Unscented Kalman Filtering on Lie Groups for Fusion of IMU and Monocular Vision

Martin Brossard, Silvère Bonnabel, Axel Barrau

Unscented Kalman Filtering on Lie Groups for Fusion of IMU and Monocular Vision

Martin BROSSARD*, Silvère BONNABEL* and Axel BARRAU†
*MINES ParisTech, PSL Research University, Centre for Robotics, 60 Boulevard Saint-Michel, 75006 Paris, France
†Safran Tech, Groupe Safran, Rue des Jeunes Bois-Châteaufort, 78772, Magny Les Hameaux Cedex, France

***Abstract*—Combining visual information with inertial measurements represents one popular approach to achieve robust and autonomous navigation in robotics, specifically for low-cost aerial vehicles in GPS-denied environments. In this paper, building upon both the recent theory of Unscented Kalman Filtering on Lie Groups (UKF-LG) and the theory of invariant Kalman filter based SLAM we proposed recently, an innovative UKF for the monocular visual inertial problem is derived, where the body pose, velocity, and the 3D landmarks' positions are viewed as a single element of a (high dimensional) Lie group $SE_{2+p}(3)$, and where accelerometers' and gyrometers' biases are appended to the state and estimated as well. The efficiency of the approach is validated on a real data set, where the proposed filter is shown to compare favorably to the conventional UKF, and to some recent UKF variants of the literature.**

***Index Terms*—Lie groups, unscented Kalman filter, visual inertial system, SLAM, sensor fusion, localization**

I. INTRODUCTION

Fusion of visual and inertial measurements, albeit an old field of research, is receiving increasing attention owing to the development of highly manoeuvring autonomous robots able to navigate in cluttered environments and solve tasks such as mapping or search and rescue [1,2]. The monocular Visual INertial System (VINS), which equips most of Micro Aerial Vehicles (MAVs), consists of an Inertial Measurement Unit (IMU) associated with a single camera, and constitutes an attractive sensor suite both for accurate state estimation and environment awareness, due to its low-cost, lightweight and power characteristics. Indeed, the IMU measures noisy biased accelerations and rotational velocities at high sampling time ($100 - 200\,\mathrm{Hz}$) whereas camera provides rich information for landmark visual tracking at lower rate ($20\,\mathrm{Hz}$).

Over the last decades, tremendous progresses have been achieved in visual localization frameworks (e.g. [3]), whose estimation and robustness can be improved by tightly coupling visual and inertial information. Most approaches combine data using filtering based solutions [4]–[8], or optimization/bundle adjustment techniques, e.g., [9,10]. Optimization based methods are more efficient but generally come with higher computational demands, and filtering approaches are well suited to real time.

In this paper, we tackle the problem of fusing IMU signals with monocular vision, also known as VINS or visual featured-based Simultaneous Localization And Mapping (visual SLAM) where the odometer information is replaced by inertial measurements. We propose a novel algorithm that mainly builds upon two components. First, the recent Lie group structure of SLAM advocated in the field of invariant filtering, see [11]–[13]. Secondly, the UKF on Lie Groups (UKF-LG), whose general methodology has been recently introduced in [14]. The effectiveness of our algorithm is tested on real data. The method favorably compares to the conventional UKF, as well as to some state of the art Kalman filter variants for the considered problem.

A. Links with Previous Literature and Contributions

In robotics, it has been long known that the Lie group structure of the space of poses $SE(3)$ plays an important role, see e.g., [15,16]. More recently, probability distributions on $SE(3)$, and their role for control and estimation, have been well studied, see e.g., [17]–[19], and the monographs [20,21].

In [11], we proposed the Right Invariant Extended Kalman Filter (RIEKF) based SLAM. Letting the output function $h(.)$ of [11] be the projection onto the camera frame, we readily obtain a RIEKF for visual SLAM, and this filter has been described and advocated recently in [7] for 3D VINS. However, in the implementation section of the latter paper, the authors rather opt for a multi-state constrained RIEKF, which is not the same as the RIEKF based SLAM, since in their RI-MSCKF the landmarks are removed from the state (see eq. 21 therein). In Section V dedicated to experiments, we also apply the framework [11] to the inertial and vision fusion, to implement a RIEKF where the landmarks are part of the state indeed. To our best knowledge this is the first published implementation of RIEKF based inertial visual SLAM and we demonstrate experimentally the efficiency of the approach, notably its superiority over the conventional UKF based visual inertial SLAM.

We also propose a Right-UKF-LG, which can be viewed as an unscented-based transform alternative to the RIEKF, but which has the advantage of being much more versatile than the RIEKF, though. Indeed, it spares the user the computation of Jacobians, that can prove tedious in the Invariant EKF framework (the Jacobians are defined with respect to the Lie groups structure, see e.g., [12]). As a result,

- the practitioner can *readily* implement our algorithm when using, e.g., a different camera model, or if one wants to add additional measurements such as GPS measurements outdoors, or a complementary depth sensor;
- should additional parameters/variables be estimated, such as IMU's scale factors and/or harmonization angles, the

algorithm is straightforward to adapt following the state augmentation technique of Section III-C3.

Moreover, the difference between the Right-UKF-LG and the RIEKF echoes the difference between conventional UKF and conventional EKF. In the experiments below, the noise level is moderate, and both methods perform identically. This is because under this level of noise the Jacobian approximation is accurate indeed. However in the presence of stronger noise, the unscented transform based method might prove more robust, as demonstrated on a robot localization problem in [14].

Note that, the present paper, even though concerned with a particular application, also brings some improvements to the general UKF-LG methodology, by presenting a state augmentation technique at Section III-C3 to deal with state spaces that are not exactly Lie groups, and a modification in Section IV-F to deal with large updates, as well as a detailed square-root form implementation in the Appendix.

In [8], the authors consider the same visual inertial fusion problem, and devise an UKF that takes advantage of the Lie group structure of the robot's (quadrotor) pose $SE(3)$. The main differences are threefold. First, the Lie group we use $SE_{2+p}(3)$, introduced in [11,22], is much bigger than $SE(3)$, and includes the pose but also the velocity and the landmarks' positions. Then, and more generally, the UKF-LG [14] generates sigma points in the Lie algebra, and then uses concentrated Gaussians (as in e.g., [18]) to map them onto the group. In contrast, [8] uses a probability distribution directly defined on the group (the distributions in [20]) to generate the sigma points, which is akin to the general unscented Kalman filtering on manifolds of [23]. Moreover, while [8] uses parallel transport operations based on left multiplications, we explore two variants based both on left and right multiplications, and the right one turns out to be better. Finally, experiments in Section V suggest the Right-UKF-LG achieves better performance than the $SE(3)$-based UKF of [8].

