \mathbf{O}_{E0}^\top \delta\boldsymbol{o}_i, \qquad \boldsymbol{\Sigma}_{\mathbf{h}_f} = \sigma_C I_{2m-3}. \tag{80}$$

Note that everything happens here as if we introduced the features position into the state, but instantly marginalized it.

4.4. Filter Initialization

One sensitive issue in a VINS filter is initialization. It usually involves some specific algorithm as described in [21,22] for instance. In our implementation, we proceed as follows: after switch on, the current state is initialized at origin with an attitude matching the current acceleration direction and a zero speed, both with high variance. The filter is then ran using the high frequency magnetic update equation in order to get rapidly a stabilized trajectory from the first few seconds. This trajectory is used to bootstrap the first keyframe poses state and then to start using features information. This initialization process relies on the empirical observation that the MI-DR filter convergence basin is large. Admittedly, it would degrade severely the filter if the system's switch on occurs in an area where magneto-inertial dead-reckoning is not reliable.

5. Experimental Study

5.1. Hardware Prototype Description and Data Syncing

The sensor system is pictured on Figure 4. The camera is rigidly attached 47 cm away of the MIMU system to avoid any potential magnetic perturbation. Such a large distance is necessary because the off-the-shelf camera has not been specifically designed for reducing its magnetic footprint. Reducing the hardware to a wearable size would involve sensors co-design that was not in the scope of this work. For the vision part, we use an IDS uEye 3241-LE equipped with a Lensagon BM4018S118 lens. It provides around 100 degrees of field of view. Camera intrinsics and extrinsic parameters are calibrated with the Kalibr toolbox [23]. The MIMU provides data at 325 Hz, the camera at 20 Hz. Magnetometers, accelerometers and gyro-meters are all MEMS sensors digitized with sigma-delta ADCs which were carefully calibrated.

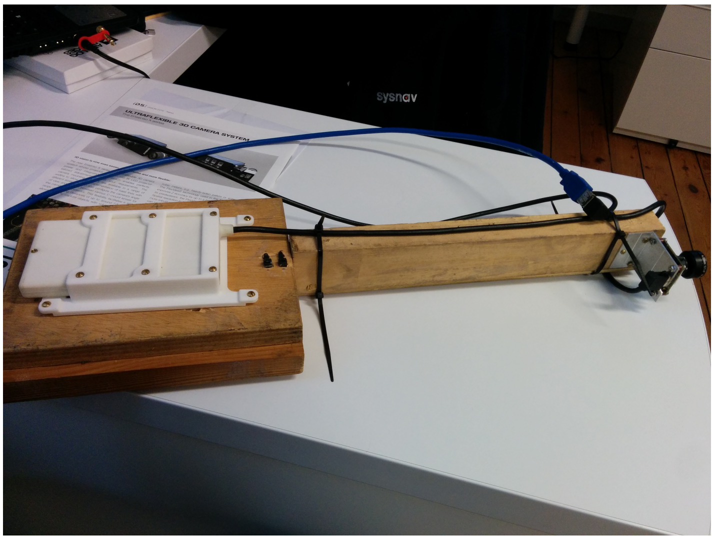


Figure 4. The sensor setup used in this work. The white box on the left side contains the MIMU sensor, the camera is on the right side. Both sensors are rigidly attached through a non-magnetic, non conductive material (wood).

The camera and MIMU provide timestamps computed from different clocks. We synchronize them offline: timestamps shifts are estimated both at start and at the end of the records by the checkerboard-based Kalibr calibration toolbox; clock drift is then deduced and corrected for.

The camera exposure time is fixed at 10 milliseconds maximum, reducing worst case motion blur and allowing to timestamp accordingly each image observation. However, this choice limits the camera's ability to adapt to a very low-light environment.

5.2. Filter Parameters Tuning

Most parameters of the filter are chosen in a deterministic and consistent fashion. MIMU noise standard deviation $\sigma_{\mathbf{a}}, \sigma_{\omega}, \sigma_{\mathbf{B}}$ and biases evolution parameters $\tau_{bg}, \tau_{ba}, \sigma_{bg;c}, \sigma_{ba;c}$ are derived from sensors characteristics measured empirically with an Allan standard deviation. The pixel reprojection noise σ_C is set to $\sqrt{2}$, which is the diagonal size of a pixel. Only the magnetic part of covariance $\mathbf{\Sigma}_{\nabla \mathbf{B}}$ is tuned empirically. It is set greater than what would derive from the covariance of magnetometers white noise, so as to absorb some modeling errors of the magnetic field, such as small non-stationarities or high values of the 2nd order spatial term. Note that parameter tuning is unchanged for all presented datasets in order to draw fair conclusions.

