Hybrid-MSCEqF: An Equivariant Visual-Inertial Navigation System

Alessandro Fornasier[1], Martin Scheiber[1], Pieter van Goor[2], Robert Mahony[2] and Stephan Weiss[1]

***Abstract*—This paper presents the Hybrid-MSCEqF. A real-time suitable extension of the recently introduced multi state constraint equivariant filter (MSCEqF), aiming at reducing the accumulated drift in unobservable states and handling static and quasi-static scenarios for a wide-range of mobile robotics applications. We introduce a novel equivariant formulation of the zero velocity update (ZVU) together with an extended symmetry for the visual-inertial navigation problem to include fixed visual landmarks into the filter state and define an update routine that achieves third-order linearization error. The Hybrid-MSCEqF is successfully deployed on a Raspberry Pi4 for real-time closed-loop control of a resource-constrained aerial platform.**

I. Introduction and related work

Visual-inertial odometry (VIO) stands as a key technology in mobile robotics, autonomous vehicles, and augmented reality, providing accurate ego-motion estimation by fusing visual and inertial sensor data.

In the realm of filter-based frameworks for visual-inertial odometry, the multi state constraint Kalman filter (MSCKF) introduced by *Mourikis and Roumeliotis* [1], has seen numerous modifications, including first estimate jacobian (FEJ) and observability constraint (OC) filters to ensure consistency [2], [3], [4], [5], stereo versions [6], [7], uncertainty propagation through the feature triangulation [8], [9], and invariant approaches [10], [11], [12].

Recently, a novel methodology for filter design based on the theory of equivariant filter (EqF) [13], [14], [15] for biased inertial navigation systems [16], [17], [18] has demonstrated superior performance in terms of robustness to wrong initialization, transient behavior, and consistency properties. Building on these findings, van Goor *et al.* proposed a symmetry for fixed landmark measurements in the context of VIO [19], [20]. In our previous work [21], we leveraged the idea of the multi state constraint in the context of equivariant filtering and introduced the MSCEqF.

As advocated in our prior work [21], a *people's visual-inertial odometry* is an algorithm whose operation requires minimal knowledge, little to no tuning, and exhibits reliable performance across many different real-world scenarios. Despite the plateau reached by state-of-the-art visual-inertial navigation system (VINS) in terms of accuracy and precision, considerable room for improvement remains in enhancing robustness across many scenarios and ensuring ease of deployment in practical settings. This paper presents the *Hybrid-MSCEqF* as the next step towards the *people's VIO*. Specifically, we propose a real-time suitable extension of the MSCEqF to minimize the accumulated drift and to handle static and quasi-static scenarios commonly encountered in mobile robotics applications. The main contributions of this work include:

- Development of the equivariant formulation of the Hybrid-MSCEqF as an extension of the MSCEqF that incorporates persistent visual features as part of its state exploiting the polar symmetry of $\mathbf{SOT}(3)$ for fixed landmark measurements.
- Derived an equivariant persistent features update routine achieving third-order linearization error.
- Introduction of a novel zero velocity update formulation that exploits the equivariance of the system.
- Validation of the proposed algorithm through real-world experiments and deployment on a Raspberry Pi4 for closed-loop control of a resource-constrained aerial platform.

II. The visual Localization problem

Consider a mobile platform equipped with an inertial measurement unit (IMU) providing biased acceleration and angular velocity measurements, $\boldsymbol{w} = (\boldsymbol{\omega}, \boldsymbol{a}) \in \mathbb{R}^6$, and a camera observing global visual features. Let $\mathbf{T} = (\mathbf{R}, \boldsymbol{v}, \boldsymbol{p}) \in \mathcal{SE}_2(3)$ be the extended pose of the system. $\mathbf{R}$, $\boldsymbol{p}$ and $\boldsymbol{v}$ correspond respectively to the rigid body orientation, the global position, and global velocity of the IMU. Define $\mathbf{P} = (\mathbf{R}, \boldsymbol{p}) \in \mathcal{SE}(3)$ to be the pose of the system. Define $\boldsymbol{b} = (\boldsymbol{b_\omega}, \boldsymbol{b_a}, \boldsymbol{b_\nu}) \in \mathbb{R}^9$ to be the gyroscope, accelerometer, and virtual velocity [16], [18] biases, respectively. Let g denote the magnitude of the acceleration due to gravity, and let $\boldsymbol{e}_3 \in \mathbb{R}^3$ denote the direction of gravity in the global frame. Finally, define $\mathbf{S} \in \mathcal{SE}(3)$ to be the camera extrinsic calibration, and $\mathbf{K} \in \mathcal{IN}(3)$ be the camera intrinsic calibration.