B. Paper's Organization

The paper is organized as follows. Section II formulates the fusion problem. Section III contains mathematical preliminaries on unscented Kalman filtering on Lie groups. Section IV describes the two proposed UKFs for monocular VINS. Section V illustrates the performances of the proposed filters based on real experiments. The Matlab code used for this paper is publicly available at `https://github.com/mbrossar/ICRA2018.git`.

II. Problem Modeling

We recall in this section the standard kynodynamic model for flying devices equipped with an IMU. We then detail the visual measurement model, and we finally pose the filtering (or more precisely the SLAM) problem we seek to address.

A. Variables of Interest and Dynamical Model

Let us consider an aerial body navigating on flat earth equipped with an IMU, whose biases are modeled as random walks. Assume moreover that p fixed landmarks of the scene can be tracked visually, and that they constitute the map. The state we want to estimate consists of the position $\mathbf{x} \in \mathbb{R}^3$, velocity $\mathbf{v} \in \mathbb{R}^3$, orientation $\mathbf{R} \in SO(3)$ of the body (that is, the rotation matrix that maps the body frame to the global frame), the IMU biases $\mathbf{b}_{\boldsymbol{\omega}} \in \mathbb{R}^3$ and $\mathbf{b}_{\mathbf{a}} \in \mathbb{R}^3$, as well as the 3D positions $\mathbf{p}_1, \dots, \mathbf{p}_p \in \mathbb{R}^3$ of the landmarks in the global frame. The dynamics of the system read

$$\text{body state } \begin{cases} \dot{\mathbf{R}} = \mathbf{R}\left(\boldsymbol{\omega} - \mathbf{b}_{\boldsymbol{\omega}} + \mathbf{n}_{\boldsymbol{\omega}}\right)_\times \\ \dot{\mathbf{v}} = \mathbf{R}\left(\mathbf{a} - \mathbf{b}_{\mathbf{a}} + \mathbf{n}_{\mathbf{a}}\right) + \mathbf{g}\,, \\ \dot{\mathbf{x}} = \mathbf{v} \end{cases} \tag{1}$$

$$\text{IMU biases } \begin{cases} \dot{\mathbf{b}}_{\boldsymbol{\omega}} = \mathbf{n}_{\mathbf{b}_{\boldsymbol{\omega}}} \\ \dot{\mathbf{b}}_{\mathbf{a}} = \mathbf{n}_{\mathbf{b}_{\mathbf{a}}} \end{cases}, \tag{2}$$

$$\text{landmarks } \left\{\dot{\mathbf{p}}_i = \mathbf{0},\ i = 1, \dots, p \right. \quad , \tag{3}$$

where the various white Gaussian noises can be stacked as

$$\mathbf{n} = \left[\mathbf{n}_{\boldsymbol{\omega}}^T\ \mathbf{n}_{\mathbf{a}}^T\ \mathbf{n}_{\mathbf{b}_{\boldsymbol{\omega}}}^T\ \mathbf{n}_{\mathbf{b}_{\mathbf{a}}}^T\right]^T \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}\right), \tag{4}$$

and where $(\boldsymbol{\omega})_\times$ denotes the skew symmetric matrix associated with the cross product with vector $\boldsymbol{\omega} \in \mathbb{R}^3$. These equations model the dynamics of small MAVs such as quadrotors where the IMU measurements $\boldsymbol{\omega}$ and $\mathbf{a}$ in (1) are considered as noisy and biased inputs of the system.

B. Measurement Model

In addition to the IMU measurements used as input for the dynamics, the vehicle gets visual information from a calibrated monocular camera, observes and tracks the p landmarks through a standard perspective projection model [24]. Landmark $\mathbf{p}_i$ is observed through the camera as

$$\mathbf{y}_i = \begin{bmatrix} y_u^i \\ y_v^i \end{bmatrix} + \mathbf{n}_{\mathbf{y}}^i, \tag{5}$$

where $\mathbf{y}_i$ is the measured pixel location of the landmark in the camera, that is,

$$\lambda \begin{bmatrix} y_u^i \\ y_v^i \\ 1 \end{bmatrix} = \boldsymbol{\Pi}\left[\mathbf{R}_C^T\left(\mathbf{R}^T\left(\mathbf{p}_i - \mathbf{x}\right) - \mathbf{x}_C\right)\right], \tag{6}$$

with λ the scale parameter, $\boldsymbol{\Pi}$ the calibration matrix of the pinhole camera model, and the term in bracket corresponds to the distance from the landmark to the camera expressed in the camera frame with $\mathbf{R}_C$ and $\mathbf{x}_C$ the known rotation matrix and translation mapping the body frame to the camera frame. Finally, $\mathbf{n}_{\mathbf{y}} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{N}\right)$ represents the pixel image noise.

C. Estimation/Fusion Problem

We would like to compute the probability distribution of the system's state $(\mathbf{x}, \mathbf{v}, \mathbf{R}, \mathbf{b}_{\boldsymbol{\omega}}, \mathbf{b}_{\mathbf{a}}, \mathbf{p}_1, \dots, \mathbf{p}_p)$ defined through an initial Gaussian prior and the probabilistic evolution model (1)-(3), *conditionally* on the measurements of the form (5) for $1 \leq i \leq p$. This is the standard probabilistic formulation of the visual 3D SLAM problem, when the robot is equipped with an IMU. The problem is also referred to as VINS, see e.g.,

[7], camera-aided navigation, fusion of inertia and vision, and sometimes visual inertial odometry [8].

III. Unscented Kalman Filtering on Lie Groups

In this section we provide the reader with the bare minimum about the UKF on Lie Groups (UKF-LG) introduced in [14].

A. Matrix Lie Groups

A matrix Lie group $G \subset \mathbb{R}^{N\times N}$ is a subset of square invertible matrices such that the following properties hold

$$\mathbf{I} \in G;\ \forall \boldsymbol{\chi} \in G, \boldsymbol{\chi}^{-1} \in G;\ \forall \boldsymbol{\chi}_1, \boldsymbol{\chi}_2 \in G, \boldsymbol{\chi}_1\boldsymbol{\chi}_2 \in G. \quad (7)$$

