MATEC Web of Conferences **139**, 00068 (2017) DOI: 10.1051/matecconf/201713900068
*ICMITE 2017*

# Monocular Visual-Inertial State Estimation for Micro Aerial Vehicles

*Yao* Xiao[1,⋆], *Xiaogang* Ruan[1], *Xiaoping* Zhang[1], and *Pengfei* Dong[1]

[1] *Faculty of Information Technology, Beijing University of Technology*

**Abstract.** Autonomous micro aerial vehicles (MAVs) equipped with onboard sensors, are idea platforms for missions in complex and confined environments for its low cost, small size and agile maneuver. Due to the size, power, weight and computation constraints inherent in the filed of MAVs, monocular visual-inertial system that consist of one camera and an inertial measurement (IMU) are the most suitable sensor suit for MAVs. In this paper, we proposed a monocular visual-inertial algorithm for estimating the state of a MAV. Firstly, the Semi-Direct Visual Odometry (SVO) algorithm used as the vision front-end of our framework was modified so that it can be used for forward-looking camera case. Second, an Error-state Kalman Filter was designed so that it can fuse the output of the SVO and IMU data to estimate the full state of the MAVs. We evaluated the proposed method with EuRoc Dataset and compare the results to the state-of-the-art visual-inertial algorithm, VINS-Mono. Experiments show that our estimator can achieve comparable accurate results.

## 1 Introduction

MAVs are ideal platforms for missions such as exploration, inspecting, search and rescue in complex and confined environments due to its low cost, small size and agile maneuver. To achieve these goals, it is essential that the MAVs is capable of autonomous navigation in unknown environments, which include reliable state estimation, control, environment mapping, planning and obstacles avoidance. Among these, state estimation is the first and most critical components for autonomous flight. Despite GPS-based autonomous flight have been successful apply in commercial MAVs, it is unavailable when operate in indoor or tunnel-link environments due to effects like shadowing.

At present, the most common used sensors for state estimation of MAVs are monocular [1–4], stereo [5–7] and RGB-D [8, 9] cameras as well as laser scanners [10]. However, MAVs usually come with tight size, weight and power constraints, limiting their ability to carry active but heavy sensors such as radars or LiDARs. The dimensions of a small platform also limit its ability to carry stereo or multicamera systems due to insufficient baseline length. As the platform becomes smaller, a monocular visual-inertial navigation system (VINS), consisting of only a low-cost inertial measurement unit (IMU) and a camera, becomes the only viable sensor suite allowing autonomous flights with sufficient environmental awareness [4].

Recent research on VINS for MAVs have yielded a number of significant results[1–4, 11]. Most of these existing approach can be classified into filter-based and optimization-based systems. In [1], a parallel tracking and mapping (PTAM) algorithm from the Simultaneous Localization and Mapping (SLAM) community is used as front-end of the VINS system which estimate a 6DOF with arbitrarily scaled and drifting camera pose measurement, and an Extended Kalman Filter is used as the back-end which can fuse the output of the PTAM and the IMU information to get a full state metric estimation of the MAVs. In [11], Faessler et al. present a system that enables a monocular-vision–based quadrotor to automatically recover from any unknown, initial attitude with significant velocity, such as after loss of visual tracking due to an aggressive maneuver. The images from the camera are processed by means SVO pipeline [12]. The visual-odometry pipeline outputs an un-scaled pose that is then fused with the IMU readings in an Extended Kalman Filter framework (Multi Sensor Fusion, MSF [13]) to compute a metric state estimate. This work have been extended in [3]. However, SVO is a direct-based method and designed for MAVs with downward-looking camera, which made the system cannot used for forward-looking case. In [2], Shen proposed a optimization-based VINS framework for rotorcraft MAVs. This estimator tightly couples the vision and IMU measurements and allows the MAV to execute trajectories at 2 m/s with roll and pitch angles up to 30 degrees. To the best our knowledge, this is the first tightly-coupled nonlinear optimization-based monocular VINS estimator successful apply for autonomous flight. This work have been extended in their work in [4]. However, tightly-coupled nonlinear optimization-based method is more computation-intensive.