Define $u = (\boldsymbol{w}, \boldsymbol{\tau}) \in \mathbb{L} \subset \mathbb{R}^{18}$ to be the system's input. Define $\mathbf{W} = \boldsymbol{w}^\wedge$ and $\mathbf{B} = \boldsymbol{b}^\wedge$. Then, without loss of generality, the visual-inertial navigation system defined in [21] is

[1]Alessandro Fornasier, Martin Scheiber and Stephan Weiss are with the Control of Networked Systems Group, University of Klagenfurt, Austria. {name.surname}@ieee.org

[2]Pieter van Goor is with the Robotics and Mechatronics group at the university of Twente. {p.c.h.vangoor@}utwente.nl

[3]Robert Mahony is with the System Theory and Robotics Lab, Australian Centre for Robotic Vision, Australian National University, Acton ACT 2601, Australia. {name.surname}@anu.edu.au

extended with a single visual feature $\boldsymbol{p}_f \in \mathbb{R}^3$ as follows.

$$\dot{\mathbf{T}} = \mathbf{T}\left(\mathbf{W} - \mathbf{B} + \mathbf{N}\right) + \left(\mathbf{G} - \mathbf{N}\right)\mathbf{T}, \tag{1a}$$
$$\dot{\boldsymbol{b}} = \boldsymbol{\tau}, \tag{1b}$$
$$\dot{\mathbf{S}} = \mathbf{0}^{\wedge}, \tag{1c}$$
$$\dot{\mathbf{K}} = \mathbf{0}^{\wedge}, \tag{1d}$$
$$\dot{\boldsymbol{p}}_f = \mathbf{0}. \tag{1e}$$

$\boldsymbol{\tau}$ is used to model the deterministic dynamics of the biases and it is considered zero in our formulation.

Define $\xi_I = (\mathbf{T}, \boldsymbol{b}) \in \mathcal{SE}_2(3) \times \mathbb{R}^9$ to be the inertial navigation state. Define $\xi_S = (\mathbf{S}, \mathbf{K}) \in \mathcal{SE}(3) \times \mathcal{IN}(3)$ to be the camera calibration state. Define $\xi_f = \boldsymbol{p}_f \in \mathbb{R}^3$ to be the persistent feature state. Then the full system state is defined as $\xi = (\xi_I, \xi_S, \xi_f) \in \mathcal{M} \coloneqq \mathcal{SE}_2(3) \times \mathbb{R}^9 \times \mathcal{SE}(3) \times \mathcal{IN}(3) \times \mathbb{R}^3$. Define $u = (\boldsymbol{w}, \boldsymbol{\tau}) \in \mathbb{L} \subset \mathbb{R}^{18}$ to be the system's input.

The camera measurement is modeled as the measurement of the bearing of the feature $\boldsymbol{p}_f$ seen from the camera.

$$h(\xi) = \mathbf{K}\pi\left(\left(\mathbf{PS}\right)^{-1} * \boldsymbol{p}_f\right), \tag{2}$$

where the operation $* : \mathcal{SE}(3) \times \mathbb{R}^3 \to \mathbb{R}^3$ is defined by $\mathbf{P} * \boldsymbol{x} = \mathbf{R}\,\boldsymbol{x} + \boldsymbol{p}$ for all $\mathbf{P} = (\mathbf{R}, \boldsymbol{p}) \in \mathcal{SE}(3)$, $\boldsymbol{x} \in \mathbb{R}^3$. π represents the projection method, and it can either be the projection on the unit plane $\pi_{\mathbb{Z}_1}$, or the projection on the unit sphere $\pi_{\mathbb{S}^2}$.

III. VINS SYMMETRY

The symmetry for the inertial navigation state ξ_I is given by the Tangent group of $\mathbf{SE}_2(3)$: $\mathbf{G_{TG}} \coloneqq \mathbf{SE}_2(3) \ltimes \mathfrak{se}_2(3)$, the symmetry for the extrinsic calibration state is given by the special Euclidean group $\mathbf{SE}(3)$, and the symmetry for the intrinsic calibration state is given by the intrinsics group $\mathbf{IN}$.

The notation employed in the next sections follows those of our previous work [21]. Specifically, we make use of the maps $\Pi(\cdot), \Upsilon(\cdot), \Gamma(\cdot)$, and $\Theta(\cdot)$, introduced in [21, Section II.F].