Locally about the identity matrix $\mathbf{I}$, the group G can be identified with an Euclidean space $\mathbb{R}^q$ using the matrix exponential map $\exp_m(.)$, where $q = \dim G$. Indeed, to any $\boldsymbol{\xi} \in \mathbb{R}^q$ one can associate a matrix $\boldsymbol{\xi}^\wedge$ of the tangent space of G at $\mathbf{I}$, called the Lie algebra $\mathfrak{g}$. We then define the exponential map $\exp : \mathbb{R}^q \to G$ for Lie groups as

$$\exp(\boldsymbol{\xi}) = \exp_m(\boldsymbol{\xi}^\wedge), \quad (8)$$

Locally, it is a bijection, and one can define the Lie logarithm map $\log : G \to \mathbb{R}^q$ as the exponential inverse, leading to

$$\log(\exp(\boldsymbol{\xi})) = \boldsymbol{\xi}. \quad (9)$$

B. Uncertainty on Lie Groups

To define random variables on Lie groups, we cannot apply the usual approach of additive noise for $\boldsymbol{\chi} \in G$ as G is not a vector space. In contrast, we define the probability distribution $\boldsymbol{\chi} \sim \mathcal{N}_L(\bar{\boldsymbol{\chi}}, \mathbf{P})$ for the random variable $\boldsymbol{\chi} \in G$ as [18,25]

$$\boldsymbol{\chi} = \bar{\boldsymbol{\chi}} \exp(\boldsymbol{\xi}),\ \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}), \quad (10)$$

where $\mathcal{N}(.,.)$ is the classical Gaussian distribution in Euclidean space $\mathbb{R}^q$ and $\mathbf{P} \in \mathbb{R}^{q\times q}$ is a covariance matrix. In (10), the original Gaussian $\boldsymbol{\xi}$ of the Lie algebra is moved over by left multiplication to be centered at $\bar{\boldsymbol{\chi}} \in G$, hence the letter L which stands for "left", this type of uncertainty being also referred to as left-equivariant [14]. We similarly define the distribution $\boldsymbol{\chi} \sim \mathcal{N}_R(\bar{\boldsymbol{\chi}}, \mathbf{P})$ for right multiplication of $\bar{\boldsymbol{\chi}}$, as

$$\boldsymbol{\chi} = \exp(\boldsymbol{\xi})\bar{\boldsymbol{\chi}},\ \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}). \quad (11)$$

In (10) and (11), $\bar{\boldsymbol{\chi}}$ may represent a large, noise-free and deterministic value, whereas $\mathbf{P}$ is the covariance of the small, noisy perturbation $\boldsymbol{\xi}$. We stress that we have defined these probability density functions directly in the vector space $\mathbb{R}^q$ such that both $\mathcal{N}_\mathsf{L}(.,.)$ and $\mathcal{N}_\mathsf{R}(.,.)$ are not Gaussian distributions.

Remark 1: defining random Gaussians on Lie groups through (10) and (11) is advocated notably in [18,25], and the corresponding distribution is sometimes referred to as concentrated Gaussian on Lie groups, see [26]. An alternative approach, introduced to our best knowledge in [20], and used in [19], consists of defining a (Gaussian) density directly on the group using the Haar measure. In the latter case, the group needs be unimodular, but such a requirement is in fact unnecessary to define the random variables (10) and (11).

C. Unscented Kalman Filtering on Lie Groups

By representing the state error as a variable $\boldsymbol{\xi}$ of the Lie algebra, we can build two alternative unscented filters for any state living in Lie groups following the methodology very recently introduced in [14]. Let us consider a discrete time dynamical system of the form

$$\boldsymbol{\chi}_{n+1} = f(\boldsymbol{\chi}_n, \mathbf{u}_n, \mathbf{w}_n), \quad (12)$$

where the state $\boldsymbol{\chi}_n$ lives in G, $\mathbf{u}_n$ is a known input variable and $\mathbf{w}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_n)$ is a white Gaussian noise, associated with generic discrete measurements of the form

$$\mathbf{y}_n = h(\boldsymbol{\chi}_n, \mathbf{v}_n), \quad (13)$$

where $\mathbf{v}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_n)$ is a white Gaussian noise. Essentially two different UKFs follow from the above uncertainty representation.

1) Left-UKF-LG: the state is modeled as $\boldsymbol{\chi}_n \sim \mathcal{N}_L(\bar{\boldsymbol{\chi}}_n, \mathbf{P}_n)$, that is, using the left-equivariant formulation (10) of the uncertainties. The mean state is thus encoded in $\bar{\boldsymbol{\chi}}_n$ and dispersion in $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_n)$. The sigma points are generated based on the $\boldsymbol{\xi}$ variables, and mapped to the group through the model (10). Note that, this is in slight contrast with [8,19], which generate sigma points through a distribution defined directly on the group. The filter consists of two steps along the lines of the conventional UKF: propagation and update, and compute estimates $\bar{\boldsymbol{\chi}}_n$ and $\mathbf{P}_n$ at each n.

2) Right-UKF-LG: the state is alternatively modeled as $\boldsymbol{\chi}_n \sim \mathcal{N}_R(\bar{\boldsymbol{\chi}}_n, \mathbf{P}_n)$, that is, using the right-equivariant formulation (11) of the uncertainties.

3) UKG-LG with state augmentation: the paper [14] is dedicated to the case where the state is a Lie group. However, when the state consists of a matrix $\boldsymbol{\chi}$ belonging to a Lie group, and an additional vector, say $\mathbf{b}$, the methodology can be readily extended by augmenting the state space: vector $\mathbf{b}$ is appended to the state which then becomes the couple $(\boldsymbol{\chi}, \mathbf{b})$. The Lie group variable $\boldsymbol{\chi}$ is treated through the UKF-LG, whereas the vector variable is treated through the conventional UKF methodology. This is the approach advocated below, where the vector variable correspond to the IMU biases.

D. The Special Euclidean Group $SE_{2+p}(3)$

As early noticed in [22], the SLAM problem bears a natural Lie group structure, through the group $SE_{1+p}(3)$, that is properly introduced and leveraged for SLAM in [11], where a Lie group based EKF (namely an Invariant EKF) is shown to resolve some well-known consistency issues of EKF based SLAM. Some other stochastic properties of the Invariant EKF based SLAM have also recently been proved in [13].

Any matrix $\boldsymbol{\chi} \in SE_{2+p}(3)$ is defined as

$$\boldsymbol{\chi} = \begin{bmatrix} \mathbf{R} & \mathbf{v} & \mathbf{x} & \mathbf{p}_1 & \cdots & \mathbf{p}_p \\ \mathbf{0}_{2+p\times 3} & & & \mathbf{I}_{p+2\times p+2} & & \end{bmatrix}, \quad (14)$$

The dimension of the group, and thus of the Lie algebra, is $3 + 3(2 + p)$. The uncertainties, defined as $\boldsymbol{\xi} =$

$\left[\boldsymbol{\xi}_{\mathbf{R}}^T \boldsymbol{\xi}_{\mathbf{v}}^T \boldsymbol{\xi}_{\mathbf{x}}^T \boldsymbol{\xi}_{\mathbf{p}_1}^T \cdots \boldsymbol{\xi}_{\mathbf{p}_p}^T\right]^T \in \mathbb{R}^{9+3p}$, are mapped to the Lie algebra through the transformation $\boldsymbol{\xi} \mapsto \boldsymbol{\xi}^\wedge$ defined as

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} (\boldsymbol{\xi}_{\mathbf{R}})_\times & \boldsymbol{\xi}_{\mathbf{v}} & \boldsymbol{\xi}_{\mathbf{x}} & \boldsymbol{\xi}_{\mathbf{p}_1} & \cdots & \boldsymbol{\xi}_{\mathbf{p}_p} \\ & & \mathbf{0}_{2+p\times 5+p} & & & \end{bmatrix}. \tag{15}$$

The closed form expressions for exponential, logarithm and Jacobian can be deduced along the lines of [11,25] and can be found in the Matlab source code for the present paper.

