IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

Bayesian Filtering for Homography Estimation

Arturo Del Castillo Bernal, Philippe Decoste, and James Richard Forbes

***Abstract*—This paper considers homography estimation in a Bayesian filtering framework using rate gyro and camera measurements. The use of rate gyro measurements facilitates a more reliable estimate of homography in the presence of occlusions, while a Bayesian filtering approach generates both a homography estimate along with an uncertainty. Uncertainty information opens the door to adaptive filtering approaches, post-processing procedures, and safety protocols. In particular, herein an iterative extended Kalman filter and an interacting multiple model (IMM) filter are tested using both simulated and experimental datasets. The IMM is shown to have good consistency properties and better overall performance when compared to the state-of-the-art homography nonlinear deterministic observer in both simulations and experiments.**

***Index Terms*—Visual Tracking, Vision-Based Navigation, Sensor Fusion**

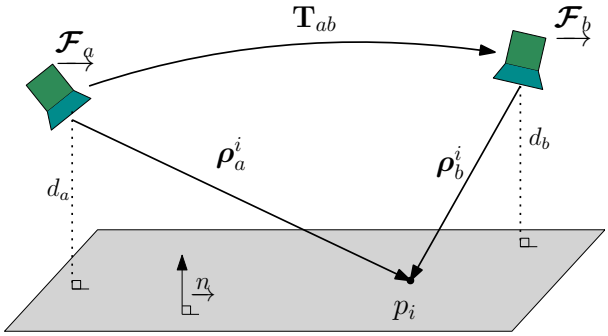

Fig. 1: The transformation of the observed position of points on a plane by a camera viewing the scene from different perspectives is described by a homography matrix.

I. Introduction and Related Work

A Homography is a mapping that relates two views of the same planar scene. It is exploited in robotics applications when the structure of the environment is sufficiently planar, such as indoor hallways, manmade experimental settings, and aerial coverage. Robotics applications that have successfully used homography include visual-servoing [1], image stabilization [2], ego-motion [3], [4], and monocular SLAM initialization [5]. Homographies can better explain the structure of planar scenes and low parallax compared to using the epipolar constraint [6], which is commonly employed for unstructured scenes.

Homography is usually estimated using feature correspondences between a pair of images, such as points, lines, conics or a combination thereof [7], [8]. Direct methods [9] and learning-based approaches [10], [11] are alternative means to estimate homography. All of these methods consider camera measurements independently and ignore any temporal correlations, thus making them susceptible to failures in the presence of occlusions or lack of feature correspondences [6].

In [12], temporal information is exploited to estimate homography by designing a nonlinear deterministic observer. This observer is able to operate even during occlusions by propagating prior estimates using angular velocity measurements, thus providing a more robust estimate of the homography. This comes at the cost of assuming the camera's velocity is constant parallel to the plane or exponentially converging towards to the plane. Further, homography is parameterized as an element of the special linear group $SL(3)$. Lie group properties are exploited in the observer structure to prove local asymptotic stability under various assumptions.

Nonlinear deterministic observers aim to show *a priori* stability properties, and they do not take into account stochastic processes, such as noise in sensor measurements, which is present in all real systems. This paper leverages the tools of Bayesian filtering to take into account noise statistics to produce an accurate and consistent estimate with a covariance describing the error distribution. A Bayesian filtering approach opens the door for procedures such as smoothing [13, Ch. 8], loop-closure detection [14], adaptive approaches [15, Ch. 11], or simply monitoring the filter's quality [15, Sec. 5.4].

This paper presents a Bayesian filtering approach to homography estimation using rate gyro and camera measurements in a tightly-coupled manner, with covariance information. The same assumption on the camera's inertial-frame velocity from [12] is made to enable the use of a rate gyro and a specific motion model. Because this assumption is violated from time to time in practice, focus is placed on a specific type of Bayes filter, the interacting multiple model (IMM) filter. The IMM filter adapts the noise level of the process model when the velocity assumption of the process model is violated. This approach is similar to the IMM application found in [16]. The IMM used here is composed of two iterated extended Kalman filters (EKFs). This paper clearly demonstrates improved performance, along with consistency, in simulation, as well as better overall performance in experiments of the IMM relative to a nonlinear deterministic observer [2]. As such, this paper's contribution is the combination of 1) a Bayesian filtering framework, 2) utilizing the camera velocity

Manuscript received June 14, 2021; Revised September 25, 2023; Accepted October 13, 2023.

This paper was recommended for publication by Editor P. Vasseur upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by the NSERC Discovery Grant program, and the Canadian Innovation Fund (CFI) program. This work was also supported in part by funding from the Innovation for Defence Excellence and Security (IDEaS) program from the Department of National Defence (DND). Any opinions and conclusions in this work are strictly those of the authors and do not reflect the views, positions, or policies of - and are not endorsed by - IDEaS, DND, or the Government of Canada.

A. Del Castillo Bernal, P. Decoste, and J. R. Forbes are with the Department of Mechanical Engineering, McGill University, Montreal, QC, Canada, H3A 0C3. arturo.delcastillo@mail.mcgill.ca

Digital Object Identifier (DOI): see top of this page.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

assumption of [12] in a probabilistic setting, thus providing a simple process model and enabling the use of the rate gyro, 3) considering the IMM structure to account for violations of the camera velocity assumption, and 4) simulating and experimentally testing the proposed IMM filter relative to a nonlinear deterministic observer.

Deterministic and Bayesian approaches to homography estimation are not mutually exclusive nor competitors. The choice to use one or the other, or perhaps both in "primary" and "back-up" roles, or even both in synergy, will be application dependent. In situations where a computationally simple observer is needed, but covariance information is not needed, a deterministic observer is a natural choice. On the other hand, in situations where covariance information is needed, and computational resources are not so limited, a Bayesian filter is appropriate. As such, this paper does not advocate for the displacement of [2], [4], [12] and similar work. Rather, this paper builds on [12] by providing a means to use both a rate gyro and camera to generate a homography estimate along with a covariance in a way that accounts for the limitations of the assumed process model of [12].

II. Preliminaries

Herein, $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is used to denote a Gaussian random variable $\mathbf{x} \in \mathbb{R}^n$ with mean $\boldsymbol{\mu} \in \mathbb{R}^n$ and positive definite covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{n\times n}$. The notation $\check{(\cdot)}$ and $\hat{(\cdot)}$ denotes the predicted and measurement-corrected state estimates, respectively. The superscripts $(\cdot)^i$ are indices, not exponents. For a random variable $\mathbf{X}$ evolving in a matrix Lie group G, a "Gaussian" distribution is represented by $\mathbf{X} = \exp(-\boldsymbol{\xi}^\wedge)\bar{\mathbf{X}}$, where $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, and $\bar{\mathbf{X}}$ is the mean [17].