In this paper, we proposed a monocular visual-inertial algorithm for estimating the state of a micro aerial vehicles (MAV). The SVO algorithm was modified so that it can be used for forward-looking camera case. And an Error-state Kalman filter was designed which can fuse the output of

⋆e-mail: xiaoyao1103@mails.bjut.edu.cn

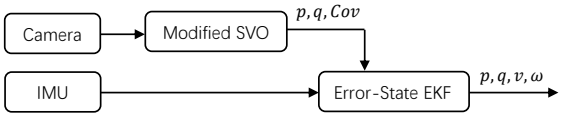


**Figure 1.** Sofware arichiteture of our algorithm

the SVO and IMU data to estimate the full state of the MAVs.

One contribution of this paper is that we modified the original SVO algorithm such that it can be used in the forward-looking camera case. As a second contribution, we formulate a fusion framework based on error-state Kalman filter which can fuse the output of the SVO and the IMU data to get the full state estimate of the MAVs. Compare to the work of [1, 14, 15], we use the SVO as the front end of the frame work which would less computational demand. And compare to the work of [3], our estimator can be run in the case of forward-looking camera. As a third contribution, we compare our propose algorithm against the state-of-the-art visual-inertial algorithm (Mono-VINVS) and demonstrate its comparable performance.

The rest of this paper is organized as follows: Section 2 presents a general overview of the software architecture. Section 3 describes what we have improve from the original SVO; Section 4 describes the details of the design the error-state Kalman filter. Section 4 consists the experiments to evaluate our proposed approach. Finally, Section 6 concludes the work.

## 2 Software Architecture

The Software Architecture of our algorithm is shown in Figure 1. The $p, q, v, \omega$ represent the position, attitude, translation velocity, angular velocity of the MAVs respectively. And $Cov$ represents the covariance of $p$ and $q$. The software consists two components. The first one is the modified SVO, which input is the image scream capture from the camera. The Error-State EKF component fuses the output of the SVO and the IMU data to estimate the full state of the MAVs.

## 3 Modified SVO

### 3.1 Keyframe Selection

The original version of SVO algorithm was designed for downward-looking case. And a keyframe is selected if the Euclidean distance of the new frame relative to all keyframes exceeds 12% of the average scene depth [12]. However, if the camera is facing forwards, the scene depth is very large and no new keyframes are selected.

Generally, a keyframe should be required in there cases:

(1) The field of view changes with a threshold.
(2) The camera translate exceeds a threshold.
(3) The environment light condition changes or the camera exposure time changes.

Compare to the feature-based method, SVO is more sensitive to the light condition change or exposure time change as it is a direct-based method. However, determining thresholds of the three cases is a problem in reality. As a second analyze of the cases above, we could find that all of the cases would cause tracking features to be lost. According to this idea, in this paper we set the keyframe selection strategy as a simple criteria, that is, a keyframe is selected if the percent of the lost tracking feature respect to the last keyframe exceeds 15%.

### 3.2 Tracking Failure Process

In the original version of SVO tracking pipline, the pose of a new coming frame is initialize through sparse model-based image alignment. The camera pose relative to the last frame is found through minimizing the photometric error between pixels corresponding to the projected location of the same 3D points [12]. And if one frame tracking failure, the system will try to do image alignment with the last keyframe. This would be fine if the frame rate is high enough (>50Hz). Experiments showed that one or two continue frame may be very easily failed for tracking due to focus losing or motion blur. In this case, if the SVO are run on low frame rate ($\sim$ 20 Hz), it would be very difficult to re-localzie the new frame and continue the tracking again. This maybe for several reasons:

(1) The image align process dose not wrap the patches used to align for computing speed reasons.
(2) Compare to two consecutive frames, the change of field of view or translation between the last keyframe is more larger.
(3) If more than two consecutive are failed for tracking, the change of field of view or translation between the last keyframe become larger and larger. As a result, it is more and more difficult to re-localize the new frame.