IV. Proposed Algorithms

To apply UKF on Lie groups methodology, the dynamics must first be linearized, and the state space must be (partly) embedded in a matrix Lie group.

A. Time Discretization

Equations (1) are standard navigation equations, and various methods exist to discretize them with respect to time. The most common choice for a small time step ΔT is to set $\mathbf{R}(t+\Delta T) = \mathbf{R}(t)\exp_m[(\boldsymbol{\omega}(t) - \mathbf{b}_{\boldsymbol{\omega}}(t))\Delta T + \mathrm{Cov}\,(\mathbf{n}_{\boldsymbol{\omega}})^{1/2} g\sqrt{\Delta T})_\times]$, with g a standard 3-dimensional Gaussian. The remaining equations of model (1)-(3) are discretized using the Euler-Maruyama method (see also [10]).

B. Lie Group Embedding

The state space can be partially embedded into a Lie group, by letting $\boldsymbol{\chi}_n$ be the matrix of the group $G = SE_{2+p}(3)$ that represents the state variables $(\mathbf{R}, \mathbf{v}, \mathbf{x}, \mathbf{p}_1, \cdots, \mathbf{p}_p)$ at time step n through representation (14). Using this embedding, the state at time n can in turn be represented as $(\boldsymbol{\chi}_n, \mathbf{b}_n)$, letting the bias vector be $\mathbf{b} = \left[\mathbf{b}_{\boldsymbol{\omega}}^T \ \mathbf{b}_{\mathbf{a}}^T\right]^T \in \mathbb{R}^6$. The dispersion on $\boldsymbol{\chi}_n$ can be encoded using the left uncertainty (10) or the right one (11), leading to two alternative filters (see Section III-C). In the following, we detail the Right-UKF-LG which adopts the right-equivariant uncertainties (11) of $\boldsymbol{\chi}_n$ and we leave to the reader the derivation of the Left-UKF-LG, based upon left-equivariant uncertainties (10).

C. Final Retained Model and Filter Architecture

Defining the input vector $\mathbf{u} = \left[\boldsymbol{\omega}^T \ \mathbf{a}^T\right]^T$, and gathering the results of the two preceding subsections, we obtain the following discrete time model associated to the Right-UKF-LG:

$$\text{state} \quad \begin{cases} \boldsymbol{\chi}_n = \exp(\boldsymbol{\xi})\,\bar{\boldsymbol{\chi}}_n \\ \mathbf{b}_n = \bar{\mathbf{b}}_n + \tilde{\mathbf{b}} \end{cases}, \begin{bmatrix} \boldsymbol{\xi} \\ \tilde{\mathbf{b}} \end{bmatrix} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_n), \tag{16}$$

$$\text{dynamics} \quad \left\{ \boldsymbol{\chi}_n, \mathbf{b}_n = f\left(\boldsymbol{\chi}_{n-1}, \mathbf{u}_n - \mathbf{b}_{n-1}, \mathbf{n}_n\right), \right. \tag{17}$$

$$\text{observations} \quad \begin{cases} \mathbf{Y}_n = \left[\mathbf{y}_1^T \cdots \mathbf{y}_p^T\right]^T := \mathbf{Y}\left(\boldsymbol{\chi}_n, \mathbf{w}_n\right) \\ \mathbf{y}_i \text{ given in (5)}, \ i = 1, \ldots, p \end{cases}, \tag{18}$$

such that $\left(\bar{\boldsymbol{\chi}}_n, \bar{\mathbf{b}}_n\right) \in \mathbb{R}^{(15+3p)}$ represents the mean (estimated) state at time n, $\mathbf{P}_n \in \mathbb{R}^{(15+3p)\times(15+3p)}$ is the covariance matrix that defines the state uncertainties $\left(\boldsymbol{\xi}, \tilde{\mathbf{b}}\right)$, and the vector $\mathbf{Y}_n$ contains the observations of the p landmarks with associated Gaussian noise $\mathbf{w}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{W})$. The filter consists of two steps: propagation and update; as shown in Algorithm 1. We detail these two main steps in the following with the formalism of square-root implementation [27] where $\mathbf{S}$ is the Cholesky decomposition of the covariance, sparing the computation of covariance matrices and being numerically more stable.

Remark 2: for the Left-UKF-LG, we define $\boldsymbol{\chi}_n = \bar{\boldsymbol{\chi}}_n \exp(\boldsymbol{\xi})$ and substitute it in (16). Albeit a seemingly minor difference, both filters may exhibit quite different behaviors (in [11] the RIEKF based SLAM is proved to come with consistency properties that the Left-IEKF does not actually possess).

Algorithm 1: Left and Right UKF on Lie groups

Input: $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S} = \mathrm{chol}\,(\mathbf{P}), \mathbf{u}, \mathbf{Q}, \mathbf{Y}, \mathbf{W}$;
1 $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S} \leftarrow$ `Propagation` $\left(\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S}, \mathbf{u}, \mathbf{Q}\right)$;
if *received measurement* **then**
2 $\quad \bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S} \leftarrow$ `Update` $\left(\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S}, \mathbf{Y}, \mathbf{W}\right)$;
Output: $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S}$;

D. Propagation Step

The propagation step is described in Algorithm 2 and operates as follow. The filter first computes the propagated mean state, and then the $2J$ sigma points obtained at lines 3 and 5 are propagated at lines 6-7. It is then convenient to view the propagated Cholesky factors $\mathbf{S}$ as an output of the function $\mathtt{qr}(.)$. Details are provided in Appendix along with the definitions of J and γ. Although more details on the methodology can be found in [14] (see also [18] regarding propagation), line 7 deserves a few explanations for the paper to be self contained. According to uncertainty model (11), dispersion around the mean is modeled as $\exp(\boldsymbol{\xi})\boldsymbol{\chi}$, so if $\bar{\boldsymbol{\chi}}$ denotes the propagated mean, and $\boldsymbol{\chi}_j$ denotes a propagated sigma point, then the corresponding sigma point in the Lie algebra is defined through $\exp(\boldsymbol{\xi}_j)\bar{\boldsymbol{\chi}} = \boldsymbol{\chi}_j$, i.e., $\boldsymbol{\xi}_j = \log\left(\boldsymbol{\chi}_j \bar{\boldsymbol{\chi}}^{-1}\right)$.