For a generic function $\mathbf{g} : G \to \mathbb{R}^m$ of a Lie group element, the shorthand notation for a Lie derivative is defined as

$$\left.\frac{D\mathbf{g}(\mathbf{X})}{D\mathbf{X}}\right|_{\bar{\mathbf{X}}} \triangleq \left.\frac{\partial \mathbf{g}(\exp(-\boldsymbol{\xi}^\wedge)\bar{\mathbf{X}})}{\partial \boldsymbol{\xi}}\right|_{\boldsymbol{\xi}=\mathbf{0}}. \tag{1}$$

A. Homography

As shown in Figure 1, a homography mapping relates features on a plane from two distinct views. A point p_i lying on a plane has coordinates resolved in the camera frame $\mathcal{F}_a$ given by $\boldsymbol{\rho}_a^i = \begin{bmatrix} x & y & z \end{bmatrix}^\mathsf{T}$. The normalized image coordinates are

$$\mathbf{p}_a^i = \begin{bmatrix} x/z \\ y/z \\ 1 \end{bmatrix} = \frac{1}{z}\boldsymbol{\rho}_a^i. \tag{2}$$

It is possible to obtain the projection of $\mathbf{p}_a^i$ on the image plane $\mathbf{q}_a^i$ by using the intrinsic parameter matrix $\mathbf{K}$, written as

$$\mathbf{q}_a^i = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_u & 0 & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \mathbf{p}_a^i, \tag{3}$$

where f_u and f_v are the horizontal and vertical focal lengths and c_u and c_v are the optical center coordinates in pixels. As the camera's pose changes by $\mathbf{T}_{ab} \in SE(3)$ [18, Sec. 8.1], it is possible to relate a new observation $\mathbf{q}_b^i$ of p_i to $\mathbf{q}_a^i$ by a *homography matrix* $\mathbf{H}_{ab}$ [12],

$$\mathbf{q}_b^i = \frac{z_a}{z_b}\mathbf{K}\left(\mathbf{C}_{ab} - \frac{1}{d_b}\mathbf{r}_a^{ba}\mathbf{n}_b^\mathsf{T}\right)^{-1}\mathbf{K}^{-1}\mathbf{q}_a^i = \mathbf{K}\mathbf{H}_{ab}^{-1}\mathbf{K}^{-1}\mathbf{q}_a^i, \tag{4}$$

where $\mathbf{C}_{ab} \in SO(3)$ is the direction cosine matrix that encodes the orientation of $\mathcal{F}_a$ relative to $\mathcal{F}_b$, $\mathbf{r}_a^{ba}$ is the position of the origin of $\mathcal{F}_b$ relative to the origin of $\mathcal{F}_a$ resolved in $\mathcal{F}_a$, $\mathbf{n}_b$ is the normalized plane direction resolved in $\mathcal{F}_b$, and d_b is the orthogonal distance from the camera position to the plane. The definition $\gamma \triangleq z_a/z_b$ will be used throughout the paper.

B. Special Linear Group

There are 6 degrees of freedom associated with rotation and translation, and 3 degrees of freedom associated with relative distance. However, a homography matrix only has 8 degrees of freedom, not 9, because it is defined only up to the scalar γ. One way to parameterize homography is by the Special Linear Group $SL(3)$ [19, Chap. 1],

$$SL(3) \equiv \{\mathbf{H} \in \mathbb{R}^{3\times 3} \mid \det \mathbf{H} = 1\}. \tag{5}$$

Any nonsingular matrix $\mathbf{X} \in \mathbb{R}^{3\times 3}$ can be projected onto $SL(3)$ by $\mathbf{X}/\left(\det \mathbf{X}\right)^{1/3} \in SL(3)$. As with any Lie group, there is a Lie algebra associated with $SL(3)$, defined as

$$\mathfrak{sl}(3) \equiv \{\boldsymbol{\Xi} \in \mathbb{R}^{3\times 3} \mid \operatorname{tr}(\boldsymbol{\Xi}) = 0\}, \tag{6}$$

where $\operatorname{tr}(\cdot) : \mathbb{R}^{3\times 3} \to \mathbb{R}$ is the trace operator. The matrix exponential is a surjective mapping that relates the Lie algebra to the Lie group, $\mathbf{X} = \exp(\boldsymbol{\Xi})$. For $SL(3)$, the inverse mapping is defined only for elements near identity, $\boldsymbol{\Xi} = \log(\mathbf{X})$. A vector can be uniquely associated to an element of $\mathfrak{sl}(3)$ by $(\cdot)^\wedge : \mathbb{R}^8 \to \mathfrak{sl}(3)$, which is defined as [20],

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} \xi_4 + \xi_5 & -\xi_3 + \xi_6 & \xi_1 \\ \xi_3 + \xi_6 & \xi_4 - \xi_5 & \xi_2 \\ \xi_7 & \xi_8 & -2\xi_4 \end{bmatrix}. \tag{7}$$

The operator $(\cdot)^\vee : \mathfrak{sl}(3) \to \mathbb{R}^8$ is defined as the inverse of $(\cdot)^\wedge$. The adjoint operator $\mathrm{Ad} : SL(3) \times \mathfrak{sl}(3) \to \mathfrak{sl}(3)$ is defined as $\mathrm{Ad}(\mathbf{H})\boldsymbol{\Xi} = \mathbf{H}\boldsymbol{\Xi}\mathbf{H}^{-1}$, where $\boldsymbol{\Xi} \in \mathfrak{sl}(3)$. The adjoint matrix $\mathbf{Ad}(\mathbf{H})$ can be obtained by $(\mathrm{Ad}(\mathbf{H})\boldsymbol{\Xi})^\vee = \mathbf{Ad}(\mathbf{H})\boldsymbol{\xi}$. The Lie bracket $[\cdot,\cdot] : \mathfrak{sl}(3) \times \mathfrak{sl}(3) \to \mathfrak{sl}(3)$ is defined as $[\boldsymbol{\Xi}_1, \boldsymbol{\Xi}_2] = \boldsymbol{\Xi}_1\boldsymbol{\Xi}_2 - \boldsymbol{\Xi}_2\boldsymbol{\Xi}_1$, and a *little adjoint* matrix can be defined also as, $[\boldsymbol{\Xi}_1, \boldsymbol{\Xi}_2]^\vee = \mathbf{ad}(\boldsymbol{\xi}_1)\boldsymbol{\xi}_2$. The $(\cdot)^\odot : \mathbb{R}^3 \to \mathbb{R}^{3\times 8}$ operator is defined as $\boldsymbol{\xi}^\wedge\mathbf{p} = \mathbf{p}^\odot\boldsymbol{\xi}$, where $\mathbf{p} \in \mathbb{R}^3$.

C. Homography Kinematics

The kinematics of $\mathbf{H}_{ab}$ are [12]

$$\dot{\mathbf{H}}_{ab} = \mathbf{H}_{ab}\left(\boldsymbol{\omega}_b^{ba\times} - \frac{\mathbf{v}_b^{ba}\mathbf{n}_b^\mathsf{T}}{d_b} + \frac{\mathbf{n}_b^\mathsf{T}\mathbf{v}_b^{ba}}{3d_b}\mathbf{1}\right), \tag{8}$$

where $\boldsymbol{\omega}_b^{ba}$ is the angular velocity of the body with respect to (w.r.t.) the inertial frame resolved in $\mathcal{F}_b$, $\mathbf{v}_b^{ba}$ is the linear velocity of the body w.r.t. the inertial frame resolved in $\mathcal{F}_b$, and $(\cdot)^\times : \mathbb{R}^3 \to \mathfrak{so}(3)$, where $\mathfrak{so}(3)$ is the Lie algebra of $SO(3) \subset SL(3)$. The kinematic model (8) depends on the plane parameters $\mathbf{n}_b$ and d_b, which are unmeasurable.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

However, by assuming that $\mathbf{s}_a \triangleq \frac{\dot{\mathbf{r}}_a^{ba}}{d_b}$ is constant in time as in [2], [12], a simplified kinematic model can be derived. Setting $\mathbf{\Gamma}_{ab} = -\frac{\mathbf{v}_b^{ba}\mathbf{n}_b^\mathsf{T}}{d_b} + \frac{\mathbf{n}_b^\mathsf{T}\mathbf{v}_b^{ba}}{3d_b}\mathbf{1} \in \mathfrak{sl}(3)$, where $\mathbf{1}$ is the identity matrix, the assumption that $\mathbf{s}_a$ is constant yields the simplified process model

$$\dot{\mathbf{H}}_{ab} = \mathbf{H}_{ab}\left(\boldsymbol{\omega}_b^{ba\times} + \mathbf{\Gamma}_{ab}\right), \qquad \dot{\mathbf{\Gamma}}_{ab} = \left[\mathbf{\Gamma}_{ab},\ \boldsymbol{\omega}_b^{ba\times}\right]. \quad (9)$$

This simplified process model depends only on angular velocity, which is easily measured by a rate gyro. However, assuming $\mathbf{s}_a$ is constant restricts the trajectory of the camera to have constant velocity parallel to the plane, or to exponentially converge towards the plane [12].