In our modified version of SVO, the strategy we use is that if one frame is failed for tracking, we would drop out the current frame and keep the last frame unchanged, and after the new frame arrive, we would align with the last frame. As we discuss above, the new coming frame would be more easier to re-localize with this strategy.

## 4 Error-State Kalman Filter

The unit quaternion has the lowest dimension of globally nonsigular attitude parameterization and represents the attitude matrix as a homogeneous quadratic function. Due to the quaternion norm constraint, the $4 \times 4$ quaternion covariance is assumed to have rank three [16]. As a result, we cannot simply apply the standard Exetend Kalman filter for the system. In this paper, we use the Error-State Kalman Filter to fuse the IMU data and the output of SVO.

### 4.1 Notations

There are several quaternion conventions. In this paper, the Hamilton convention is used, which is defined as:

$q = q_w + q_x i + q_y j + q_z k$, and $i^2 = j^2 = k^2 = ijk = -1$. The quaternion $q_{ab}$ describes a ratation from frame a to the frame b, and $q_{ac} = q_{ab} \otimes q_{bc}$. $R_{ab}$ is the corresponding rotation matrix of $q_{ab}$.

The subscript w, i, v of the position $p$, quaternion $q$ and velocity $v$ denote the world frame, IMU frame, vision frame, camera frame respectively, such as $p_{\mathrm{wi}}$ means the IMU position of the MAV express in the world frame. We assume the IMU frame is coincide with the body frame, which is the general case in reality. The skew symmetric matrix of $w$ is represented as $[w]_\times$. The hat of a variable $x$ means its estimate value $\hat{x}$.

### 4.2 System Kinematics

We assumed the IMU measurements contain a slowly varying bias b and white Gaussian noise n:

$$\omega = \omega_m - b_\omega - n_\omega \quad a = a_m - b_a - n_a \tag{1}$$

$$\dot{b}_\omega = n_{b_\omega} \quad \dot{b}_a = n_{b_a} \tag{2}$$

Where, $a, \omega$ is the real acceleration and angular velocity respectively. The $a_m, \omega_m$ denotes the measurements value.

Define the system state as

$$X = \begin{bmatrix} p_{\mathrm{wi}}^T & q_{\mathrm{wi}}^T & v_{\mathrm{wi}}^T & b_w^T & b_a^T & \lambda \end{bmatrix} \tag{3}$$

The system kinematics is represented as following differential equations:

$$\dot{p}_{\mathrm{wi}} = v_{\mathrm{wi}} \tag{4}$$

$$\dot{q}_{\mathrm{wi}} = \frac{1}{2}\Omega\left(\omega_m - b_\omega - n_\omega\right) q_{\mathrm{wi}} \tag{5}$$

$$\dot{v}_{\mathrm{wi}} = R_{\mathrm{wi}}(a_m - b_a - n_a) - g \tag{6}$$

$$\dot{b}_w = n_{b_w} \quad \dot{b}_a = n_{b_a} \quad \dot{\lambda} = 0 \tag{7}$$

The nominal-state kinematics is defined as:

$$\dot{\hat{p}}_{\mathrm{wi}} = \hat{v}_{\mathrm{wi}} \tag{8}$$

$$\dot{\hat{q}}_{\mathrm{wi}} = \frac{1}{2}\Omega\left(\omega_m - \hat{b}_\omega\right)\hat{q}_{\mathrm{wi}} \tag{9}$$

$$\dot{\hat{v}}_{\mathrm{wi}} = R_{\mathrm{wi}}\left(a_m - \hat{b}_a\right) - g \tag{10}$$

$$\dot{\hat{b}}_w = 0 \quad \dot{\hat{b}}_a = 0 \quad \dot{\hat{\lambda}} = 0 \tag{11}$$

### 4.3 Error State Kinematics

Define $q_{\mathrm{wi}} = \hat{q}_{\mathrm{wi}} \otimes \delta q_{\mathrm{wi}}$, which means

$$\delta q_{\mathrm{wi}} = q_{\hat{i}i} = \hat{q}_{\mathrm{wi}}^{-1} \otimes q_{\mathrm{wi}} \approx [1 \ \frac{1}{2}\delta\theta_{\mathrm{wi}}]^T \tag{12}$$