Algorithm 2: `Propagation` function for Right-UKF-LG

Input: $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S}, \mathbf{u}, \mathbf{Q}$;
1 $\mathbf{u} \leftarrow \mathbf{u} - \bar{\mathbf{b}}$; `// unbiased input`
2 $\mathbf{S}^a = \mathrm{blkdiag}\,(\mathbf{S}, \mathrm{chol}\,(\mathbf{Q}))$;
3 $\boldsymbol{\chi} = \bar{\boldsymbol{\chi}}$; `// save non propagated state`
4 $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}} = f\left(\boldsymbol{\chi}, \mathbf{u}, \mathbf{0}\right)$; `// propagate mean`
`// step 5: sigma points generation`
5 $\left[\boldsymbol{\xi}_j^\mp \ \mathbf{b}_j^\mp \ \mathbf{n}_j^\mp\right] = \mp\gamma\,\mathrm{col}_j\,(\mathbf{S}^a)\,, \ j = 1, \ldots, J;$
`// steps 6-7: sigma point propagation`
6 $\boldsymbol{\chi}_j^\mp, \mathbf{b}_j^\mp \leftarrow f\left(\exp\left(\boldsymbol{\xi}_j^\mp\right)\boldsymbol{\chi}, \mathbf{u} - \mathbf{b}_j^\mp, \mathbf{n}_j^\mp\right), \ j = 1, \ldots, J;$
7 $\boldsymbol{\xi}_j^\mp \leftarrow \log\left(\boldsymbol{\chi}_j^\mp \bar{\boldsymbol{\chi}}^{-1}\right), \ j = 1, \ldots, J$;
8 $\mathbf{S} \leftarrow \mathtt{qr}\left(\boldsymbol{\xi}_j^\mp, \mathbf{b}_j^\mp, j = 1, \ldots, J, \mathbf{Q}\right)$;
`// see Appendix for definition of qr`
`// the notation` $\mathbf{x}^\mp$ `is used to to denote the two variables` $+\mathbf{x}$ `and` $-\mathbf{x}$
Output: $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S}$;

Algorithm 3: `Update` function for the Right-UKF-LG

Input: $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S}, \mathbf{Y}, \mathbf{W}$;

1 $\mathbf{S}^a = \text{blkdiag}\left(\mathbf{S}, \text{chol}\left(\mathbf{W}\right)\right)$;

2 $\mathbf{Y}_0 = \mathbf{Y}\left(\bar{\boldsymbol{\chi}}, \mathbf{0}\right)$; `// see (18) and (5)-(6)`

3 $\left[\boldsymbol{\xi}_j^{\mp} \ \mathbf{b}_j^{\mp} \ \mathbf{w}_j^{\mp}\right] = \mp\gamma \, \text{col}_j\left(\mathbf{S}^a\right), \ j = 1, \ldots, J'$;

4 $\boldsymbol{\chi}_j^{\mp} = \exp\left(\boldsymbol{\xi}_j^{\mp}\right) \bar{\boldsymbol{\chi}}, \ j = 1, \ldots, J'$;

5 $\mathbf{Y}_j = \mathbf{Y}\left(\boldsymbol{\chi}_j^{\mp}, \mathbf{w}_j^{\mp}\right), \ j = 1, \ldots, J'$;

6 $\delta\bar{\boldsymbol{\xi}}, \delta\bar{\mathbf{b}}, \mathbf{S} \leftarrow \mathtt{qr}'\left(\mathbf{Y}_0, \mathbf{Y}_j^{\mp}, \boldsymbol{\xi}_j^{\mp}, j = 1, \ldots, J', \mathbf{W}\right)$;

7 $\bar{\boldsymbol{\chi}} \leftarrow \exp\left(\delta\bar{\boldsymbol{\xi}}\right) \bar{\boldsymbol{\chi}}, \bar{\mathbf{b}} \leftarrow \bar{\mathbf{b}} + \delta\bar{\mathbf{b}}$; `// update mean`

`// See Appendix for definition of qr'`

Output: $\bar{\boldsymbol{\chi}}, \bar{\mathbf{b}}, \mathbf{S}$;

E. Update Step

The update step incorporates the observation of the p landmarks at time n and is described in Algorithm 3. It operates as follow. The sigma points generated in the Lie algebra at lines 3 are mapped to the group through model (11) at line 4, and used to compute $2J' + 1$ measurement sigma points at line 5. The function $\mathtt{qr}'(.)$ then evaluates the updated Cholesky factors and the innovation term $\left(\delta\bar{\boldsymbol{\xi}}, \delta\bar{\mathbf{b}}\right)$ used to update the mean state, along the lines of the conventional UKF methodology, and it is detailed in the Appendix. Line 7 is the update of [14], see next subsection for more details.

Remark 3: following [28], the square-root implementation can add or remove landmarks, initializing landmark position as inverse depth point and allows computationally efficient propagation steps. Sigma point selections [5] is also an interesting issue left for future work.

F. Discussion on the Final Update

Let $\bar{\boldsymbol{\chi}}$ denote the *propagated* mean, and $\mathbf{P}$ the *propagated* covariance matrix of the state error. According to model (11), it means the propagated state is approximately described by $\boldsymbol{\chi} \approx \exp\left(\boldsymbol{\xi}\right) \bar{\boldsymbol{\chi}}, \ \boldsymbol{\xi} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{P}\right)$. The unscented transform and the Kalman update then provide an approximation to the *posterior* for $\boldsymbol{\xi}$ of the form $\boldsymbol{\xi} \sim \mathcal{N}(\delta\bar{\boldsymbol{\xi}}, \mathbf{P}^+)$, i.e., $\boldsymbol{\xi} = \delta\bar{\boldsymbol{\xi}} + \boldsymbol{\xi}^+$ with $\boldsymbol{\xi}^+ \sim \mathcal{N}(\mathbf{0}, \mathbf{P}^+)$. Back to the Lie group this implies

$$\boldsymbol{\chi} \approx \exp\left(\boldsymbol{\xi}^+ + \delta\bar{\boldsymbol{\xi}}\right) \bar{\boldsymbol{\chi}}, \text{ where } \boldsymbol{\xi}^+ \sim \mathcal{N}\left(\mathbf{0}, \mathbf{P}^+\right), \tag{19}$$