D. Interacting Multiple Model

An IMM manages multiple models of a dynamic system in estimation tasks. The model at time step k is assumed to be among n possible models $\theta(k) = \{\theta_i\}_{i=1}^n$. The model switching in the system is assumed to be a Markov chain with known transition probabilities

$$p_{ij} \triangleq P\left(\theta(k) = \theta_j | \theta(k-1) = \theta_i\right). \quad (10)$$

In robotics applications, the IMM is used to mix the estimates of multiple filters each with their own process model of the system, where the i^{th} process model $\mathbf{f}_i(\cdot)$ is written

$$\mathbf{x}_k^i = \mathbf{f}_i(\mathbf{x}_{k-1}^i, \mathbf{u}_{k-1}). \quad (11)$$

To avoid the exponential complexity of accounting for all possible n^k combination of models at time step k, the IMM runs n process models in parallel at all times, each with a weight $w_k^i \triangleq P(\theta(k) = \theta_i | \mathbf{y}_{1:k})$. When a new measurement $\mathbf{y}_k$ arrives, the IMM executes the following three steps [15, Sec. 11.6].

1) Interaction: Mixing probabilities are calculated by

$$\begin{aligned}\mu_{k-1}^{ij} &\propto P\left(\theta(k) = \theta_j | \theta(k-1) = \theta_i\right) \times \\ &\qquad P\left(\theta(k-1) = \theta_i | \mathbf{y}_{1:k-1}\right), \\ \mu_{k-1}^{ij} &= \frac{1}{c} p_{ij} w_{k-1}^i,\end{aligned} \quad (12)$$

where c is a normalization constant. The mixing probabilities represent how likely a switch is to happen given the history of measurements or current knowledge of the trajectory. Every model i computes mixing probabilities with the rest of the models including itself.

2) Mixing: Since it is assumed that the states are Gaussian distributed, a Gaussian mixture procedure, considering the mixing probabilities, is carried out to update the state and covariance estimate of each filter,

$$\check{\mathbf{x}}_{k-1}^i \leftarrow \sum_j^n \mu_{k-1}^{ij} \check{\mathbf{x}}_{k-1}^j, \quad (13)$$

$$\check{\mathbf{P}}_{k-1}^i \leftarrow \sum_j^n \mu_{k-1}^{ij} (\check{\mathbf{P}}_k^j + (\check{\mathbf{x}}_{k-1}^j - \check{\mathbf{x}}_{k-1}^i)(\check{\mathbf{x}}_{k-1}^j - \check{\mathbf{x}}_{k-1}^i)^\mathsf{T}). \quad (14)$$

The mixing is straightforward for states in vector space. For states that evolve in a Lie group, a reparametrization of each estimate's $\check{\mathbf{X}}_{k-1}^j$ distribution is needed about a reference point, in this case $\check{\mathbf{X}}_{k-1}^i$, to perform a Gaussian mixture [21]. To this end,

$$\begin{aligned}\boldsymbol{\epsilon}_{k-1}^{ji} &= \log\left(\check{\mathbf{X}}_{k-1}^i \mathbf{X}^{j}{}_{k-1}^{-1}\right)^\vee \\ &\approx \bar{\boldsymbol{\epsilon}}_{k-1}^{ji} + \mathbf{J}^r(\bar{\boldsymbol{\epsilon}}_{k-1}^{ji})^{-1} \boldsymbol{\xi}_{k-1}^j,\end{aligned} \quad (15)$$

$$\check{\mathbf{P}}_{k-1}^{ji} = \mathbf{J}^r(\bar{\boldsymbol{\epsilon}}_{k-1}^{ji})^{-1} \check{\mathbf{P}}_{k-1}^j \mathbf{J}^r(\bar{\boldsymbol{\epsilon}}_{k-1}^{ji})^{-\mathsf{T}}, \quad (16)$$

where $\bar{\boldsymbol{\epsilon}}_{k-1}^{ji}$ and $\check{\mathbf{P}}_{k-1}^{ij}$ are the reprojected mean and covariance of $\check{\mathbf{X}}_k^j$ about $\check{\mathbf{X}}_{k-1}^i$. Equation (15) is a first-order approximation, with $\mathbf{J}^r(\bar{\boldsymbol{\epsilon}}_{k-1}^{ji})$ being the right group Jacobian of $SL(3)$ about $\bar{\boldsymbol{\epsilon}}_{k-1}^{ji}$, calculated using a backward finite-difference. By applying (14) and (13), the reprojected means and covariances $\bar{\boldsymbol{\epsilon}}_{k-1}^{ji}$ and $\check{\mathbf{P}}_{k-1}^{ij}$ can be combined to produce $\bar{\boldsymbol{\epsilon}}_{k-1}^{i}$ and $\check{\mathbf{P}}_{k-1}^{i}$. To finalize the Gaussian mixture on $SL(3)$, these values are then projected onto the $SL(3)$ manifold as follows,

$$\check{\mathbf{X}}_{k-1}^i \leftarrow \exp\left(-\bar{\boldsymbol{\epsilon}}_{k-1}^i\right) \check{\mathbf{X}}_{k-1}^i, \quad (17)$$

$$\check{\mathbf{P}}_{k-1}^i \leftarrow \mathbf{J}^r(\bar{\boldsymbol{\epsilon}}_{k-1}^i) \check{\mathbf{P}}_{k-1}^i \mathbf{J}^r(\bar{\boldsymbol{\epsilon}}_{k-1}^i)^\mathsf{T}. \quad (18)$$

3) Correction: Once each filter has an updated mixed state estimate, the likelihood that filter i generated the measurement is computed to update the weights of the filters,

$$\Lambda_k^i = p(\mathbf{y}_k | \mathbf{X}_{k-1}^i, \theta_{k-1}), \quad w_k^i = \frac{1}{c} \Lambda_k^i \sum_j^n w_{k-1}^j p_{ij}, \quad (19)$$

where c is a normalization constant. A more in-depth description of the IMM can be found in [15, Sec. 11.6].

III. Homography Estimation using the IMM Filter

The states to be estimated are $\mathbf{H}_{ab}$ and $\mathbf{\Gamma}_{ab}$. An IMM is implemented employing two (iterated) EKFs. Both EKFs share the same measurement and process model, but have different process model noise levels, similar to [16]. Although the true noise levels are unknown in this work, the same principle will be used to manage how confident the filter is in the assumption that $\mathbf{s}_a$ is constant. To implement an EKF, the Jacobians of the process and measurement models must be obtained.