Define error state of the system as

$$\delta x = \begin{bmatrix} \delta p_{\mathrm{wi}}^T & \delta\theta_{\mathrm{wi}}^T & \delta v_{\mathrm{wi}}^T & \delta b_w^T & \delta b_a^T & \delta\lambda \end{bmatrix} \tag{13}$$

The error state kinematics is represented as following differential equations:

$$\delta\dot{p}_{\mathrm{wi}} = \delta\hat{v}_{\mathrm{wi}} \tag{14}$$

$$\delta\dot{\theta}_{\mathrm{wi}} = -[\hat{\omega}]_\times \delta\theta_{\mathrm{wi}} - \delta b_\omega - n_\omega \tag{15}$$

$$\delta\dot{v}_{\mathrm{wi}} = -R_{\mathrm{wi}}^T[\hat{a}]_\times \delta\theta_{\mathrm{wi}} - R_{\mathrm{wi}}^T\delta b_a - R_{\mathrm{wi}}^T n_a \tag{16}$$

$$\delta\dot{b}_w = n_{b_w} \quad \delta\dot{b}_a = n_{b_a} \quad \delta\dot{\lambda} = 0 \tag{17}$$

This can be summarized to the linearized continous-time error state equation

$$\delta\dot{x} = F_c\delta x + B_c u \tag{18}$$

with u being the noise vector. Discretize the error state kinematics we can get $F_d$:

$$F_d = \exp\left(F_c\Delta t\right) = I_d + F_c\Delta t + \frac{1}{2!}F_c^2\Delta t + \ldots \tag{19}$$

$$F_d = \begin{bmatrix} I_3 & A & \delta t & B & -C\frac{\Delta t^2}{2} \\ 0 & C & 0 & D & 0 \\ 0 & E & I_3 & F & -E\Delta dt \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{20}$$

Where:

$$A = R_{\mathrm{wi}}^T[\hat{a}]_\times\left(-\frac{\Delta t^2}{2} + \frac{\Delta t^3}{3!}[\omega]_\times - \frac{\Delta t^4}{4!}[\omega]_\times^2\right) \tag{21}$$

$$B = R_{\mathrm{wi}}^T[\hat{a}]_\times\left(\frac{\Delta t^3}{3!} - \frac{\Delta t^4}{4!}[\omega]_\times + \frac{\Delta t^5}{5!}[\omega]_\times^2\right) \tag{22}$$

$$C = I_3 - \Delta t[\omega]_\times + \frac{\Delta t^2}{2}[\omega]_\times^2 \tag{23}$$

$$D = -\Delta t + \frac{\Delta t^2}{2}[\omega]_\times + \frac{\Delta t^3}{3!}[\omega]_\times^2 \tag{24}$$

$$E = R_{\mathrm{wi}}^T C \tag{25}$$

$$F = -A \tag{26}$$

### 4.4 Update

Measurement model of the system is described as

$$z_p = p_{\mathrm{vc}} = R_{\mathrm{vw}}\left(p_{\mathrm{wi}} + R_{\mathrm{wi}}p_{\mathrm{ic}}\right)\lambda + n_p \tag{27}$$

The measurement error can be define as:

$$\begin{aligned} \tilde{z}_q &= z_q - \hat{z}_q \\ &= R_{\mathrm{vw}}\left(p_{\mathrm{wi}} + R_{\mathrm{wi}}p_{\mathrm{ic}}\right)\lambda + n_p \\ &\quad - R_{\mathrm{vw}}\left(\hat{p}_{\mathrm{wi}} + \hat{R}_{\mathrm{wi}}p_{\mathrm{ic}}\right)\hat{\lambda} \end{aligned} \tag{28}$$