A. VINS extended symmetry

Define a novel group $\mathbf{G} \coloneqq \mathbf{G_{TG}} \times \mathbf{SE}(3) \times \mathbf{IN} \times \mathbf{SOT}(3)$ termed **VINS** group. Elements of the **VINS** group $\mathbf{G}$ and the state action $\phi : \mathbf{G} \times \mathcal{M} \to \mathcal{M}$ are defined as follows:

$$X = \left(\left(D, \delta\right), E, L, Q\right) \in \mathbf{G}, \tag{3}$$
$$X^{-1} = \left(\left(D^{-1}, -\mathrm{Ad}_{D^{-1}}\left[\delta\right]\right), E^{-1}, L^{-1}, Q^{-1}\right) \in \mathbf{G}, \tag{4}$$
$$\begin{aligned}\phi\left(X, \xi\right) \coloneqq \Big(&\mathbf{T}D, \mathbf{Ad}^{\vee}_{D^{-1}}\left(\boldsymbol{b} - \delta^{\vee}\right), C^{-1}\mathbf{S}E, \mathbf{K}L, \\ &\mathbf{P}\,\mathbf{S}E * \left(Q^{-1}\left(\left(\mathbf{P}\,\mathbf{S}\right)^{-1} * \boldsymbol{p}_f\right)\right)\Big) \in \mathcal{M},\end{aligned} \tag{5}$$

where $D = (A, a, b) \in \mathbf{SE}_2(3)$ such that $A \in \mathbf{SO}(3)$, $a, b \in \mathbb{R}^3$. Define the subgroup elements and $C = \Theta(D) \in \mathbf{SE}(3)$. Define $E \in \mathbf{SE}(3)$, $L \in \mathbf{IN}$, and finally, define $Q = (\mathrm{Q}, \mathrm{q}), Q^{-1} = (\mathrm{Q}^{\intercal}, \frac{1}{\mathrm{q}}) \in \mathbf{SOT}(3)$. Note that $\mathbf{P}\,\mathbf{S}E * (Q^{-1}((\mathbf{P}\,\mathbf{S})^{-1} * \boldsymbol{p}_f))$ is a concatenation of roto-translation actions of $\mathbf{SE}(3)$, and rotation and scaling actions of $\mathbf{SOT}(3)$ on $\mathbb{R}^3$.

The lift presented in [21] is extended as follows.

Theorem III.1. *Define* $\Lambda_1, \Lambda_2 : \mathcal{M} \times \mathbb{L} \to \mathfrak{se}_2(3)$, $\Lambda_3 : \mathcal{M} \times \mathbb{L} \to \mathfrak{se}(3)$, *and* $\Lambda_4 : \mathcal{M} \times \mathbb{L} \to \mathfrak{in}$ *as follows:*

$$\Lambda_1\left(\xi, u\right) \coloneqq \left(\mathbf{W} - \mathbf{B} + \mathbf{N}\right) + \mathbf{T}^{-1}\left(\mathbf{G} - \mathbf{N}\right)\mathbf{T}, \tag{6a}$$
$$\Lambda_2\left(\xi, u\right) \coloneqq \mathrm{ad}_{\boldsymbol{b}^{\wedge}}\left[\Lambda_1\left(\xi, u\right)\right] - \boldsymbol{\tau}^{\wedge}, \tag{6b}$$
$$\Lambda_3\left(\xi, u\right) \coloneqq \mathrm{Ad}_{\mathbf{S}^{-1}}\left[\Upsilon\left(\Lambda_1\left(\xi, u\right)\right)\right], \tag{6c}$$
$$\Lambda_4\left(\xi, u\right) \coloneqq \mathbf{0}^{\wedge}. \tag{6d}$$

Let $\Lambda_3(\xi, u) = (\boldsymbol{\Omega}, \boldsymbol{\omega}) \in \mathfrak{se}(3)$, *with* $\boldsymbol{\Omega} \in \mathfrak{so}(3)$ *and* $\boldsymbol{\omega} \in \mathbb{R}^3$. *Let* $\boldsymbol{c}_f = \left(\mathbf{P}\,\mathbf{S}\right)^{-1} * \boldsymbol{p}_f$. *Define the map* $\Lambda_5 : \mathcal{M} \times \mathbb{L} \to \mathfrak{sot}(3)$ *by*

$$\Lambda_5\left(\xi, u\right) = \left(\boldsymbol{\Omega} + \frac{\boldsymbol{c}_f^{\wedge}\,\boldsymbol{\omega}}{\left\|\boldsymbol{c}_f\right\|^2}, \frac{\boldsymbol{c}_f^{\intercal}\,\boldsymbol{\omega}}{\left\|\boldsymbol{c}_f\right\|^2}\right) \tag{7}$$