5.3. Visual Processing Implementation

The visual processing pipeline aims at constantly track 200 interest points well spread in the image. In order to enforce a good repartition of corners across the entire image, we use a bucket strategy. Harris-corner response is computed over the entire image and we retain only the strongest features in buckets for which the number of already tracked points is below a threshold. We use here a partition of the image in 6 × 8 buckets. Detected corners are tracked from frame to frame with OpenCV pyramidal KLT algorithm until either:

- they go out of the field of view;

- the tracking fails;
- they are classified as outliers;
- the frame where they were firstly detected is to be marginalized at next propagation step.

We ran a 2-point RANSAC algorithm between subsequent frames for outliers detection and rejection, using the relative orientation from the integrated gyroscope as rotation between the two frames.

An ended feature track is used as measurement, only if:

- it spans at least three poses;
- its initial triangulation did not exhibit any degeneracy;
- its re-projection error is below a threshold.

Contrarily to our previous implementation [13], this threshold is dynamically set with a χ^2 threshold. As a result, the criterion becomes looser when the estimated uncertainty of relative poses increases.

In order to make the visual pipeline more robust to dark areas, we normalize each input image by its averaged intensity before corner detection and tracking. Some very dark images then become usable for corner detection despite a significant increase of the photometric noise which affect them. Noise amplification leads to spurious features track, but these are most often correctly rejected by our outlier rejection strategy. Overall, we found that normalization significantly improves the performance of pure MSCKF VINS algorithm on our dataset. Some raw/normalized images are presented in Figure 5.

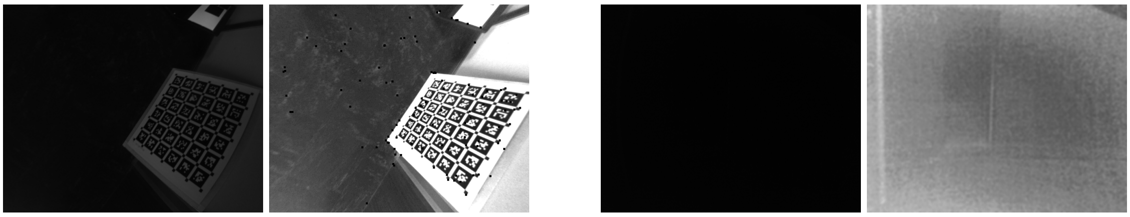


Figure 5. Image processing pipeline. (**Left**): Raw input images. (**Right**): after rectification, intensity normalization, and corner detection. On the second example, the normalization reveals a faint signal, but it is too noisy for corner detection to works.

Note that, in contrast with the tuning of the filter, the parameters of the vision pipeline (KLT window size, number of octaves in the pyramid of KLT, RANSAC threshold and minimum Harris score for corner detection) were chosen empirically.

5.4. Trajectory Evaluation

5.4.1. Dataset Presentation

We evaluate our algorithm on a dataset of five test trajectories. A pedestrian is carrying the system depicted in Figure 4 and walks through an industrial facility. Trajectories are specifically designed to be challenging for MI-DR and VINS: they are partly done outdoor, with very low magnetic field gradient and they also contains portions visually non informative made in the non lit basement of the building. Detailed results on Traj2 and Traj5 are presented on Figures 6–8, the others being more briefly depicted in Figure 9. In all cases, a dedicated plot indicates with a color code the parts of the trajectory where the magnetic field gradient vanishes and parts of the trajectories where the mean intensity is low.