Which can be linearized to

$$\tilde{z}_q = H_p\delta x \tag{29}$$

with

$$H_p^T = \begin{bmatrix} R_{\mathrm{vw}}\hat{\lambda} \\ -R_{\mathrm{vw}}\hat{R}_{\mathrm{wi}}[p_{\mathrm{ic}}]_\times\hat{\lambda} \\ 0_{3\times3} \\ 0_{3\times3} \\ 0_{3\times3} \\ \left(R_{\mathrm{vw}}\hat{p}_{\mathrm{wi}} + R_{\mathrm{vw}}\hat{R}_{\mathrm{wi}}\hat{p}_{\mathrm{wi}}\right) \end{bmatrix}^T \tag{30}$$

Measurement model of rotation:

$$z_q = q_{\mathrm{vc}} = q_{\mathrm{vw}} \otimes q_{\mathrm{wi}} \otimes q_{\mathrm{ic}} \tag{31}$$

Which yields for the error measurement:

$$\begin{aligned} \tilde{z_q} &= \left(q_{\mathrm{vw}} \otimes \hat{q}_{\mathrm{wi}} \otimes q_{\mathrm{ic}}\right)^{-1} \otimes \left(q_{\mathrm{vw}} \otimes q_{\mathrm{wi}} \otimes q_{\mathrm{ic}}\right) \\ &= \begin{bmatrix} 1 & 0 \\ 0 & R_{\mathrm{ic}}^T \end{bmatrix}\begin{bmatrix} 1 \\ \delta\theta \end{bmatrix} \end{aligned} \tag{32}$$

### 4.5 Filter Pipeline

(1) Propagate nominal-stated with the IMU measurements according to equations (8)-(11).

(2) Compute the progated covariance matrix according to:

$$P_{k+1|k} = F_d P_{k|k} F_d + Q_d \tag{33}$$

(3) Compute the residual according to equation (28)

(4) Compute the innovation:

$$S = HPH^T + R \tag{34}$$

(5) Compute the Kalman gain:

$$K = PH^T S^{-1} \tag{35}$$

(6) Compute the correction:

$$\hat{\tilde{x}} = K\tilde{z} \tag{36}$$

(7) Correct the propagate nominal-stated, where the attitude is corrected by:

$$\hat{q}_{wi} = \hat{q}_{wi} \otimes \hat{\tilde{x}}_{(q_{\mathrm{wi}})} \tag{37}$$

## 5 Experiment Results

We evaluated the proposed algorithm with the Machine Hall 01 of EuRoc Visual-Inertial Dataset. The dataset are collected on-board a Micro Aerial Vehicle (MAV), which contain stereo images(Aptina MT9V034 global shutter, WVGA monochrome, $2 \times 20$ FPS), synchronized IMU measurements(ADIS16448, angular rate and acceleration, 200 Hz), and accurate ground-truth states. We only use the left camera of the stereo images set.

The results do not contain the comparison of the performance difference between the original version of SVO and the our modified SVO as the the original version of SVO cannot run on this dataset. We compare the performance of the proposed algorithm with ground-truth and the VINS-Mono framework [4].

The whole trajectories are shown in Figure 2. Compare to the ground-truth, both of the proposed method and VINS-Mono exist the drift in the trajectories as the time goes on. It is hard to say which one drift more. However, benefit from the SVO, the trajectory of our propose method nearly return to the starting point at the end of flying. In contrast, the result of VINS-Mono show more drift at the end of the trajectory.

Figure 3 shows the translation error of the trajectories in $x, y, z$ axises respectively. It shows that the VINS-Mono achieve more accurate in $y$ axis. However, proposed algorithm shows less drift on $z$ axis.

The attitude errors expressed in yaw, roll, pitch respectively are showed in Figure 4. It is obvious that the proposed method achieve more accuracy with the attitude estimation compare to the results ofvalidity VINS-Mono.

Figure 5 shows the velocity error. It can be seen that Both the method achieve accurate velocity estimating. It is obvious that there is a abrupt jump in the measurements of our propose methods, this should be caused by tracking failure. However, the vision estimator have successful relocalize the subsequent frames, which shows the effectiveness our modified tracking failure process.