Then $\Lambda : \mathcal{M} \times \mathbb{L} \to \mathfrak{g}$ *given by*

$$\Lambda\left(\xi, u\right) \coloneqq \left(\left(\Lambda_1\left(\xi, u\right), \Lambda_2\left(\xi, u\right)\right), \cdots, \Lambda_5\left(\xi, u\right)\right),$$

is an equivariant lift for the system in equation (1)*, with respect to the symmetry group* $\mathbf{G}$.

IV. HYBRID-MSCEQF

Multi state constraint filters are specifically designed to minimize computational complexity. They do not retain visual landmarks as part of their state, resulting in high sensitivity to accumulated drift. In contrast, classical filter-based algorithms incorporate a subset of visual landmarks as part of their state, with the benefit of limiting the accumulated drift, allowing for loop-closures [22], [23], albeit at the expense of increased computational complexity.

In this work, we exploit the polar symmetry of fixed landmark measurements introduced by van Goor *et al.* [24], [25], [26], [19], [20], [27], and propose an extension of the MSCEqF to include a subset of visual landmark as part of its state.

A. Equivariant persistent feature update

We exploit the equivariance of visual features projected onto the unit-sphere given by the polar symmetry [20] to derive an update routine for the Hybrid-MSCEqF that achieves third order third-order linearization error. In doing so, we assume the camera intrinsic parameters $\mathbf{K}$ as fixed and only update them during the multi state constraint update routine [21].

Lemma IV.1. *Consider the configuration output in equation* (2) *when the projection on the unit sphere* $\pi_{\mathbb{S}^2}$ *is chosen:*

$$h(\xi) = \pi_{\mathbb{S}^2}\left(\left(\mathbf{PS}\right)^{-1} * \boldsymbol{p}_f\right) \in \mathcal{N} \coloneqq \mathbb{S}^2. \tag{8}$$

Applying the action ϕ of the symmetry group in equation (5) *yields*

$$\begin{aligned} h(\phi_X(\xi)) &= \pi_{\mathbb{S}^2}\left((\mathbf{P}\,\mathbf{S}E)^{-1}\,\mathbf{P}\,\mathbf{S}E * \left(Q^{-1}\left((\mathbf{P}\,\mathbf{S})^{-1} * \boldsymbol{p}_f\right)\right)\right) \\ &= \pi_{\mathbb{S}^2}\left(Q^{-1}\left((\mathbf{P}\,\mathbf{S})^{-1} * \boldsymbol{p}_f\right)\right) \\ &= \pi_{\mathbb{S}^2}\left(\frac{1}{\mathrm{q}}\mathrm{Q}^{\mathsf{T}}\left((\mathbf{P}\,\mathbf{S})^{-1} * \boldsymbol{p}_f\right)\right) \\ &= \mathrm{Q}^{\mathsf{T}}\pi_{\mathbb{S}^2}\left((\mathbf{P}\,\mathbf{S})^{-1} * \boldsymbol{p}_f\right) \\ &= \mathrm{Q}^{\mathsf{T}}h(\xi). \end{aligned}$$

Let $y \in \mathcal{N}$ be a measurement defined according to the model in equation (8)*. Define $\rho : \mathbf{G} \times \mathcal{N} \to \mathcal{N}$ as*

$$\rho(X, y) \coloneqq \mathrm{Q}^{\mathsf{T}} y. \tag{9}$$

Then, the configuration output defined in equation (8) *is equivariant.*

Let $\mathring{y} \coloneqq h(\mathring{\xi})$, then local coordinates for the output are defined as follows:

$$\delta(y) = \mathring{y}^{\wedge} y \in \mathbb{R}^3. \tag{10}$$

The equivariance of the output in equation (8) allows us to achieve third-order linearization error through the equivariant output approximation [14], [18], hence the $\mathbf{C}^{\star}$ matrix is written

$$\begin{aligned} \mathbf{C}^{\star}\varepsilon &= \frac{1}{2}\mathring{y}^{\wedge}\left(\mathring{y} + \hat{\mathrm{Q}}y\right)^{\wedge}\varepsilon_Q \\ &= \begin{bmatrix}\mathbf{0}_{3\times 9} & \mathbf{0}_{3\times 6} & \mathbf{0}_{3\times 6} & \mathbf{0}_{3\times 4} & \frac{1}{2}\mathring{y}^{\wedge}\left(\mathring{y} + \hat{\mathrm{Q}}y\right)^{\wedge} & \mathbf{0}_{3\times 1}\end{bmatrix}\varepsilon. \end{aligned}$$