Both correction $\delta\bar{\boldsymbol{\xi}}$ and uncertainty $\boldsymbol{\xi}^+$ in (19) being generally sufficiently small, the Baker-Campbell-Hausdorff (BCH)

$$\exp\left(\boldsymbol{\xi}^+ + \delta\bar{\boldsymbol{\xi}}\right) = \exp\left(\boldsymbol{\xi}^+\right) \exp\left(\delta\bar{\boldsymbol{\xi}}\right) + O\left(\delta\bar{\boldsymbol{\xi}}^+, (\boldsymbol{\xi}^+)^2, \boldsymbol{\xi}^+\delta\bar{\boldsymbol{\xi}}\right)$$

is sufficiently accurate to approximate the posterior as $\boldsymbol{\chi} \simeq \exp\left(\boldsymbol{\xi}^+\right) \bar{\boldsymbol{\chi}}^+$, where $\bar{\boldsymbol{\chi}}^+ = \exp\left(\delta\bar{\boldsymbol{\xi}}\right) \bar{\boldsymbol{\chi}}$, recovering the form imposed by model (11). This is what is proposed in [14].

Alternatively, when the innovation $\delta\bar{\boldsymbol{\xi}}$ is important, we propose in the present paper to possibly use the more accurate approximation of [20,29]

$$\exp\left(\boldsymbol{\xi}^+ + \delta\bar{\boldsymbol{\xi}}\right) = \exp\left(\mathbf{J}\boldsymbol{\xi}^+\right) \exp\left(\delta\bar{\boldsymbol{\xi}}\right) + O\left(\boldsymbol{\xi}^+\right), \tag{20}$$

where $\mathbf{J}$ is the left Jacobian. In this case we compute the updated parameters as

$$\bar{\boldsymbol{\chi}}^+ = \exp\left(\delta\bar{\boldsymbol{\xi}}\right) \bar{\boldsymbol{\chi}}, \tag{21}$$

$$\mathbf{P}^+ \leftarrow \mathbf{J}\mathbf{P}^+\mathbf{J}^T, \tag{22}$$

When $\delta\bar{\boldsymbol{\xi}}$ remains small, $\mathbf{J} \approx \mathbf{I}$ such that we can discard $\mathbf{J}$ in (22) for computational efficiency, recovering the update [14].

V. Experimental Results

To demonstrate the performances of the two proposed filters (Right and Left UKF-LG), we evaluate and compare them using Euroc public dataset [30] which collects real IMU and camera measurements of a quadrotor in indoor environments with available ground truth, and we select the sequence `V2_01_medium` (see Figures 1 and 2) where the flight dynamics and illumination conditions are judged as moderately complex. Five different filters are compared:

- an UKF that considers the attitude as an element $SO(3)$ and the remaining variables as a vector space;
- the SE(3)-based UKF recently introduced in [8]. This filter is an UKF which encodes body attitude and position in $SE(3)$ and uses parallel transport associated to left-invariant vector fields of $SE(3)$;
- the Right-Invariant EKF SLAM of [7,11];
- the proposed Right-UKF-LG described in Section IV;
- the proposed Left-UKF-LG, as an alternative to Right-UKF-LG based on the left uncertainty representation (10).

A. Experimental Setting

The different filters are configured with the same parameters, where we set $p = 30$ landmarks, IMU noise parameters provided by [30] and 1 pixel standard deviation for the landmark observations. The initial body pose corresponds to the ground truth values and we compute an initialization process based on [9] for IMU bias estimation and scale (i.e., landmarks) initialization. The filters obtain the same visual measurements by tracking features as in Figure 2 via the KLT tracker using minimum eigenvalue feature detection [31]. When a landmark is ceased to be observed, it is replaced with a new one which is initialized following [5]. Source code from our experiments is publicly available at `https://github.com/mbrossar/ICRA2018.git`.

B. Results

The different filters are thus launched on the real data and we plot the different state errors, respectively, in Figure 3 for the attitude, and in Figure 4 for the position. Performances over the whole experiment are numerically summarized in Figure 5. On this particular experiment, we see that

- the filters are rather equivalent regarding position estimate, but behave differently regarding attitude estimate. This is logical since orientation uncertainty is a key feature in SLAM inconsistency [32], and invariant Kalman filtering based methods precisely aim at better coping with non-linearities to improve consistency [11];

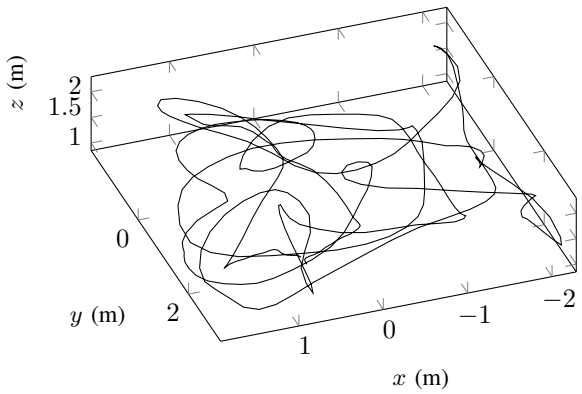

Fig. 1. Ground truth trajectory provided by Euroc dataset [30].

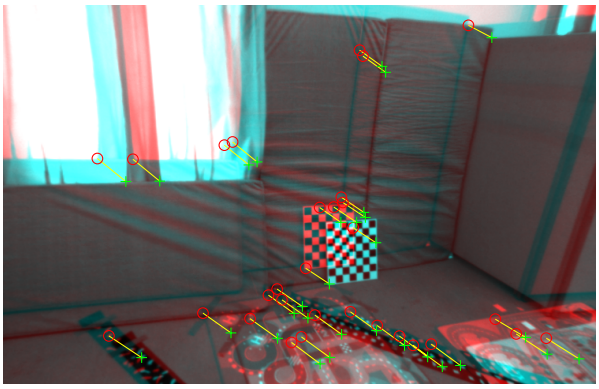

Fig. 2. Landmark tracking in the experiment. Green crosses are the current pixel locations of the landmarks and red circles are the pixel locations of the landmarks five images (i.e., 1 s) earlier. Picture comes from the Euroc dataset [30].

- various groups appear: the RIEKF and Right-UKF-LG achieve the best results. This is logical as right-invariant errors on $SE_{p+2}(3)$ are best suited to SLAM, as explained in [11,13]. The Left-UKF-LG achieves less accurate performances but is better than conventional UKF and the SE(3)-based UKF of [8];
- in terms of execution time, we indicate that our RIEKF implementation is faster than the remaining filters (which all exhibit similar complexity);
- due to the selection of 30 random landmarks at the beginning of the experiment, when running our code at `https://github.com/mbrossar/ICRA2018.git`, slight modifications can be observed, although the main tendencies remain unchanged. Notably, RIEKF and Right-UKF-LG are always the best, and in turn the Left-UKF-LG is always better than the conventional UKF, and we observed that the performances of the SE(3)-based UKF of [8] oscillates between those of Left-UKF-LG and conventional UKF. Thus, one can safely claim the SE(3)-based UKF is better than conventional UKF, although this is not obvious in the present particular run.