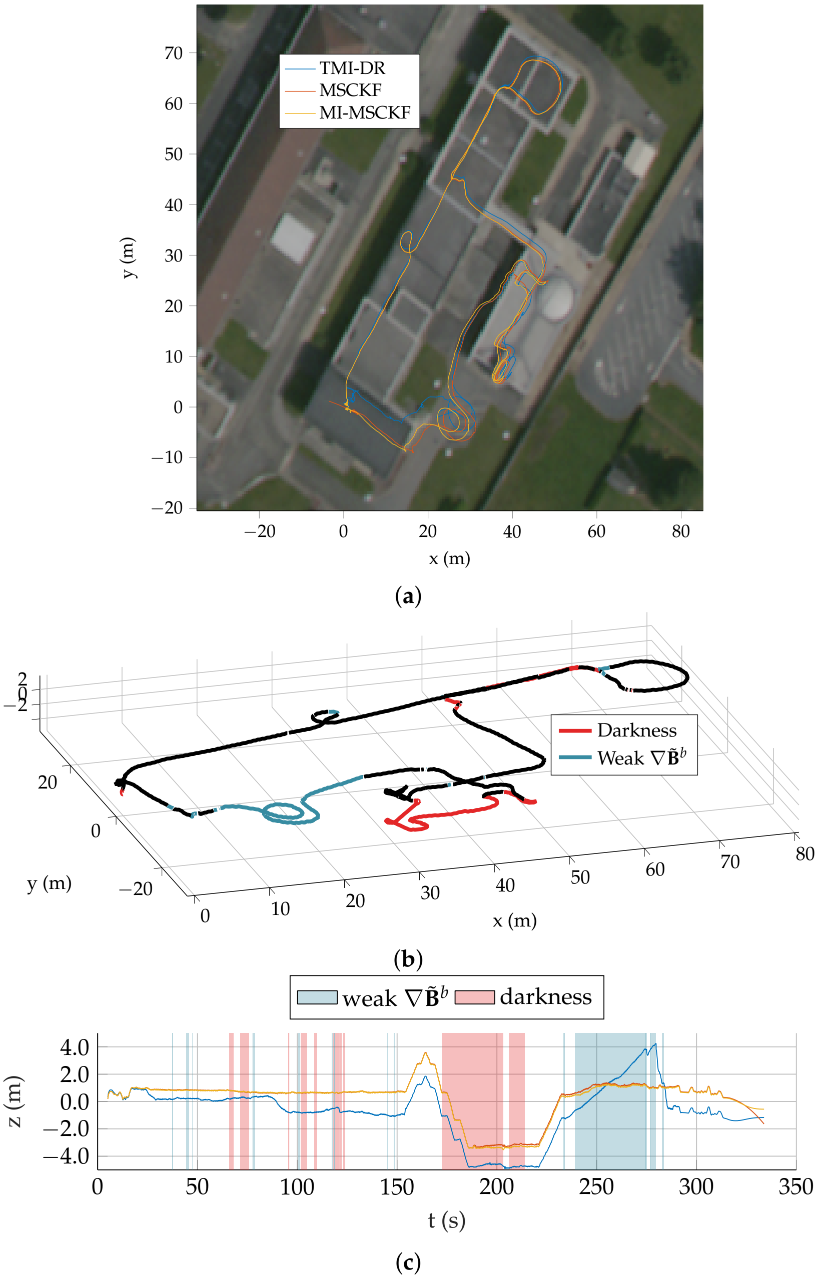


Figure 6. (**a**) Overview of trajectory Traj2 as reconstructed by the three filters. (**b**) Visualisation of dark areas and low-gradient areas over the entire trajectory surimposed on MI-MSCKF estimate. (**c**) Height profile of the three estimators on this trajectory.

5.4.2. Overall Comparison

Three estimators derived from the presented filter have been compared. The MI-DR is the presented filter without any visual update. MSCKF is the presented filter without any magnetic update. MI-MSCKF is the proposed filter fusing both information. Moreover, we also compare our results with the state of the art VINS filter of [24].

Unfortunately, we do not have access to a ground truth trajectory for our datasets. We then consider three evaluation criteria:

- the estimated trajectories are superimposed to a georeferenced orthoimage in which one pixel represents precisely 0.5 m. We compute an alignment of the trajectory when the checkerboard detected for the first time in the sequence. This alignement results from setting manually the *position and heading* of the checkerboard frame relative to the coordinates system of the satellite image. Note that no manual scale alignment has been made, hence this visualization allows to evaluate roughly the correctness of the global scale of the estimate, for instance on Figure 6a;

- the z profile is globally known as the pedestrian walks along flat corridors—except when he takes stairs to change levels;
- a translational error is computed each time the system comes back to its initial position, thanks to a static checkerboard placed at starting point. This criterion can be visualized in Figure 6c for Traj2 where it is clear the MI-DR estimate is less stable vertically over the entire trajectory.

The last criterion can be quantitavely evaluated: results are displayed in Table 1 with an error given in percentage of the trajectory length. Next sections emphasize some differences in the behaviors of the tested methods.

Table 1. Summary of final drift error on full dataset (% of trajectory length).