V. Equivaraint zero velocity update

For multi state constraint filters type, different strategies are proposed to better handle zero motion scenarios. Common approaches include optimized keyframe section [28], [29] and zero velocity update (ZVU) [3]. While the former seeks to optimize the selection of cloned poses in the sliding window to always ensure sufficient parallax and hence a successful feature triangulation, the latter seeks to identify zero motion scenarios through heuristics and perform a zero velocity filter update.

Here, we exploit an extended pose measurement with constant position, rotation, and zero velocity to define a novel *equivariant zero velocity update* formulation, which models the stationarity of the platform rather than purely zero velocity.

Lemma V.1. *Define the zero velocity update measurement model $h(\xi)$ as follows:*

$$h(\xi) = \mathbf{T} = (\mathbf{R}, \boldsymbol{v}, \boldsymbol{p}) \in \mathcal{N} \coloneqq \mathcal{SE}_2(3). \tag{11}$$

Let $y = (\hat{\mathbf{R}}, \mathbf{0}, \hat{\boldsymbol{p}}) \in \mathcal{N}$ be a measurement defined according to the model in equation (11)*, define $\rho : \mathbf{G} \times \mathcal{N} \to \mathcal{N}$ as*

$$\rho(X, y) \coloneqq yD. \tag{12}$$

Then, the configuration output defined in equation (11) *is equivariant.*

Local coordinates for the output are chosen to be logarithmic coordinates of $\mathbf{SE}_2(3)$, hence define

$$\delta(y) = \log_{\mathbf{SE}_2(3)}(\mathring{\mathbf{T}}^{-1}y). \tag{13}$$

Exploiting the equivariance of the ZVU output, the $\mathbf{C}^{\star}$ matrix for the zero velocity update is written

$$\begin{aligned} \mathbf{C}^{\star}\varepsilon &= \frac{1}{2}\left(\left(\mathbf{I}_9 + \mathring{\mathbf{T}}^{-1}y\hat{D}^{-1}\right)\varepsilon_D^{\wedge}\right)^{\vee} = \varepsilon_D \\ &= \begin{bmatrix}\mathbf{I}_9 & \mathbf{0}_{9\times 6} & \mathbf{0}_{9\times 6} & \mathbf{0}_{9\times 4} & \mathbf{0}_{9\times 6} & \cdots & \mathbf{0}_{9\times 6}\end{bmatrix}\varepsilon. \end{aligned}$$

In our approach, stationary detection is performed based on a threshold of the image disparity over a rolling window of image data. This approach can be extended with robust methodologies such as statistical stationarity test [30] or frequency-domain techniques [31] on a rolling window of IMU data.

VI. Implementation details

The proposed framework is implemented as a stand-alone C++ library and used with a ROS wrapper for the experiments.

The vision frontend is implemented utilizing the OpenCV library [32]. The feature detector can be chosen between *Good Feature to Track (GFTT)* and *Fast*. The input image is divided into a $N \times M$ grid, and the feature extraction is executed in parallel. A minimum pixel distance between detected features is imposed; furthermore, sub-pixel refinement is performed. Detected features are subsequently tracked between consecutive images using the pyramidal Lucas-Kanade algorithm [33] with the previous flow as the initial value, and RANSAC is used to remove outliers.

The estimation back-end is implemented utilizing a self-developed Lie group library: Lie++. A High-level overview of the filter logic is shown in the flowchart in figure 1. At the beginning, the filter tries to initialize the state origin $\mathring{\xi}$. This is done by collecting a window of IMU measurements during a still phase and computing the roll and pitch of the platform. Origin position and velocity are set to zero, and origin calibration is set to the provided values. Once the state origin is initialized, the feature extraction and matching, and the propagation are executed in parallel on different threads. Once the propagation is performed, either a stochastic cloning or a zero velocity update is performed. When the zero velocity update is not active or the platform is moving, the multi state constraint update is performed. Furthermore, active persistent features are updated. Finally, features that are not yet persistent but belong to long tracks, hence, that have been tracked in every active clone, are initialized as persistent.

VII. Closed-loop control

To evaluate the real-time performance of the Hybrid-MSCEqF, we run the proposed algorithm online for autonomous closed-loop control of the aerial platform.