A. Process Model

The process model has access to a rate gyro measurement $\mathbf{u} = \boldsymbol{\omega}_b^{ba} + \mathbf{w}$ resolved in the body frame $\mathcal{F}_b$. It is assumed that the rate gyro is unbiased and is only corrupted by white Gaussian noise $\mathbf{w}(t) \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\mathcal{Q}}_g \delta(t-\tau)\right)$. From (9), the process model is

$$\dot{\mathbf{H}}_{ab} = \mathbf{H}_{ab}\left((\mathbf{u}^\times - \mathbf{w}^\times) + \mathbf{\Gamma}_{ab}\right), \quad (20)$$

$$\dot{\mathbf{\Gamma}}_{ab} = \left[\mathbf{\Gamma}_{ab}, \mathbf{u}^\times - \mathbf{w}^\times\right] + \mathbf{w}^{m\wedge}. \quad (21)$$

As previously mentioned, assuming $\mathbf{s}_a$ is constant in time comes at the cost of process model inaccuracies when the assumption is broken. One way to account for modelling errors is adding a noise term $\mathbf{w}^m(t) \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\mathcal{Q}}_m \delta(t-\tau)\right) \in \mathbb{R}^8$. The power spectral density (PSD) is modelled as $\boldsymbol{\mathcal{Q}}_m = \sigma_m^2 \mathbf{1}$. The size of σ_m^2 represents how confident the filter is on the motion model for $\mathbf{\Gamma}_{ab}$.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

The IMM introduced in Section II-D is a common adaptive method that weights the available motion hypotheses on a Bayesian framework to produce an estimate. The idea is to have two similar models, with different values of σ_m^2. Low values accommodate for scenarios where the assumption that $\mathbf{s}_a$ is constant is respected, and higher values can deal with scenarios where the motion violates this assumption. The IMM should provide an estimate with better consistency properties than using a single filter in cases where the trajectory is more varied, switching from slow to aggressive maneuvers.

To linearize the process model, a right-invariant error definition is used, where

$$\exp(\delta\boldsymbol{\xi}^{\wedge}) = \bar{\mathbf{H}}\mathbf{H}^{-1}, \qquad \delta\boldsymbol{\gamma}^{\wedge} = \boldsymbol{\Gamma} - \bar{\boldsymbol{\Gamma}}, \tag{22}$$

applying a Taylor series expansion, the linearized process model evolving in the Lie algebra is

$$\delta\dot{\boldsymbol{\xi}}_{ab}^{\wedge} = -\mathrm{Ad}\left(\bar{\mathbf{H}}_{ab}\right)\delta\boldsymbol{\gamma}_{ab}^{\wedge} + \mathrm{Ad}\left(\bar{\mathbf{H}}_{ab}\right)\delta\mathbf{w}^{\times}, \tag{23a}$$

$$\delta\dot{\boldsymbol{\gamma}}_{ab}^{\wedge} = -\mathrm{ad}\left(\mathbf{u}\right)\delta\boldsymbol{\gamma}_{ab}^{\wedge} - \mathrm{ad}\left(\bar{\boldsymbol{\gamma}}_{ab}\right)\delta\mathbf{w}^{\times} + \delta\mathbf{w}^{m\wedge}. \tag{23b}$$

When applying the linear $(\cdot)^{\vee}$ operator on (23), $\delta\mathbf{w}^{\times}$ has to be addressed. A projection matrix $\mathbf{B}$ can be found such that, $\delta\mathbf{w}^{\times\vee} = \mathbf{B}\delta\mathbf{w}$. As such, the linearized process model is thus

$$\delta\dot{\boldsymbol{\xi}}_{ab} = -\mathbf{Ad}\left(\bar{\mathbf{H}}_{ab}\right)\delta\boldsymbol{\gamma}_{ab} + \mathbf{Ad}\left(\bar{\mathbf{H}}_{ab}\right)\mathbf{B}\delta\mathbf{w}, \tag{24a}$$

$$\delta\dot{\boldsymbol{\gamma}}_{ab} = -\mathbf{ad}\left(\mathbf{u}\right)\delta\boldsymbol{\gamma}_{ab} - \mathbf{ad}\left(\bar{\boldsymbol{\gamma}}_{ab}\right)\mathbf{B}\delta\mathbf{w} + \delta\mathbf{w}^{m}. \tag{24b}$$

Setting $\delta\mathbf{x}_{ab} = \begin{bmatrix}\delta\boldsymbol{\xi}_{ab}^{\mathsf{T}} & \delta\boldsymbol{\gamma}_{ab}^{\mathsf{T}}\end{bmatrix}^{\mathsf{T}}$, and $\delta\mathbf{w}_{ab} = \begin{bmatrix}\delta\mathbf{w}^{\mathsf{T}} & \delta\mathbf{w}^{m\mathsf{T}}\end{bmatrix}^{\mathsf{T}}$, the linearized process model can be written as,

$$\delta\dot{\mathbf{x}}_{ab} = \underbrace{\begin{bmatrix}\mathbf{0} & -\mathbf{Ad}\left(\bar{\mathbf{H}}_{ab}\right)\\ \mathbf{0} & -\mathbf{ad}\left(\mathbf{u}\right)\end{bmatrix}}_{\mathbf{A}}\delta\mathbf{x}_{ab} + \underbrace{\begin{bmatrix}\mathbf{Ad}\left(\bar{\mathbf{H}}_{ab}\right)\mathbf{B} & \mathbf{0}\\ -\mathbf{ad}\left(\bar{\boldsymbol{\gamma}}_{ab}\right)\mathbf{B} & \mathbf{1}\end{bmatrix}}_{\mathbf{L}}\delta\mathbf{w}_{ab}. \tag{25}$$

At this point, the linearized process model given by (25) can be discretized using the matrix exponential [18], [22]. The resulting discrete-time Jacobians are then used in a standard iterated EKF framework [23].

B. Measurement Model

From (4), given a point correspondence obtained from camera measurements of a point feature p_i lying on a plane, the pair can be related by a homography transformation $\mathbf{H}_k \triangleq \mathbf{H}_{ab}$ via

$$\mathbf{y}_b^i = \mathbf{g}(\mathbf{H}_k^{-1}\mathbf{p}_a^i) + \mathbf{v}_k, \tag{26}$$

where $\mathbf{p}_a^i$ represents the measurement of p_i resolved in $\mathcal{F}_a$, in normalized image coordinates, $\mathbf{y}_b^i$ the noisy measurement in pixel coordinates of the same feature resolved in $\mathcal{F}_b$, and $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \in \mathbb{R}^2$ models white noise on the pixel measurement. Setting $\mathbf{r}_b^i \triangleq \mathbf{H}_k^{-1}\mathbf{p}_a^i = \begin{bmatrix}x & y & z\end{bmatrix}^{\mathsf{T}}$,

$$\mathbf{g}\left(\mathbf{r}_b^i\right) \triangleq \frac{1}{z}\mathbf{D}\mathbf{K}\mathbf{r}_b^i, \tag{27}$$

where $\mathbf{D} = \begin{bmatrix}\mathbf{1}_{2\times 2} & \mathbf{0}_{2\times 1}\end{bmatrix}$ and $\mathbf{K}$ is the previously introduced intrinsic parameter matrix to model a pinhole camera. The linearized measurement model is then

$$\delta\mathbf{y}_b^i \approx \frac{\partial\mathbf{g}(\mathbf{r}_b^i)}{\partial\mathbf{r}_b^i}\left[\left.\frac{D\mathbf{r}_b^i(\mathbf{H}_k)}{D\mathbf{H}_k}\right|_{\bar{\mathbf{H}}_k} \quad \left.\frac{\partial\mathbf{r}_b^i(\boldsymbol{\Gamma}_k)}{\partial\boldsymbol{\Gamma}_k}\right|_{\bar{\boldsymbol{\Gamma}}_k}\right]\delta\mathbf{x}_k + \delta\mathbf{v}_k, \tag{28}$$

where

$$\frac{\partial\mathbf{g}(\mathbf{r}_b^i)}{\partial\mathbf{r}_b^i} = \frac{1}{z}\begin{bmatrix}f_u & 0 & -f_u\frac{x}{z}\\ 0 & f_v & -f_v\frac{y}{z}\end{bmatrix}, \tag{29}$$

$$\left[\left.\frac{D\mathbf{r}_b^i(\mathbf{H}_k)}{D\mathbf{H}_k}\right|_{\bar{\mathbf{H}}_k} \quad \left.\frac{\partial\mathbf{r}_b^i(\boldsymbol{\Gamma}_k)}{\partial\boldsymbol{\Gamma}_k}\right|_{\bar{\boldsymbol{\Gamma}}_k}\right] = \left[\bar{\mathbf{H}}_k^{-1}\mathbf{p}_a^{i\,\odot} \quad \mathbf{0}_{3\times 8}\right]. \tag{30}$$

With these Jacobians, the prediction and correction steps of the iterative EKF can be executed.