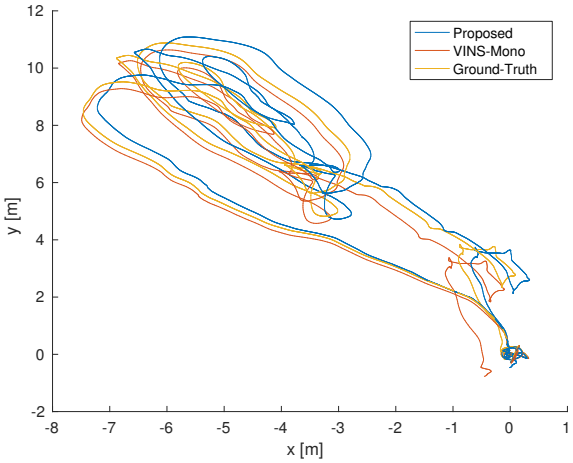


**Figure 2.** Trajectory.

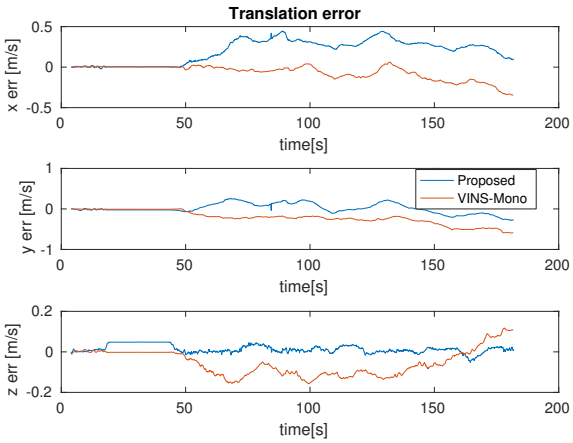


**Figure 3.** Translation error.

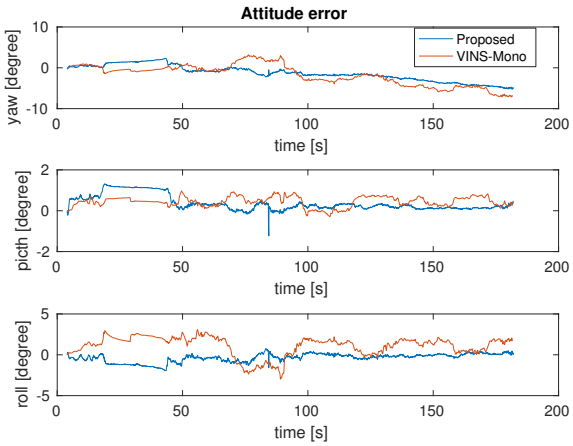


**Figure 4.** Attitude error.

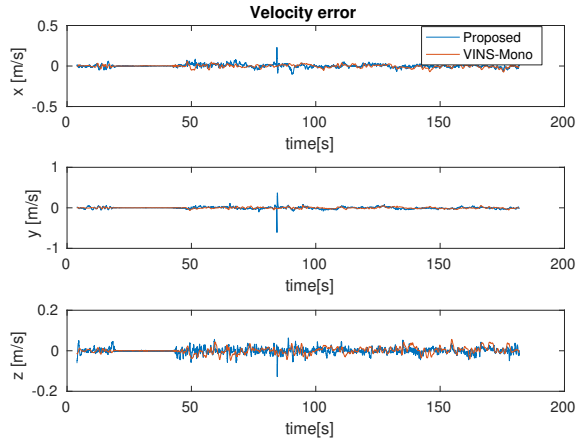


**Figure 5.** Velocity error.

## 6 Conclusion

In this paper, we presented a monocular visual-inertial state estimation algorithm based on the modified SVO and error-state Kalman Filter. The proposed method can be used for state estimation of MAVs, the result of which can be used in the higher lever task such as controlling, mapping, planing and trajectory following. The performance of the proposed method was validate on the EuRoc dataset and the results was compared to the ground-truth and the state-of-the-art algorithm, VINS-Mono. However, two component of our framework are work independent at present, which means the estimate result do not feedback to the vision-pipeline for failure detecting or dirft suppression. Feature will investigate to tightly integrate the filter component in the vision-pipeline to get more accurate results.