The platform used for the experiments is a Twins twinFOLD SCIENCE quadcopter equipped with a Raspberry Pi4 with 4GB of RAM as a companion board. The IMU data was collected from the quadcopter autopilot, a Holybro Pixhawk 4

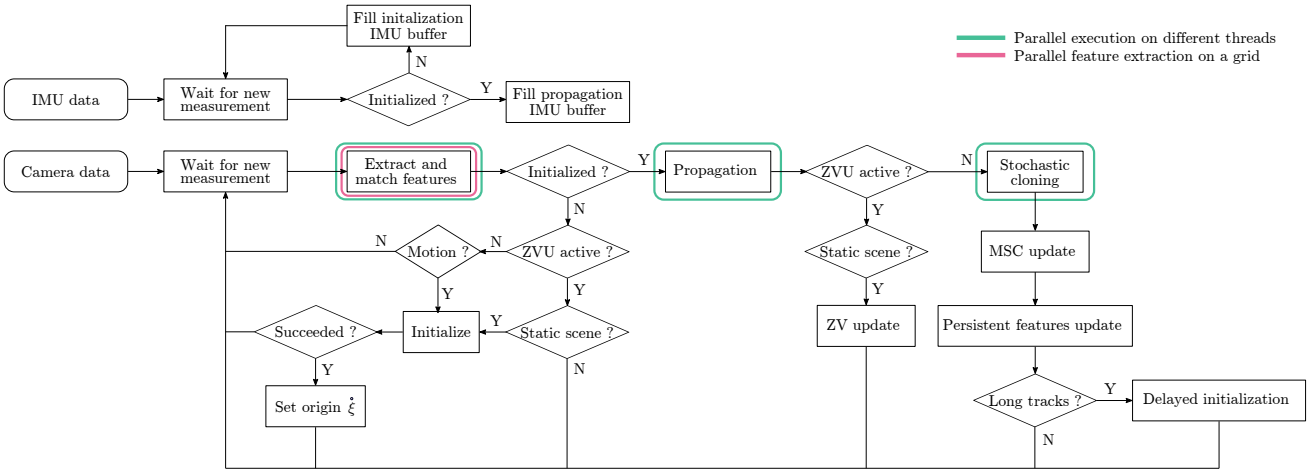

Figure 1. Flowchart of the proposed Hybrid-MSCEqF.

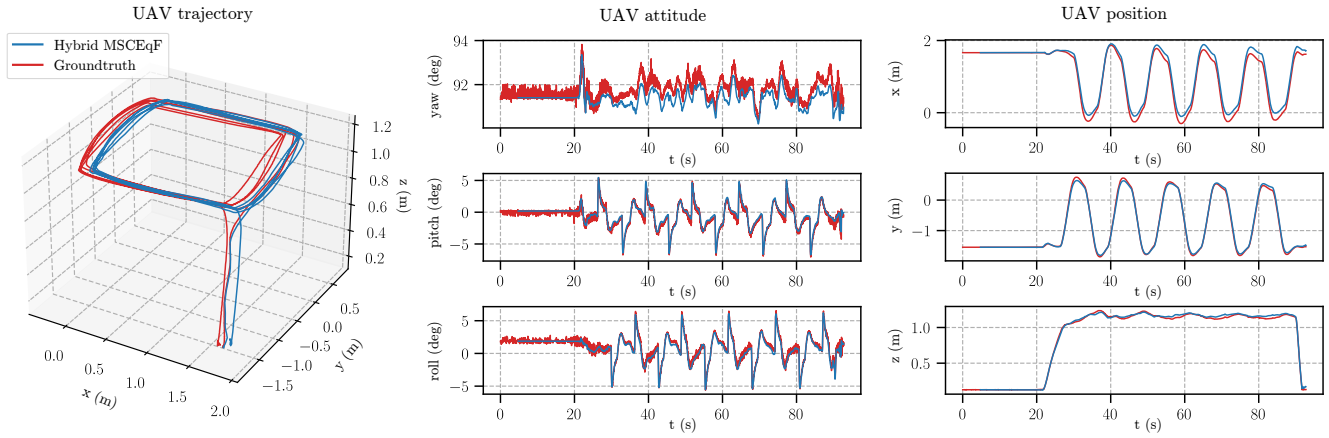

Figure 2. UAV trajectory, attitude, and position plots for the closed-loop control experiment. In red is the groundtruth and in blue is the estimate produced by the proposed Hybrid-MSCEqF. The algorithm was run on a Raspberry Pi4, and its estimate was used for closed-loop control of an autonomous quadcopter.

at 200Hz, whereas the image data was collected at 15Hz from a 60° downward-looking Matrix Vision BlueFox MLC200WG global-shutter camera with 752×480 CMOS grayscale sensor and a lens with diagonal field of view of 120°.