C. Robust Loss

An iterated EKF's correction step can be formulated as a weighted nonlinear least squares problem (WNLS) [23] solved using the Gauss–Newton algorithm. In particular, the error and cost function, respectively, associated with the WNLS problem are

$$\mathbf{e}(\mathbf{x}_k)^{\mathsf{T}} = \begin{bmatrix}(\check{\mathbf{x}}_k - \mathbf{x}_k)^{\mathsf{T}} & (\mathbf{y}_k - \mathbf{g}(\mathbf{x}_k))^{\mathsf{T}}\end{bmatrix}, \tag{31}$$

$$J(\mathbf{x}_k) = \frac{1}{2}\mathbf{e}(\mathbf{x}_k)^{\mathsf{T}}\underbrace{\begin{bmatrix}\check{\mathbf{P}}_k^{-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{R}_k^{-1}\end{bmatrix}}_{\mathbf{W}}\mathbf{e}(\mathbf{x}_k). \tag{32}$$

To mitigate the detrimental impact of outliers in the correction step, a robust loss is applied on the measurement residuals. It is straightforward to modify $\mathbf{W}$ to add the robust loss weights, as shown in [24]. The *SC/DCS* robust loss is chosen due to the properties explained in [24]. The *SC/DCS* weight function is given by $w(r) = \frac{4c^2}{(c+r^2)^2}$ if $r^2 \geq c$, otherwise, $w(r) = 1$.

IV. Simulation Results

The proposed IMM filter is first tested in simulation. A set of trajectories are generated, where a camera tracks 4 points on a plane at all times, the minimum number of point matches to define a homography [6]. The rate gyro and camera provide data at 90 Hz and 30 Hz, respectively. Gyroscope and camera measurements are corrupted by additive Gaussian noise, simulated using $\boldsymbol{\mathcal{Q}}_g = \sigma_g^2\mathbf{1}$, $\mathbf{R}_k = \sigma_r^2\mathbf{1}$, with $\sigma_g = 0.01\,\mathrm{rad\,s^{-1}}$, $\sigma_r = 1\,\mathrm{pixel}$. Initial uncertainty is set to $\hat{\mathbf{P}}_0 = 1\times 10^{-1}\mathbf{1}$.

100 Monte Carlo runs with varying initial conditions and noise realizations are performed for each trajectory to evaluate the filters' consistency and accuracy when the assumption that $\mathbf{s}_a$ is constant is broken in different ways. Among the tested filters are 2 versions of the iterated EKF. The first is *EKF tight*, which is confident in the assumption that $\mathbf{s}_a$ is constant, by setting $\sigma_m^2 = 1\times 10^{-7}$. The second is *EKF loose*, with $\sigma_m^2 = 1\times 10^{-1}$, which has little confidence in the assumption that $\mathbf{s}_a$ is constant. An IMM composed of both versions of the EKFs is also tested, as well as the observer from [2].

To evaluate the accuracy of the filters on each trajectory, the error

$$r_k = \|\log\left(\hat{\mathbf{H}}_k\mathbf{H}_k^{-1}\right)^{\vee}\|_2 \tag{33}$$

is used to compare homography estimates to the true homography value at each time step. In Figure 3, it is shown how r_k is distributed across all time steps in 4 trajectories

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

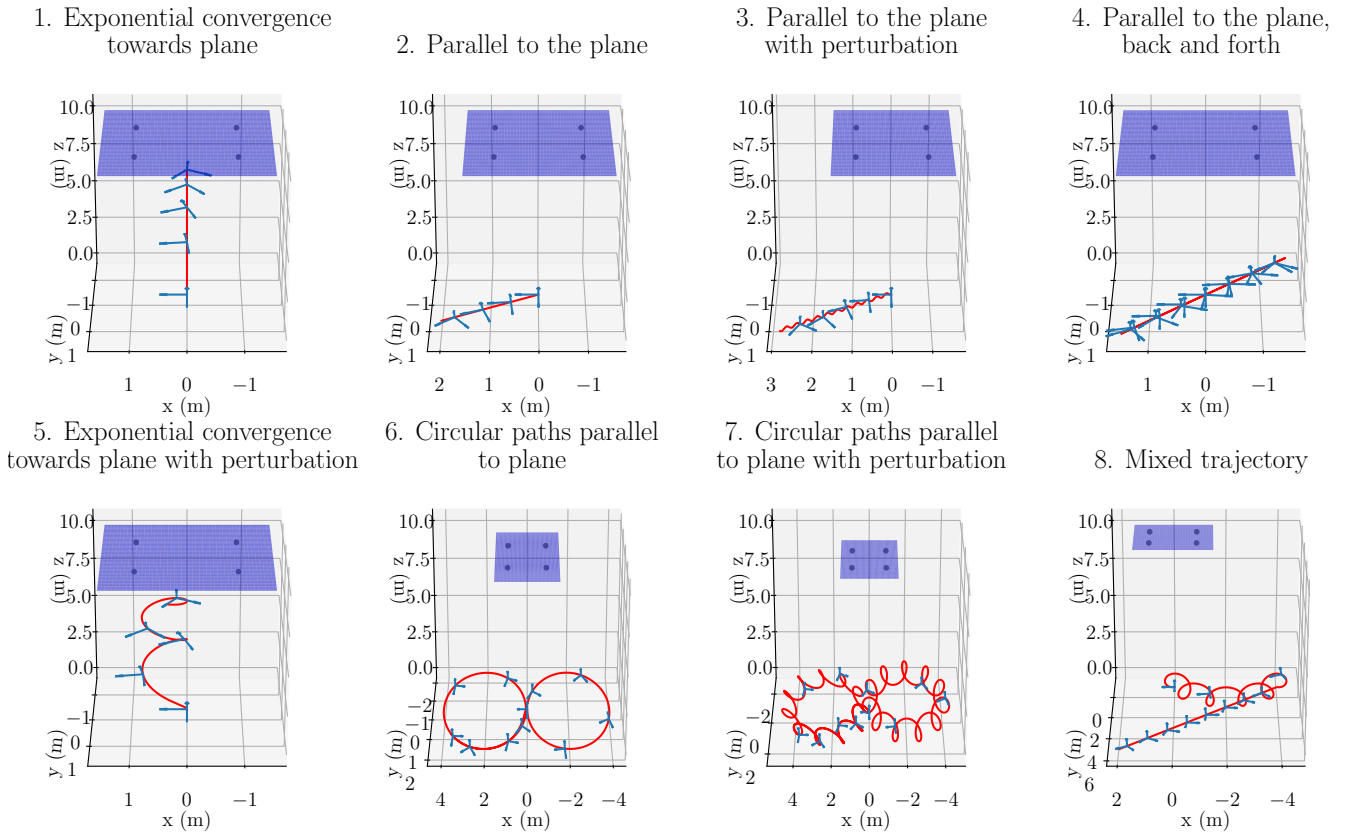

Fig. 2: Simulated trajectories. Trajectories 1 and 2 do not violate the assumption that $\mathbf{s}_a$ is constant. The rest of the trajectories violate the assumption that $\mathbf{s}_a$ is constant.

from Figure 2. The trajectories displayed in Figure 3 are chosen to provide distinctive information about the estimators' performance. For example, for Trajectories 1, 2, and 3, the plots look very similar, so only Trajectory 1 out of 1, 2, and 3, is presented. The average of r_k across all trials and then averaged across all time steps is displayed in Table I. *EKF tight* performs the best on this metric when the trajectories respect the assumption that $\mathbf{s}_a$ is constant as in Trajectories 1 and 2, or closely do, as in Trajectory 3. In the remaining trajectories, the performance of *EKF tight* degrades and becomes the worst among the tested filters. In general, the IMM offers the best performance when the assumption is less respected. Only in Trajectory 7, when the assumption is severely broken, *EKF loose* outperforms the IMM by a small margin. The observer of [2] has higher homography estimation error in all trials, since all the filters are characterizing the sensor noise properly and tuned accordingly. When tuning the observer it is observed that modifying the gains did not change the observer's performance drastically. The observer is, in effect, constant gain, while the IMM filter changes the gain at each time step.