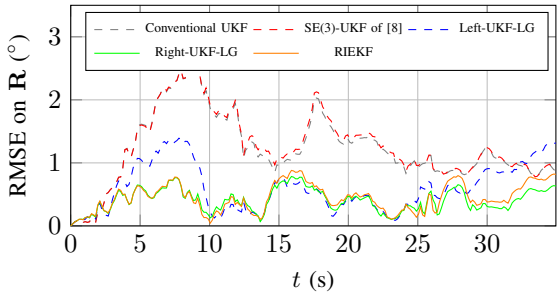

Fig. 3. Root Mean Square Error on attitude over time, based on real data and ground truth. RIEKF and Right-UKF-LG outperform the other filters on this dataset. Figure best seen in color.

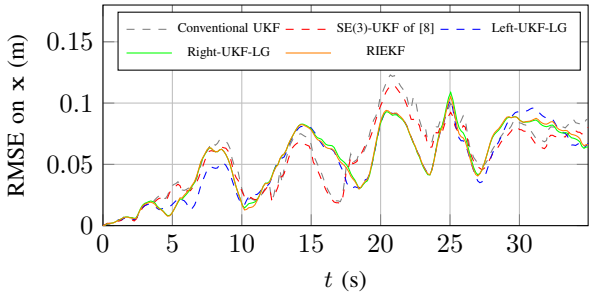

Fig. 4. Root Mean Square Error on position over time. The various filters obtain similar errors. Figure best seen in color.

VI. Conclusion

Two novel UKFs for data fusion of inertial sensors and monocular vision have been proposed. They build upon the very recent theory of UKF on Lie groups of [14], and have the merit of exploiting the full Lie group structure underlying the SLAM problem introduced in [11,22]. As such they can directly feed controllers that operate in the same space. Another advantage is that the UKF approach spares the user the computation of Jacobians inherent to EKF implementation, and thus the proposed filters can be readily adapted to small modifications in the model, estimation of additional parameters, and/or addition of one or several sensors. Results from real experimental data have shown the relevance of the approach based on invariance, and notably the Right-UKF-LG proposed here. Future works will focus on extending the approach including keyframes and descriptors in a complete (initialization, tracking, mapping, relocalization,

	$\mathbf{x}$ (cm)	$\mathbf{R}$ (°)
Conventional UKF	6.3	0.45
SE(3)-UKF of [8]	6.0	0.44
RIEKF	5.9	0.17
Left-UKF-LG	6.0	0.24
Right-UKF-LG	5.9	0.15

Fig. 5. Root Mean Square Error with respect to ground truth over the complete trajectory, on the body position and attitude, for the different filters. The proposed Right-UKF-LG achieves the best results.

and loop closing) Visual INertial System, and will explore the theoretical consistency properties that the proposed Right-UKF-LG might possess along the lines of [11].

REFERENCES

[1] M. Faessler, F. Fontana, C. Forster, E. Mueggler, M. Pizzoli, and D. Scaramuzza, "Autonomous, vision-based flight and live dense 3D mapping with a quadrotor Micro Aerial Vehicle," *Journal of Field Robotics*, vol. 33, no. 4, pp. 431–450, 2016.

[2] N. Michael, S. Shen, K. Mohta, Y. Mulgaonkar, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida, *et al.*, "Collaborative mapping of an earthquake-damaged building via ground and aerial robots," *Journal of Field Robotics*, vol. 29, no. 5, pp. 832–841, 2012.

[3] R. Mur-Artal, J. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[4] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Observability-constrained vision-aided inertial navigation," *University of Minnesota, Dept. of Comp. Sci. & Eng., MARS Lab, Tech. Rep*, vol. 1, 2012.

[5] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "A quadratic-complexity observability-constrained unscented Kalman filter for SLAM," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1226–1243, 2013.

[6] M. Li and A. I. Mourikis, "High-precision, consistent EKF-based visual-inertial odometry," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, 2013.

[7] K. Wu, T. Zhang, D. Su, S. Huang, and G. Dissanayake, "An Invariant-EKF VINS algorithm for improving consistency," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[8] G. Loianno, M. Watterson, and V. Kumar, "Visual inertial odometry for quadrotors on SE(3)," in *IEEE International Conference On Robotics and Automation (ICRA)*, pp. 1544–1551, IEEE, 2016.

[9] R. Mur-Artal and J. D. Tardos, "Visual-Inertial monocular SLAM with map reuse," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 796–803, 2017.

[10] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2017.

[11] A. Barrau and S. Bonnabel, "An EKF-SLAM algorithm with consistency properties," *Conditionally accepted in IEEE Transactions on Robotics. Preprint arXiv:1510.06263*, 2015.

[12] A. Barrau and S. Bonnabel, "Invariant Kalman filtering," *Annual Reviews of Control, Robotics, and Autonomous Systems (in press)*, vol. 1, 2018.

[13] T. Zhang, K. Wu, J. Song, S. Huang, and G. Dissanayake, "Convergence and consistency analysis for a 3-D Invariant-EKF SLAM," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 733–740, 2017.

[14] M. Brossard, S. Bonnabel, and J.-P. Condomines, "Unscented Kalman filtering on Lie groups," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[15] F. C. Park, J. E. Bobrow, and S. R. Ploen, "A Lie group formulation of robot dynamics," *The International Journal of Robotics Research*, vol. 14, no. 6, pp. 609–618, 1995.

[16] F. Bullo and A. D. Lewis, *Geometric control of mechanical systems: modeling, analysis, and design for simple mechanical control systems*, vol. 49. Springer Science & Business Media, 2004.

[17] Y. Wang and G. S. Chirikjian, "Error propagation on the euclidean group with applications to manipulator kinematics," *IEEE Transactions on Robotics*, vol. 22, pp. 591–602, Aug 2006.

[18] T. D. Barfoot and P. T. Furgale, "Associating uncertainty with three-dimensional poses for use in estimation problems," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 679–693, 2014.

[19] M. Zefran, V. Kumar, and C. Croke, "Metrics and connections for rigid-body kinematics," *International Journal of Robotics Research*, vol. 18, no. 2, pp. 243–258, 1999.

[20] G. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications*. Applied and Numerical Harmonic Analysis, Birkhäuser Boston, 2011.

[21] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017.