	Traj1	**Traj2**	**Traj3**	**Traj4**	**Traj5**
MI-DR	1.11	1.98	1.81	1.54	2.87
MSCKF (VINS)	0.33	0.63	0.59	1.05	0.21
MI-MSCKF	**0.20**	**0.31**	**0.49**	0.71	**0.15**
State of the art VINS [24]	0.26	0.52	0.79	**0.62**	0.20

5.4.3. The Fused Estimate Improves MI-DR in Outdoor Trajectories

The Figure 6 shows the three versions of our filter on Traj2. The trajectory estimated by MI-DR is very close to the others until some point in the outdoor part—note that the outdoor part corresponds to the weak gradient part of the trajectory as depicted on Figure 6b. During this outdoor part, as expected, the MI-DR drifts away compared to the two vision-based estimates which directly leads to a higher final translation error. The same effect is also clearly visible on Traj3, see Figure 9b.

5.4.4. Data Fusion Improves Local Consistency

By reducing the drift in dark areas or low gradient areas, the fused estimates improve the local position estimate consistency, an effect which is not always visible on the metrics of Table 1.

The benefit of magnetometry information in this sense is demonstrated in the details of results on the Traj2 displayed in Figure 7. The two left plots show a similar situation: the VINS estimate (red) is for a few seconds strongly corrected by the filter, leading to non continuous estimates (see green circles). The MI-MSCKF filter stays smoother during the entire trajectories, thanks to the speed observability provided by Equation (7). Interestingly, the pure VINS estimate joins the MI-MSCKF estimate later in the sequence, which makes the temporary drift mostly invisible in the final loop error metric of Table 1. This correction happens when visual information becomes available again. It means that the VINS filter is still able to correct itself after reasonable drift through the information stored in the prior.

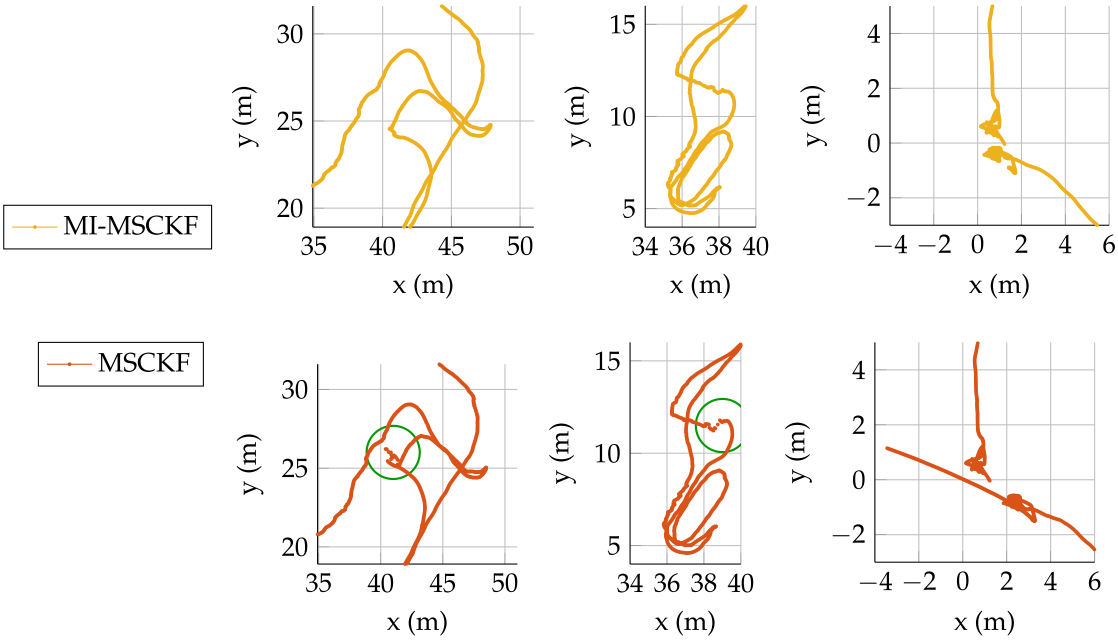


Figure 7. Details of estimation results on Traj2 showing a different behavior between the MSCKF and MI-MSCKF filter. **Left and middle plots**: while transitioning from dark area to lit environment some strong filter correction happen for the MSCKF and lead to discontinuities of the position estimate. In the same areas, MI-MSCKF stays smoother. **Right plot**: here the device is laid on the ground at the end of the trajectory. A large drift of MSCKF occurs, as visual information does not provide any feedback on position. Here again, the MI-MSCKF appears more stable.

The same effect occurs on all trajectories. Consider for instance the trajectory Traj5 depicted on Figure 8, where the lowest part goes also through the dark basement of the building. The MSCKF drifts vertically before being corrected when the pedestrian takes the stair up again. This effect is depicted on details in Figure 8c, which clarifies the evolution of the estimated height with respect to time.