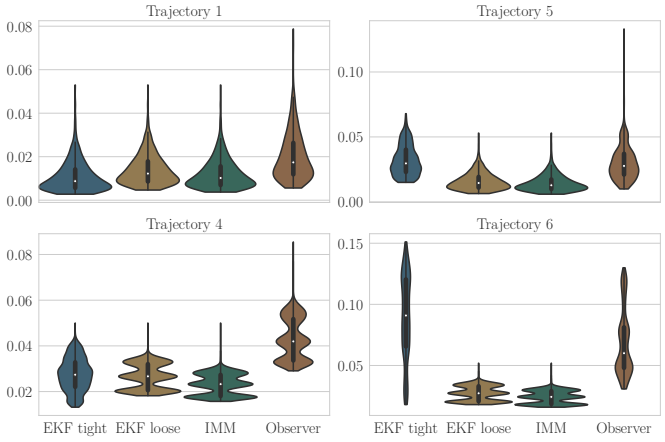

Fig. 3: Violin plots to show distributions of r_k in 4 sample trajectories from simulated data. EKF tight/loose, IMM, and observer are tested.

Assessing consistency is done using the normalized estimation error squared (NEES) test [15, Sec. 5.4] on Monte Carlo runs. The NEES test involves computing a $\epsilon_k \sim \chi^2_8$ statistic using the error trajectory and the predicted covariance of such error, where χ^2_8 is the Chi-square distribution with 8 degrees of freedom,

$$\boldsymbol{\xi}_k = \log\left(\hat{\mathbf{H}}_k \mathbf{H}_k^{-1}\right)^{\vee}, \qquad \epsilon_k = \boldsymbol{\xi}_k^{\mathsf{T}} \hat{\mathbf{P}}_{hh,k}^{-1} \boldsymbol{\xi}_k, \tag{34}$$

where $\hat{\mathbf{P}}_{hh,k}$ is the block on the diagonal of $\hat{\mathbf{P}}_k$ corresponding to the homography state. To assess with $3\sigma \approx 99.73\%$ confidence if the estimator is consistent, ϵ_k should remain between an upper and lower threshold [15]. In Figure 4, ϵ_k is averaged across all 100 trials, and then plotted as a function of time. *EKF tight* is a consistent estimator when the assumption

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

TABLE I: Performance of EKF tight/loose, IMM, and observer using simulated data. The error r_k is averaged across 100 Monte Carlo trials and then time steps.

Traj.	$\mathbb{E}[r_k]$ EKF tight	EKF loose	IMM	Observer	% Diff. IMM/Obs.
1	**0.0108**	0.0140	0.0121	0.0201	39.5%
2	**0.0212**	0.0269	0.0237	0.0424	44.1%
3	**0.0207**	0.0266	0.0235	0.0433	45.7%
4	0.0274	0.0266	**0.0231**	0.0429	46.1%
5	0.0319	0.0158	**0.0143**	0.0308	53.4%
6	0.0897	0.0273	**0.0242**	0.0684	64.6%
7	0.1319	**0.0383**	0.0388	0.1268	69.4%
8	0.0501	0.0299	**0.0272**	0.1062	74.4%

that $\mathbf{s}_a$ is constant is respected, but as soon as the assumption is broken, the NEES values diverge, as shown in the top of Figure 4. This is expected, since the errors are large. For *EKF loose*, the NEES value is below the lower threshold, producing inconsistent results since the covariance estimate is too large. However, having a large covariance is preferable to having the error be too large for the covariance. For the IMM, the NEES value goes below the lower threshold in some trajectories for the same reasons as *EKF loose* does, but stays closer to the lower threshold, indicating that the covariance is better modulated than *EKF loose*, and providing a more consistent estimate. These results favour the implementation of the IMM over a single EKF with a fixed process model noise level. Tuning the EKF is hard as the covariance is dependent on the level in which the motion breaks the assumption that $\mathbf{s}_a$ is constant, and therefore an adaptive covariance is required.

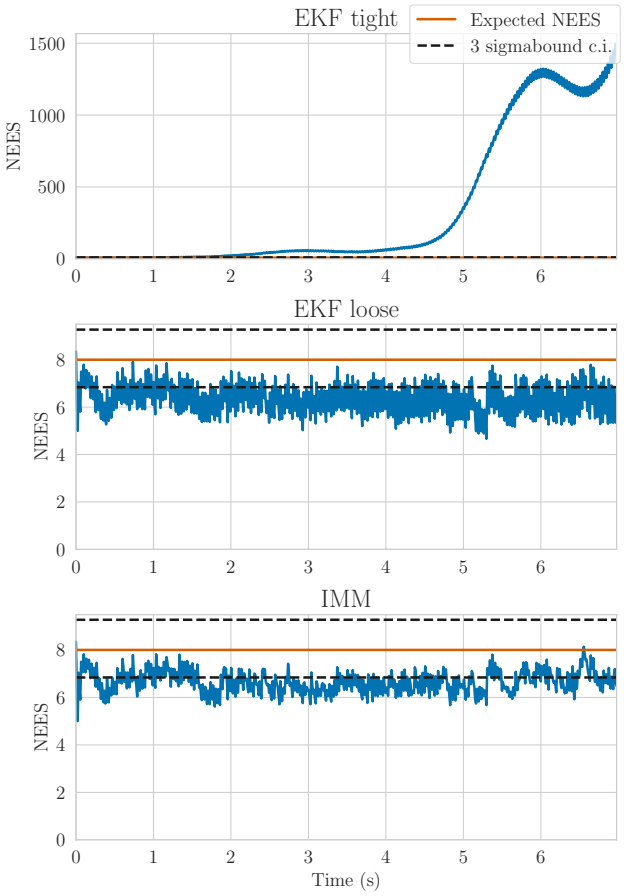

Fig. 4: Example of NEES plots for Monte Carlo trials on Trajectory 5 comparing *EKF tight*, *EKF loose* and IMM approaches.

V. Experimental Results

TABLE II: Performance of EKF loose, IMM, and observer using experimental data. The error r_k is averaged across all time steps.

Traj.	$\mathbb{E}[r_k]$ EKF loose	IMM	Observer	% Diff. IMM/Obs.
1	0.0837	**0.0381**	0.0512	25.6 %
2	**0.0679**	**0.0691**	0.0923	25.1 %
3	0.0681	**0.0577**	0.1352	57.3 %
4	0.0595	**0.0542**	0.1619	66.5 %
5	0.0910	**0.0656**	0.0787	16.7 %
6	0.0582	0.0485	**0.0434**	-11.7 %
7	0.0782	**0.0607**	0.0844	28.0 %
8	0.0673	0.0498	**0.0396**	-25.6 %
9	0.1176	**0.1002**	0.1145	12.6 %

An Intel Realsense D435i is used to collect data. Angular velocity measurements are provided by a built-in rate gyro at 200 Hz. The left camera of the stereo rig is used for camera measurements of 640×480 pixels at 30 Hz. Ground truth data is collected using an OptiTrack optical motion capture system at 120 Hz. The noise parameters used are $\sigma_g = 0.022\,\mathrm{rad\,s^{-1}}$, $\sigma_r = 1\,\mathrm{pixel}$.