## Acknowledgements

This work has been supported by National Natural Science Foundation of China (No. 61375086), Key Project (No. KZ201610005010) of S&T Plan of Beijing Municipal Commission of Education, Beijing Natural Science Foundation (4174083)

## References

[1] S. Weiss, M.W. Achtelik, S. Lynen, M.C. Achtelik, L. Kneip, M. Chli, R. Siegwart, Journal of Field Robotics **30**, 803 (2013)

[2] S. Shen, N. Michael, V. Kumar, *Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft MAVs*, in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2015), pp. 5303–5310

[3] M. Faessler, F. Fontana, C. Forster, E. Mueggler, M. Pizzoli, D. Scaramuzza, Journal of Field Robotics **33**, 431 (2016)

[4] Y. Lin, F. Gao, T. Qin, W. Gao, T. Liu, W. Wu, Z. Yang, S. Shen, Journal of Field Robotics pp. n/a–n/a (2017)

[5] L.R. García Carrillo, A.E. Dzul López, R. Lozano, C. Pégard, Journal of Intelligent & Robotic Systems **65**, 373 (2012)

[6] L. Heng, D. Honegger, G.H. Lee, L. Meier, P. Tanskanen, F. Fraundorfer, M. Pollefeys, Journal of Field Robotics **31**, 654 (2014)

[7] K. Schmid, P. Lutz, T. Tomić, E. Mair, H. Hirschmüller, Journal of Field Robotics **31**, 537 (2014)

[8] A. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, N. Roy, *Visual odometry and mapping for autonomous flight using an RGB-D camera*, in *Int. Symposium* (2011), pp. 1–16

[9] S. Shen, N. Michael, V. Kumar, *Autonomous indoor 3D exploration with a micro-aerial vehicle*, in *2012 IEEE International Conference on Robotics and Automation* (IEEE, 2012), pp. 9–15

[10] M.W. Achtelik, A. Bachrach, R. He, S. Prentice, N. Roy, *Stereo Vision and Laser Odometry for Autonomous Helicopters in GPS-denied Indoor Environments*, in *SPIE Conference on Unmanned Systems Technology* (2009), pp. 19–29

[11] M. Faessler, F. Fontana, C. Forster, D. Scaramuzza, *Automatic re-initialization and failure recovery for aggressive flight with a monocular vision-based quadrotor*, in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2015), June, pp. 1722–1729

[12] C. Forster, M. Pizzoli, D. Scaramuzza, Proceedings - IEEE International Conference on Robotics and Automation pp. 15–22 (2014)

[13] S. Lynen, M.W. Achtelik, S. Weiss, M. Chli, R. Siegwart, *A robust and modular multi-sensor fusion approach applied to MAV navigation*, in *IEEE International Conference on Intelligent Robots and Systems* (2013), pp. 3923–3929

[14] S. Weiss, R. Siegwart, *Real-time metric state estimation for modular vision-inertial systems*, in *Proceedings - IEEE International Conference on Robotics and Automation* (2011), Vol. 231855, pp. 4531–4537

[15] D. Scaramuzza, M.C. Achtelik, L. Doitsidis, F. Friedrich, E. Kosmatopoulos, A. Martinelli, M.W. Achtelik, M. Chli, S. Chatzichristofis, L. Kneip et al., IEEE Robotics & Automation Magazine **21**, 26 (2014)

[16] F.L. Markley, Journal of Guidance Control and Dynamics **26**, 311 (2003)

[17] Z. Yang, S. Shen, IEEE Transactions on Automation Science and Engineering **14**, 39 (2017)

[18] Z. Yang, S. Shen, *Monocular visual-inertial fusion with online initialization and camera-IMU calibration*, in *2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)* (IEEE, 2015), pp. 1–8

[19] R. Mur-Artal, J.D. Tardos, IEEE Robotics and Automation Letters **2**, 796 (2017)