In turn, visual information also helps trajectory consistency. It is clearly visible on the strong vertical drift of MI-DR displayd on Figure 6c around 250 s. Again, this drift is corrected as soon as magnetic gradients becomes sufficiently high to make speed observable again. Note the horizontal drift was never corrected though, leading to the larger final drift shown in Table 1.

The reader may have noticed at that point that Figure 8c shows that MI-DR fails badly on Traj5. Yet, we would like to stress that the fully fused estimate is still able to outperform the VINS estimate, leveraging magnetic information correctly beyond the breaking point of MI-DR. The reason of the failure of MI-DR is an unstationary local magnetic field in the first few seconds of Traj5. It perturbes the MI-DR initialization, which has dramatic consequence on all the rest of the trajectory.

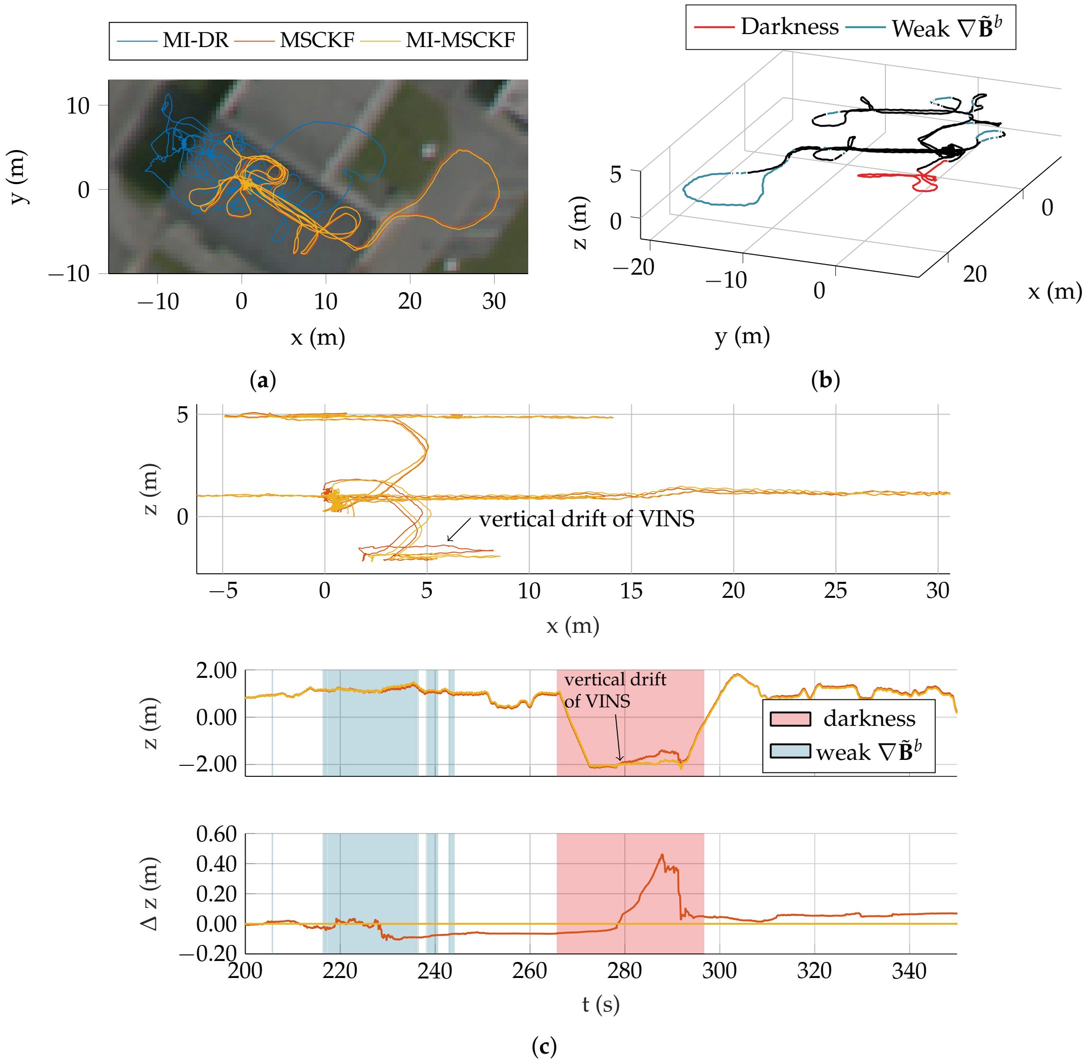
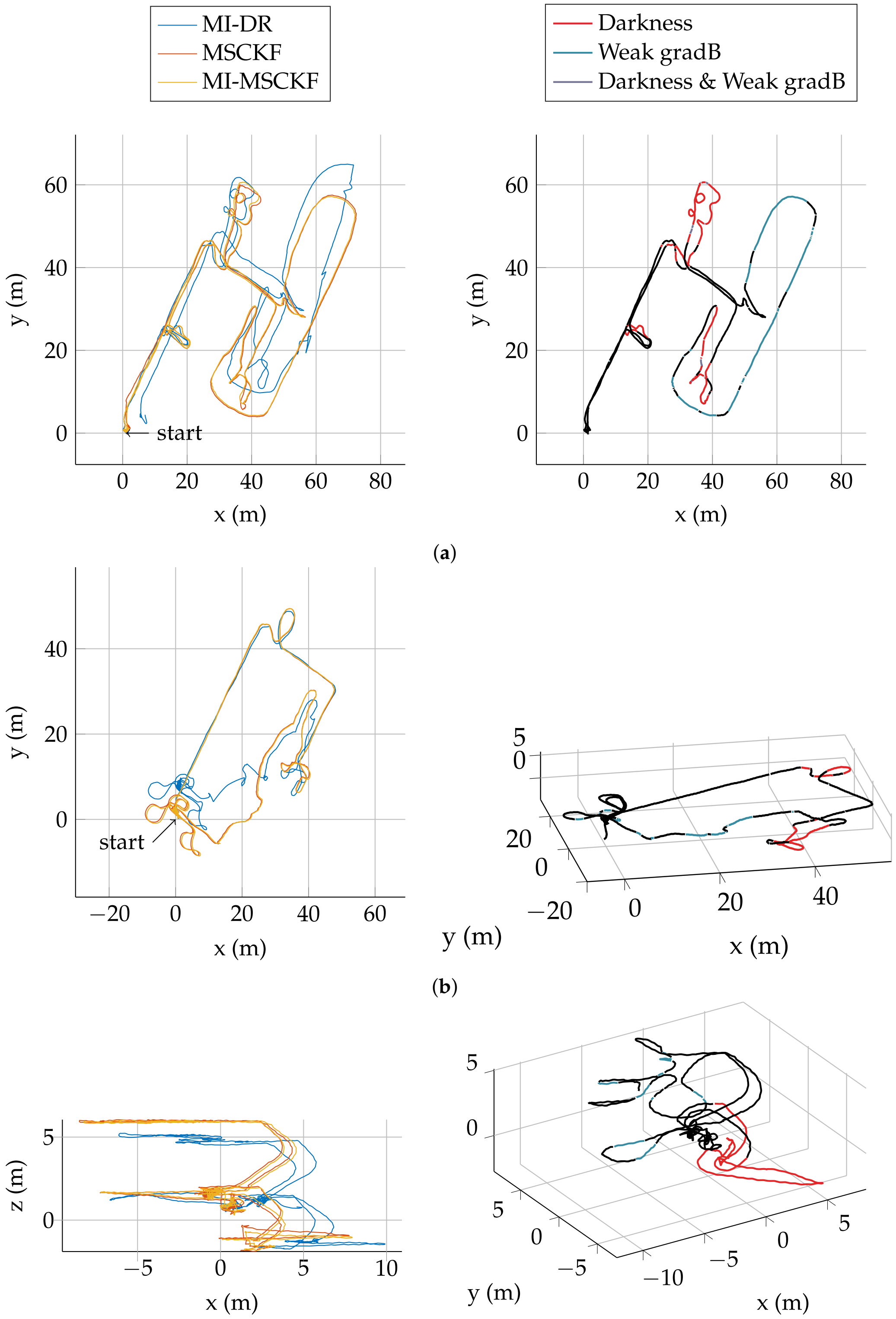


Figure 8. (**a**) Overview of trajectory Traj2 as reconstructed by the three filters. (**b**) Visualisation of dark areas and low-gradient areas over the entire trajectory surimposed on MI-MSCKF estimate. (**c**) Height profile of the three estimators on this trajectory.

The fact that the fused estimate prevents local drift could be, in our opinion, highly beneficial to the long-term performance in a Extended Kalman Filter strategy. Indeed, it could reduce overall linearization errors and the maximum magnitude of corrections, which are recognized, in VINS community, as an important drawback of filtering approaches compared to bundle adjustment or optimization-based methods.