Nine trials are recorded, all about 1 minute long. In each trial the camera tracks a plane about $1.5\,\mathrm{m}$ away, consisting of magazine pages spread on the floor, so salient features can be detected by the front-end system, as shown in Figure 6. The camera moves above this plane while rotating, with different paces in all trials to test how well the filters work when the assumption that $\mathbf{s}_a$ is constant in time is broken. The camera does not observe the plane for brief moments of time, and occlusions are also added in some trials to assess the robustness of the proposed approaches.

In order to obtain point correspondences for every tested estimator, ORB descriptors are used in its `OpenCV` implementation [25]. Since the goal is to estimate homography w.r.t. a reference frame, the feature points from each image are all matched against those from a reference image, which is picked from the first few frames in the trials. RANSAC is used to remove outliers prior to the estimation process. The robust loss described in Section III-C, is also employed in all estimators to ensure any remaining outliers are indeed discarded.

The state estimate is initialized as $\check{\mathbf{H}}_0 = \mathbf{1}, \check{\boldsymbol{\Gamma}}_0 = \mathbf{0}$ with $\check{\mathbf{P}}_0 = 10^{-4}\mathbf{1}$ for all trials since the reference frame was set as one of the frames recorded and the filter initialized at that same frame, so the initial error is very small. The IMM is composed in this case of two iterated EKFs, the first one with $\sigma_m^2 = 10^{-6}$ and the second with $\sigma_m^2 = 1$. The robust loss function implemented to reject outliers is *SC/DCS* from [24], with $c = 9.5$. The transition probabilities are set as $\boldsymbol{\Pi} = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$. This value is picked heuristically and is observed

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

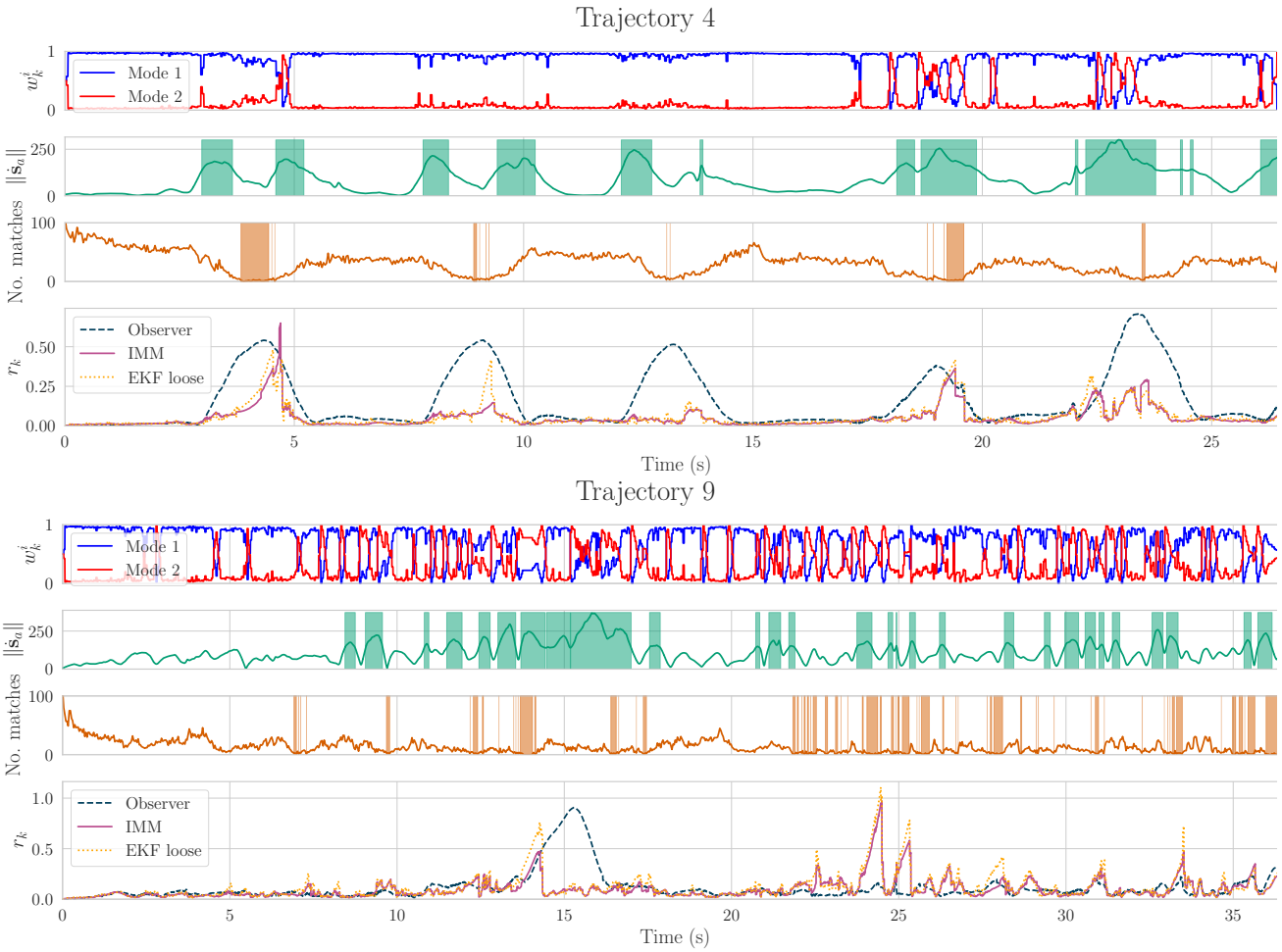

Fig. 5: IMM model probabilities depending on how the $\mathbf{s}_a$ is constant assumption is broken and number of matches in Trajectories 4 and 9. Mode 1 corresponds to $\sigma_m^2 = 10^{-6}$ and mode 2 corresponds to $\sigma_m^2 = 1$. Evolution of IMM, *EKF loose*, and observer's error r_k is shown in bottom plots. Green shaded regions represent time frames where $\|\dot{\mathbf{s}}_a\| > \alpha = 155$. Orange shaded regions represent time frames where number of tracked features goes below 4. When $\|\dot{\mathbf{s}}_a\| > \alpha$, the observer performance suffers. When insufficient features are tracked, the EKF and IMM performances suffer.

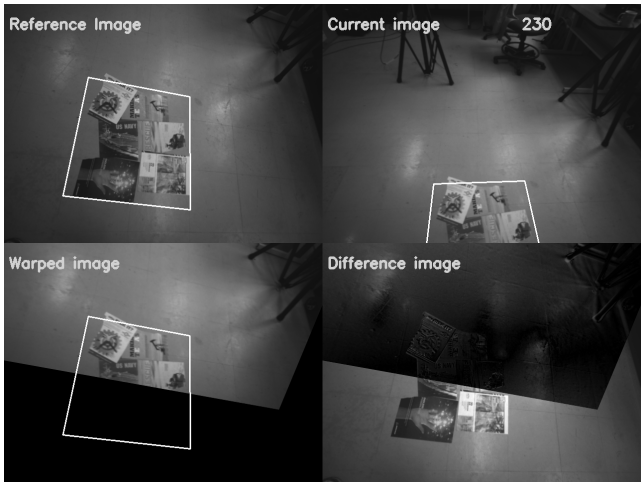

Fig. 6: Experimental method that consists of tracking features on the ground. The white square is used to visualize the estimated homography. The frame at the illustrated timestamp has the pattern, which is partially out of camera view, approximately realigned with the reference frame by the current homography matrix estimate. The current image is warped by the homography estimate on the bottom left.

to not impact the IMM's performance drastically.