The experiment consists of flying 5 times a 2.5m×1.5m square trajectory for a total length of 40m, excluding takeoff and landing. For this experiment, the Hybrid-MSCEqF was configured to track 80-100 features for a maximum of 9 keyframes. The zero velocity update was always enabled, and the maximum number of persistent features was set to 10.

Aligned trajectory, position, and orientation plots of the estimate, as well as the groundtruth obtained with a motion capture system, are reported in figure 2. The alignment was done using only the first measurement [34], rather than finding an optimal alignment that minimizes the error.

The root mean square error (RMSE) of the absolute trajectory error (ATE) for this experiment sets at $\mathbf{0.71}°$ and $\mathbf{0.14m}$. Moreover, to evaluate the accumulated drift, we calculated the mean position and angular error for the last second of the trajectory. We obtained a final angular error of $\mathbf{0.73}°$ and a final position error of $\mathbf{0.14m}$ which corresponds to a drift of $\mathbf{0.35}\%$ of the total trajectory length.

VIII. Conclusion

This paper presented the *Hybrid-MSCEqF*, an equivariant visual-inertial odometry framework extending the MSCEqF with a novel equivariant formulation of the zero velocity update and the polar symmetry for fixed landmark measurements. With the proposed algorithm, we address the need for a VIO algorithm that achieves state-of-the-art performance and superior robustness while handling static and quasi-static scenarios commonly encountered in mobile robotics.

The proposed algorithm demonstrated successful deployment on a Raspberry Pi4 for closed-loop control of a resource-constrained aerial platform, exhibiting good performance without requiring extensive fine-tuning of the filter's parameters.

The Hybrid-MSCEqF together with the Lie++ library, will be made available to the community for direct use and comparison against other approaches at the following links: https://github.com/aau-cns/MSCEqF and https://github.com/aau-cns/Lie-plusplus.

REFERENCES

[1] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3565–3572, 2007.

[2] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "OpenVINS: A Research Platform for Visual-Inertial Estimation," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 4666–4672, 5 2020.

[3] X. QIU, H. ZHANG, and W. FU, "Lightweight hybrid visual-inertial odometry with closed-form zero velocity update," *Chinese Journal of Aeronautics*, vol. 33, no. 12, pp. 3344–3359, 12 2020.

[4] C. Chen, Y. Yang, P. Geneva, and G. Huang, "FEJ2: A Consistent Visual-Inertial State Estimator Design," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 9506–9512, 2022.

[5] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Consistency analysis and improvement of vision-aided inertial navigation," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 158–176, 2014.

[6] K. Sun, K. Mohta, B. Pfrommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, "Robust Stereo Visual Inertial Odometry for Fast Autonomous Flight," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, 4 2018.

[7] S. Bahnam, S. Pfeiffer, and G. C. De Croon, "Stereo Visual Inertial Odometry for Robots with Limited Computational Resources," *IEEE International Conference on Intelligent Robots and Systems*, pp. 9154–9159, 2021.

[8] A. Solin, S. Cortés, E. Rahtu, and J. Kannala, "PIVO: Probabilistic inertial-visual odometry for occlusion-robust navigation," *Proceedings - 2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018*, vol. 2018-January, pp. 616–625, 5 2018.

[9] O. Seiskari, P. Rantalankila, J. Kannala, J. Ylilammi, E. Rahtu, and A. Solin, "HybVIO: Pushing the Limits of Real-time Visual-inertial Odometry," *Proceedings - 2022 IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2022*, pp. 287–296, 2022.

[10] S. Heo and C. G. Park, "Consistent EKF-Based Visual-Inertial Odometry on Matrix Lie Group," *IEEE Sensors Journal*, vol. 18, no. 9, pp. 3780–3788, 5 2018.

[11] K. Wu, T. Zhang, D. Su, S. Huang, and G. Dissanayake, "An invariant-EKF VINS algorithm for improving consistency," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2017-September, pp. 1578–1585, 12 2017.

[12] Y. Yang, C. Chen, W. Lee, and G. Huang, "Decoupled Right Invariant Error States for Consistent Visual-Inertial Navigation," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1627–1634, 4 2022.