(**a**)

(**b**)

(**c**)

Figure 9. Summary of trajectories on the remaining sequences of the dataset. (**Left**): Estimate of the three configuration of our filter. (**Right**) Color coded MI-MSCKF trajectory showing areas of weak gradient and weak illumination. (**a**) **Traj1** Length: ∼530 m; (**b**) **Traj3** Length: ∼368 m; (**c**) **Traj4** Length: ∼180 m.

5.4.5. Comparison with a State of the Art Filter

We also ran the released binary version (Available on https://github.com/MARSLab-UMN/MARS-VINS, we used the commit 8531daf.) of [24] on our dataset. Indeed, we think it is the state of the art in VINS filters for pedestrian navigation. As [24] takes only stereo images (even if their method works well with monocular setup also, as said in the paper) we had to trick slightly their software to use a monocular input. Note that, to be as fair as possible, we have entered in their code the same normalized monocular images we use. In doing so, we observed that normalization has also drastically improved the performance of their filter on our data.

An in-depth and comprehensive comparison between the two implementation is difficult as the code of [24] is not open. If inertial handling in both implementation should be close, the visual pipeline is very sensitive to parameters value and implementation details that are not known by us. Nevertheless, Table 1 demonstrates that both filters compare reasonably and are clearly below 1% of trajectory length error.

We also observed in [24] implementation the presence of strong filters correction after dark areas, as in described in Section 5.4.4. It indicates that this local consistency problem, which is essentially solved by the proposed fused estimate, is indeed a general issue of all VINS filters.

6. Conclusions

This work presented a filter to fuse information from a magnetometer array with other sensors traditionally used in VINS. We described in detail, the method we used and discussed its results on real datasets. Comparing the results of three estimators—one using only magnetic-inertial information, one using only visual-inertial information and one fusing both informations—we showed that the fused estimate leads to a more robust trajectory estimate. First, our fused estimate is able to reconstruct the trajectory outdoor where MI-DR techniques breaks because of the lack of gradient; secondly, our system avoids the unrealistic trajectory correction of VINS after significant duration without a good illumination. One trajectory also highlighted the need for either robust estimation technique or outlier rejection scheme for magnetic information. This should deserve more work in the future.

We think that the proposed approach could help to improve localization systems for augmented reality (AR) that are currently using VINS with consumer grade cameras and IMUs. Not only this combination extends the applicability domain of traditional VINS to degraded environments, but we also foresee opportunities to reduce power consumption. Indeed, taking advantages of the good trajectories given by the MI-DR in various conditions, it might be possible to reduce the computational load of the visual pipeline, which could be of major interest in practical applications.

Supplementary Materials: The following are available online at www.mdpi.com/1424-8220/17/12/2795/s1, Video Traj5.mp4 shows the real time results of MI-MSCKF filter on Traj5 of the dataset.

Acknowledgments: This research was performed within the framework of a CIFRE grant (ANRT contract #2014/0987) for the doctoral work of D.C. D.C. also thanks Charles-Yvan Chesneau for fruitful discussion and help with the MIMU sensors debugging.

Author Contributions: D.C. conducted this research during his doctoral work. D.C. designed the filter and did the implementation; D.C. and A.E. analyzed the resulting data; M.S. helped with the experimental part; D.V. designed the Magneto-inertial measurement unit; D.C., A.E., and G.B. wrote and revised the paper.

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

References

1. Zhang, J.; Kaess, M.; Singh, S. Real-Time Depth Enhanced Monocular Odometry. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), Chicago, IL, USA, 14–18 September 2014; pp. 4973–4980.
2. Guo, C.; Roumeliotis, S.I. IMU-RGBD Camera Navigation Using Point and Plane Features. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 3–7 November 2013; pp. 3164–3171.