[22] S. Bonnabel, "Symmetries in observer design: Review of some recent results and applications to ekf-based slam," in *Robot Motion and Control*, pp. 3–15, Springer, 2011.

[23] S. Hauberg, F. Lauze, and K. S. Pedersen, "Unscented kalman filtering on riemannian manifolds," *Journal of mathematical imaging and vision*, vol. 46, no. 1, pp. 103–120, 2013.

[24] D. Forsyth and J. Ponce, *Computer vision: a modern approach*. Upper Saddle River, NJ; London: Prentice Hall, 2011.

[25] A. Barrau and S. Bonnabel, "The Invariant Extended Kalman Filter as a stable observer," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2017.

[26] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, "Continuous-discrete extended Kalman filter on matrix lie groups using concentrated Gaussian distributions," *Journal of Mathematical Imaging and Vision*, vol. 51, no. 1, pp. 209–228, 2015.

[27] R. Van Der Merwe and E. A. Wan, "The square-root unscented Kalman filter for state and parameter-estimation," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 6, pp. 3461–3464, 2001.

[28] S. A. Holmes, G. Klein, and D. W. Murray, "An $O(N^2)$ square root unscented Kalman filter for visual simultaneous localization and mapping," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 7, pp. 1251–1263, 2009.

[29] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017.

[30] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. Achtelik, and R. Siegwart, "The EuRoC MAV datasets," *The International Journal of Robotics Research*, 2015.

[31] J. Shi *et al.*, "Good features to track," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 593–600, IEEE, 1994.

[32] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the EKF-SLAM algorithm," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, p. 35623568, 2006.

[33] S. J. Julier, "The scaled unscented transformation," in *Proceedings of the American Control Conference*, vol. 6, pp. 4555–4559, IEEE, 2002.

APPENDIX

We give here the details of parameters and functions used for LeF and RiF. We set the scale parameters γ and γ' with the scaled unscented transform [33], such that they depend on the augmented covariance size $J = 27 + 3p$, $J' = 15 + 5p$ and are defined as

$$\gamma = \sqrt{J/(1+W_0)}, W_0 = 1 - J/3, W_j = \frac{1-W_0}{2J}, \quad (23)$$

$$\gamma' = \sqrt{J'/(1+W'_0)}, W'_0 = 1 - J'/3, W'_j = \frac{1-W'_0}{2J'}. \quad (24)$$

The function $\mathtt{qr}(.)$ operates as taking the $\mathcal{QR}$ decomposition of

$$\mathcal{QR} \leftarrow \begin{bmatrix} \boldsymbol{\xi}_1^+ & \cdots & \boldsymbol{\xi}_J^+ & \boldsymbol{\xi}_1^- & \cdots & \boldsymbol{\xi}_J^- \\ \mathbf{b}_1^+ & \cdots & \mathbf{b}_J^+ & \mathbf{b}_1^- & \cdots & \mathbf{b}_J^- \\ \mathbf{0} & \mathrm{chol}\left(\mathbf{Q}\right) & \mathbf{0} & -\mathrm{chol}\left(\mathbf{Q}\right) \end{bmatrix}, \quad (25)$$

from which we can extract the Cholesky factor as

$$\mathcal{R} = \begin{bmatrix} \mathbf{S} \\ \mathbf{0} \end{bmatrix}. \quad (26)$$

The function $\mathtt{qr}'(.)$ operates as follow: first, compute the mean measurement and weighted deviation

$$\bar{\mathbf{Y}} = W_0 \mathbf{Y}_0 + \sum_{j=1}^{J'} W'_i \left(\mathbf{Y}_j^+ + \mathbf{Y}_j^-\right), \quad (27)$$

$$\mathbf{e}_0 = \sqrt{|W'_0|}\left(\mathbf{Y}_0 - \bar{\mathbf{Y}}\right), \quad (28)$$

$$\mathbf{e}_j^{\mp} = \sqrt{|W'_j|}\left(\mathbf{Y}_i^{\mp} - \bar{\mathbf{Y}}\right), \ j = 1, \ldots, J', \quad (29)$$

and compute the Cholesky factors of the measurement covariance and the cross covariance as

$$\mathcal{QR} \leftarrow \begin{bmatrix} \mathbf{e}_1^+ & \cdots & \mathbf{e}_{J'}^+ & \mathbf{e}_1^- & \cdots & \mathbf{e}_{J'}^- \\ \mathbf{0} & \mathrm{chol}\left(\mathbf{W}\right) & & \mathbf{0} & -\mathrm{chol}\left(\mathbf{W}\right) & \end{bmatrix}, \tag{30}$$

$$\mathcal{R} = \begin{bmatrix} \mathbf{S}' \\ \mathbf{0} \end{bmatrix}, \tag{31}$$

$$\mathbf{S}' \leftarrow \texttt{CholUpdate}\left(\mathbf{S}', \mathrm{sign}\left(W_0'\right), \mathbf{e}_0\right), \tag{32}$$

$$\mathbf{P}' = \sum_{j=1}^{J'} \sqrt{W_j'} \left(\begin{bmatrix} \boldsymbol{\xi}_j^+ \\ \mathbf{b}_j^+ \end{bmatrix}^T \mathbf{e}_j^+ + \begin{bmatrix} \boldsymbol{\xi}_j^- \\ \mathbf{b}_j^- \end{bmatrix}^T \mathbf{e}_j^- \right), \tag{33}$$

where $\mathbf{W}$ is a block diagonal matrix containing p times the matrix $\mathbf{N}$ along its diagonal, and then compute gain, innovation and covariance as

$$\mathbf{K} = \mathbf{P}' \left(\mathbf{S}'^T \mathbf{S}' \right) = \mathbf{P}\mathbf{S}'^{-1}\mathbf{S}'^{-T}, \tag{34}$$

$$\begin{bmatrix} \delta\bar{\boldsymbol{\xi}} \\ \delta\bar{\mathbf{b}} \end{bmatrix} = \mathbf{K}\left(\mathbf{Y} - \bar{\mathbf{Y}}\right), \tag{35}$$

$$\mathbf{S} \leftarrow \texttt{SeqCholUpdate}\left(\mathbf{S}, -1, \mathbf{K}\mathbf{S}'^T\right), \tag{36}$$

where `SeqCholUpdate` denotes repeated Cholesky updating `CholUpdate` using successive columns of $\mathbf{K}\mathbf{S}'^T$ as the updating vector. To finally consider the Jacobian (see Section IV-E), we compute

$$\mathbf{S} \leftarrow \mathbf{S}\mathbf{J}^T, \tag{37}$$

letting $\mathbf{S}$ no longer triangular, but $\mathbf{S}$ keeps a valid matrix square root which could be used to define the next set of sigma points [28].