[13] P. Van Goor, T. Hamel, and R. Mahony, "Equivariant Filter (EqF): A General Filter Design for Systems on Homogeneous Spaces," *Proceedings of the IEEE Conference on Decision and Control*, vol. 2020-Decem, no. Cdc, pp. 5401–5408, 2020.

[14] P. van Goor, T. Hamel, and R. Mahony, "Equivariant Filter (EqF)," *IEEE Transactions on Automatic Control*, 6 2022.

[15] R. Mahony, P. van Goor, and T. Hamel, "Observer design for nonlinear systems with equivariance," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, pp. 221–252, 2022.

[16] A. Fornasier, Y. Ng, R. Mahony, and S. Weiss, "Equivariant Filter Design for Inertial Navigation Systems with Input Measurement Biases," *2022 International Conference on Robotics and Automation (ICRA)*, pp. 4333–4339, 5 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9811778/

[17] A. Fornasier, Y. Ng, C. Brommer, C. Bohm, R. Mahony, and S. Weiss, "Overcoming Bias: Equivariant Filter Design for Biased Attitude Estimation With Online Calibration," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12 118–12 125, 10 2022.

[18] A. Fornasier, Y. Ge, P. van Goor, R. Mahony, and S. Weiss, "Equivariant Symmetries for Inertial Navigation Systems," *arXiv preprint arXiv:2309.03765*, 9 2023. [Online]. Available: https://arxiv.org/abs/2309.03765v1

[19] P. van Goor and R. Mahony, "An Equivariant Filter for Visual Inertial Odometry," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2021-May, pp. 1875–1881, 2021.

[20] ——, "EqVIO: An Equivariant Filter for Visual-Inertial Odometry," *IEEE Transactions on Robotics*, pp. 1–19, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10179117/

[21] A. Fornasier, P. v. Goor, E. Allak, R. Mahony, and S. Weiss, "MSCEqF: A Multi State Constraint Equivariant Filter for Vision-aided Inertial Navigation," *IEEE Robotics and Automation Letters*, 1 2023.

[22] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robotics and Automation Magazine*, vol. 13, no. 2, pp. 99–108, 6 2006.

[23] P. Geneva, K. Eckenhoff, and G. Huang, "A linear-complexity EKF for visual-inertial navigation with loop closures," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2019-May, pp. 3535–3541, 5 2019.

[24] P. V. Goor, R. Mahony, T. Hamel, and J. Trumpf, "A Geometric Observer Design for Visual Localisation and Mapping," *Proceedings of the IEEE Conference on Decision and Control*, vol. 2019-December, pp. 2543–2549, 12 2019.

[25] P. Van Goor, R. Mahony, T. Hamel, and J. Trumpf, "An Observer Design for Visual Simultaneous Localisation and Mapping with Output Equivariance," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9560–9565, 1 2020.

[26] P. van Goor, R. Mahony, T. Hamel, and J. Trumpf, "Constructive observer design for Visual Simultaneous Localisation and Mapping," *Automatica*, vol. 132, p. 109803, 10 2021.

[27] P. Van Goor, "Equivariant Filters for Visual Spatial Awareness," 2023.

[28] D. G. Kottas, K. J. Wu, and S. I. Roumeliotis, "Detecting and dealing with hovering maneuvers in vision-aided inertial navigation systems," *IEEE International Conference on Intelligent Robots and Systems*, pp. 3172–3179, 2013.

[29] E. Allak, A. Hardt-Stremayr, and S. Weiss, "Key-Frame Strategy during Fast Image-Scale Changes and Zero Motion in VIO Without Persistent Features," *IEEE International Conference on Intelligent Robots and Systems*, pp. 6872–6879, 12 2018.

[30] A. Solin, S. Cortes, E. Rahtu, and J. Kannala, "Inertial Odometry on Handheld Smartphones," *2018 21st International Conference on Information Fusion, FUSION 2018*, pp. 1361–1368, 9 2018.

[31] A. Ramanandan, A. Chen, and J. A. Farrell, "Inertial navigation aiding by stationary updates," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 235–248, 3 2012.

[32] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[33] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *IJCAI'81: 7th international joint conference on Artificial intelligence*, vol. 2, 1981, pp. 674–679.

[34] Z. Zhang and D. Scaramuzza, "A Tutorial on Quantitative Trajectory Evaluation for Visual(-Inertial) Odometry," *IEEE International Conference on Intelligent Robots and Systems*, pp. 7244–7251, 12 2018.