3. Konolige, K.; Agrawal, M.; Sola, J. Large-Scale Visual Odometry for Rough Terrain. In *Robotics Research*; Springer: Berlin, Germany, 2011; pp. 201–212.
4. Leutenegger, S.; Lynen, S.; Bosse, M.; Siegwart, R.; Furgale, P. Keyframe-Based Visual—Inertial Odometry Using Nonlinear Optimization. *Int. J. Robot. Res.* **2015**, *34*, 314–334.
5. Usenko, V.; Engel, J.; Stückler, J.; Cremers, D. Direct Visual-Inertial Odometry with Stereo Cameras. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1885–1892.
6. Li, M.; Mourikis, A.I. High-Precision, Consistent EKF-Based Visual—Inertial Odometry. *Int. J. Robot. Res.* **2013**, *32*, 690–711.
7. Hernandez, J.; Tsotsos, K.; Soatto, S. Observability, Identifiability and Sensitivity of Vision-Aided Inertial Navigation. In Proceedings of the Robotics and 2015 IEEE International Conference on Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 2319–2325.
8. Dorveaux, E.; Boudot, T.; Hillion, M.; Petit, N. Combining Inertial Measurements and Distributed Magnetometry for Motion Estimation. In Proceedings of the American Control Conference (ACC), San Francisco, CA, USA, 29 June–1 July 2011; pp. 4249–4256.
9. Chesneau, C.I.; Hillion, M.; Prieur, C. Motion Estimation of a Rigid Body with an EKF Using Magneto-Inertial Measurements. In Proceedings of the IEEE 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Alcala de Henares, Spain, 4–7 October 2016; pp. 1–6.
10. Chesneau, C.I.; Hillion, M.; Hullo, J.F.; Thibault, G.; Prieur, C. Improving Magneto-Inertial Attitude and Position Estimation by Means of Magnetic Heading Observer. In Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan, 18–21 September 2017.
11. Caruso, D.; Sanfourche, M.; Le Besnerais, G.; Vissiere, D. Infrastructureless Indoor Navigation with an Hybrid Magneto-Inertial and Depth Sensor System. In Proceedings of the 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Alcala de Henares, Spain, 4–7 October 2016.
12. Caruso, D.; Eudes, A.; Sanfourche, M.; Vissiere, D.; Le Besnerais, G. Robust Indoor/Outdoor Navigation through Magneto-Visual-Inertial Optimization-Based Estimation. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017.
13. Caruso, D.; Eudes, A.; Sanfourche, M.; Vissiere, D.; Le Besnerais, G. An Inverse Square-Root Filter for Robust Indoor/Outdoor Magneto-Visual-Inertial Odometry. In Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan, 18–21 September 2017.
14. Wu, K.; Ahmed, A.; Georgiou, G.A.; Roumeliotis, S.I. A Square Root Inverse Filter for Efficient Vision-Aided Inertial Navigation on Mobile Devices. In Proceedings of the 2015 Robotics: Science and Systems Conference, Rome, Italy, 13–17 July 2015.
15. Anderson, B.; Moore, J. *Optimal Filtering*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1979.
16. Forster, C.; Carlone, L.; Dellaert, F.; Scaramuzza, D. On-Manifold Preintegration Theory for Fast and Accurate Visual-Inertial Navigation. *arXiv* **2015**, arXiv:1512.02363.
17. Dorveaux, E.; Vissiere, D.; Martin, A.P.; Petit, N. Iterative Calibration Method for Inertial and Magnetic Sensors. In Proceedings of the 48th IEEE Conference on Decision and Control, 2009 Held Jointly with the 2009 28th Chinese Control Conference, CDC/CCC 2009, Shanghai, China, 15–18 December 2009; pp. 8296–8303.
18. Hesch, J.A.; Kottas, D.G.; Bowman, S.L.; Roumeliotis, S.I. Camera-IMU-Based Localization: Observability Analysis and Consistency Improvement. *Int. J. Robot. Res.* **2014**, *33*, 182–201.
19. Simon, D. Kalman Filtering with State Constraints: A Survey of Linear and Nonlinear Algorithms. *IET Control Theory Appl.* **2010**, *4*, 1303–1318.
20. Mourikis, A.I.; Roumeliotis, S.I. A Multi-State Constraint Kalman Filter for Vision-Aided Inertial Navigation. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, 10–14 April 2007; pp. 3565–3572.
21. Yang, Z.; Shen, S. Monocular Visual-Inertial State Estimation With Online Initialization and Camera-IMU Extrinsic Calibration. *IEEE Trans. Autom. Sci. Eng.* **2017**, *14*, 39–51.
22. Dong-Si, T.C.; Mourikis, A. Estimator Initialization in Vision-Aided Inertial Navigation with Unknown Camera-IMU Calibration. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal, 7–12 October 2012; pp. 1064–1071.

23. Furgale, P.; Rehder, J.; Siegwart, R. Unified Temporal and Spatial Calibration for Multi-Sensor Systems. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 3–7 November 2013; pp. 1280–1286.
24. Paul, M.K.; Wu, K.; Hesch, J.A.; Nerurkar, E.D.; Roumeliotis, S.I. A Comparative Analysis of Tightly-Coupled Monocular, Binocular, and Stereo VINS. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 165–172.