The error r_k defined in (33) is used for performance evaluation. In Table II, it can be seen that in the recorded trajectories, the IMM has the best performance in seven out of nine trajectories, while the observer is the best in the remaining two. The *EKF loose* also outperforms the observer in four out of nine trajectories, and even having a similar performance to the IMM in one trajectory. The *EKF tight* is omitted from the experimental results because in most of the experimental trials it diverged.

To explain the varying performance across the recorded trajectories, Figure 5 shows the performance of the estimators in two trajectories plagued with assumption breaks and occlusions. It can be observed how in Trajectory 4, in the time frames where the assumption that $\mathbf{s}_a$ is constant is violated to a larger degree, the observer's error is bigger than the IMM, since the IMM can adapt its gain to handle the assumption violation. The *EKF loose* has similar performance to the IMM in those time frames because it always has low confidence in the process model, but the performance is not as good when the assumption is respected, as shown in Section IV and Table II results. Trajectory 9 is challenging, with many occlusions and

assumption violations. In the period going from $14\,\mathrm{s}$ to $17\,\mathrm{s}$ where the assumption is violated for an extended period of time, the observer performs poorly again. It appears from the rest of the trajectories, that when the assumption is broken for a significant period, the observer struggles.

During occlusions in both Trajectories 4 and 9, the estimators utilize the process model and the rate-gyro measurements to propagate the homography estimate, and thus provide an estimate that otherwise would have been impossible to provide given only images. The *EKF loose* has low confidence in the process model, causing the uncertainty on the estimate to grow significantly during an occlusion, and it can struggle to converge to a good estimate when enough measurements are available again. The IMM offers a compromise, by having an adaptive covariance, and not completely disregarding the process model. In Trajectory 4, around the $9\,\mathrm{s}$ mark, it can be seen how the IMM puts more weight into Model 2, the same present in the *EKF loose*, but not all the weight, thus ensuring a good performance. The IMM can in some cases put significant weight in Model 2, and produce behaviours similar to *EKF loose*, but in general, the IMM performs better during occlusions than the *EKF loose*, as can be seen during the many occlusions of Trajectory 9. Concerning the observer, it usually had the best performance during occlusions, as shown in the second half of Trajectory 9. This can be, in part, explained by the theoretical guarantees of the observer [2], which the EKF and IMM lack [26].

Overall, the IMM offers the best compromise, by better handling occlusions and assumption violations than the *EKF loose* and the observer, respectively. The observer shows good robustness to occlusions, but the IMM shows the best overall performance in varied trajectories while also offering covariance information and robustness during occlusions due to the utilization of a rate gyro, as shown in Figure 5.

VI. Conclusion

In this paper, the problem of estimating homography using rate gyro and camera measurements is addressed. This paper's novelty lies in the use of the Bayesian filtering framework in concert with a simplified process model. In particular, two iterated EKFs are used within an IMM filter. The approach is compared to a nonlinear deterministic observer in both simulation and experiments where an overall better performance is realized. The proposed Bayesian approach offers covariance information, unlike the observer. A Bayesian approach opens the avenue for adaptive filtering, as in this letter, but also post-processing procedures such as outlier removal, smoothing, loop-closure detection, and quality control.

Acknowledgment

Thanks to M. A. Shalaby, C. Cossette, and M. Cohen for their helpful feedback. Thanks to J. Trumpf, R. Mahony, and T. Hamel for motivating homography estimation research.

References

[1] S. Benhimane and E. Malis, "Homography-based 2d visual tracking and servoing," *The Int. J. of Robotics Research*, vol. 26, no. 7, pp. 661–676, 2007.

[2] M.-D. Hua, J. Trumpf, T. Hamel, R. Mahony, and P. Morin, "Feature-based recursive observer design for homography estimation and its application to image stabilization," *Asian J. of Control*, vol. 21, no. 4, pp. 1443–1458, 2019.

[3] V. Grabe, H. H. Bülthoff, D. Scaramuzza, and P. R. Giordano, "Nonlinear ego-motion estimation from optical flow for online control of a quadrotor uav," *The Int. J. of Robotics Research*, vol. 34, no. 8, pp. 1114–1135, 2015.

[4] M.-D. Hua, N. Manerikar, T. Hamel, and C. Samson, "Attitude, linear velocity and depth estimation of a camera observing a planar target using continuous homography and inertial data," in *2018 IEEE Int. Conf. on Robotics and Auto. (ICRA)*, Brisbane, Australia.

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

[6] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[7] A. Agarwal, C. Jawahar, and P. Narayanan, "A survey of planar homography estimation techniques," *Centre for Visual Information Technology, Tech. Rep. IIIT/TR/2005/12*, 2005.

[8] J. Y. Kaminski and A. Shashua, "Multiple view geometry of general algebraic curves," *Int. J. of Computer Vision*, vol. 56, pp. 195–219, 2004.

[9] S. Benhimane and E. Malis, "Real-time image-based tracking of planes using efficient second-order minimization," in *2004 IEEE/RSJ Int. Conf. on Intell. Robots and Systems (IROS)*, 2004.

[10] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep image homography estimation," *arXiv preprint arXiv:1606.03798*, 2016.

[11] T. Nguyen, S. W. Chen, S. S. Shivakumar, C. J. Taylor, and V. Kumar, "Unsupervised deep homography: A fast and robust homography estimation model," *IEEE Robotics and Auto. Letters*, vol. 3, no. 3, pp. 2346–2353, 2018.

[12] T. Hamel, R. Mahony, J. Trumpf, P. Morin, and M.-D. Hua, "Homography estimation on the special linear group based on direct point correspondence," in *2011 50th IEEE Conf. on Decision and Control and European Control Conf.*, Orlando, FL.

[13] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge University Press, 2013, vol. 3.

[14] F. Caballero, L. Merino, J. Ferruz, and A. Ollero, "Homography based kalman filter for mosaic building. applications to uav position estimation," in *Proceedings 2007 IEEE Int. Conf. on Robotics and Auto.*, Rome, Italy.

[15] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2001.

[16] X. R. Li and Y. Bar-Shalom, "A recursive multiple model approach to noise identification," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 30, no. 3, pp. 671–684, 1994.

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

[18] T. D. Barfoot, *State estimation for robotics*. Cambridge University Press, 2017.

[19] B. C. Hall and B. C. Hall, *Lie groups, Lie algebras, and representations*. Springer, 2013.

[20] E. Eade, "Lie groups for computer vision," *Cambridge Univ., Cambridge, UK, Tech. Rep*, vol. 2, 2014.

[21] J. Ćesić, I. Marković, and I. Petrović, "Mixture reduction on matrix lie groups," *IEEE Signal Processing Letters*, vol. 24, no. 11, pp. 1719–1723, 2017.

[22] J. Farrell, *Aided navigation: GPS with high rate sensors*. McGraw-Hill, Inc., 2008.

[23] M. A. Skoglund, G. Hendeby, and D. Axehill, "Extended kalman filter modifications based on an optimization view point," in *2015 18th Int. Conf. on Information Fusion (Fusion)*. Washington, DC: IEEE.

[24] K. MacTavish and T. D. Barfoot, "At all costs: A comparison of robust cost functions for camera correspondence outliers," in *2015 12th Conf. on Computer and Robot Vision*. Halifax, NS: IEEE.

[25] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 Int. Conf. on Computer Vision*. Barcelona, Spain: IEEE.

[26] A. Barrau and S. Bonnabel, "The invariant extended kalman filter as a stable observer," *IEEE Trans. on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2016.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.