

# Schmidt or Compressed filtering for Visual-Inertial SLAM?

Hongkyoon Byun and Jonghyuk Kim

University of Technology Sydney

{Hongkyoon.Byun@student, jonghyuk.kim@}.uts.edu.au

Fernando Vanegas and Felipe Gonzalez

Queensland University of Technology

{f.vanegasalvarez, felipe.gonzalez}@qut.edu.au

## Abstract

Visual-Inertial SLAM has been studied widely due to the advantage of its lightweight, cost-effectiveness, and rich information compared to other sensors. A multi-state constrained filter (MSCKF) and its Schmidt version have been developed to address the computational cost, which treats key-frames as static nuisance parameters, leading to sub-optimal performance. We propose a new Compressed-MSCKF which can achieve improved accuracy with moderate computational costs. By keeping the information gain with compressed form, it can be limited to  $\mathcal{O}(L)$  with  $L$  being the number of local key-frames. The performance of the proposed system has been evaluated using a MATLAB simulator.

## 1 Introduction

The autonomous navigation system for mobile robots has been focused on and studied for the past few decades. As a solution, the Simultaneous Localisation and Mapping (SLAM) system has been used with different kinds of sensors, such as cameras, Inertial Measurement Units (IMU), Light Detection and Ranging (Lidar), sonar, and so on. Multi-sensor fusion algorithms have been applied to improve the accuracy of this system [Chang *et al.*, 2018a].

Visual-Inertial Navigation System (VINS) has been investigated due to its advantages in size, cost and robustness. The visual sensor itself can detect and track hundreds of features, and it provides rich information through the images, which can bring the improvement of state estimation accuracy. When the camera system loses the tracking of visual features, the IMU can connect the gap with the higher frequency data [Qin *et al.*, 2019].

To resolve the problem of high computational cost, a balance between computational cost and accuracy of the

estimation is required for reliable real-time performance [Campos *et al.*, 2021b]. Multi-State Constraint Kalman Filter (MSCKF) [Mourikis and Roumeliotis, 2007] delivers the localisation information utilising multiple measurements of visual features without including the position of the 3D feature in the filter state vector, which only brings linear computational complexity in the number of features. In Schmidt-MSCKF [Geneva *et al.*, 2019a], key-frame poses are selectively added to the state vector for the loop closure constraints, but without updating the key-frame states to reduce the computational cost. Compressed-SLAM(CP-SLAM) [Kim *et al.*, 2017] has been demonstrated in which the active state is actively updated within the filter and inactive states are updated in a less frequent rate by accumulating the correlation information.

In this work, we apply the compressed filtering method to the MSCKF to solve the VINS problem, in which the key-frame poses are partitioned into local and global states and updated at a lower rate. In this way, the sub-optimality related to the Schmidt filtering can be adequately handled while constraining the computational complexity.

The outline of this paper is as follows. Section 2 provides the related work, and Section 3 reviews Schmidt-MSCKF. Section 4 provides the details of the Compressed-MSCKF, and the key-frame vector is updated at a much lower rate by compressing the active and key-frame state correlation information. Section 5 provides experimental results using the high-performance MATLAB simulator to verify the Compressed version, followed by a conclusion.

## 2 Related Work

A real-time SLAM system in a resource-limited embedded computing system is still challenging in achieving reliability and robustness.

Extensive research has been conducted to achieve reliable real-time VINS. ORB-SLAM3 [Campos *et al.*, 2021a] added the multi-map data association to their

previous work [Mur-Artal and Tardos, 2017], which contains ORB sparse front-end, graph optimization back-end, re-localization, and loop closure [Chang *et al.*, 2018b]. It can perform Maximum-a-Posteriori (MAP) estimation during the IMU initialisation. Also, the global Bundle Adjustment (BA) is only performed when the key-frame number is below the threshold to reduce the computational complexity. VINS-Mono [Qin *et al.*, 2018] contributed to the robustness of initialisation, re-localisation, and reusing the pose graph. Their back-end uses a sliding window and selectively marginalises the state from IMU and features from the sliding window to reduce the computational cost. A lightweight motion-only visual-inertial optimisation also has been implemented for the low computational power devices.

In terms of computing resources, EKF based SLAM shows better efficiency in small-end applications compared to the above optimisation-based methods [Chang *et al.*, 2018b]. [Mourikis and Roumeliotis, 2008] applied the MSCKF [Mourikis and Roumeliotis, 2007] as a first layer taking the computational advantage and BA for a loop closure detection in the second layer. It could relieve the linearisation errors and reduce the drift over the long trajectory but still cause the processing overhead during the BA. Maplab [Schneider *et al.*, 2018] utilises ROVIOLI [Bloesch *et al.*, 2015] and enhances the online performance, including the tools such as loop-closure, multi-session map merging and pose-graph relaxation. However, this system only yields the accurate pose estimation within the prior map. [Geneva *et al.*, 2019b] implemented the idea from Schmidt Kalman filter [Schmidt, 1966] which selectively includes the feature information into the state vector and considers them as a nuisance parameter. There is no updating process for the key-frame state, but cross-correlation with the active state is still maintained. They further applied the method to the MSCKF as [Geneva *et al.*, 2019a]. Even though it enables linear computational complexity, the estimator performance is still sub-optimal due to the static nature of the key-frames.

### 3 Schmidt-MSCKF

In the Schmidt-MSCKF [Geneva *et al.*, 2019a], the standard MSCKF [Mourikis and Roumeliotis, 2007] is implemented which has the state vector containing the IMU state  $x_I$  and  $N$  cloned past camera poses  $x_C$  as a sliding window,

$$x_k = [ x_I^\top \quad x_C^\top ]^\top \quad (1)$$

$$x_I = [ \begin{matrix} I_k \bar{q}^\top & b_{\omega_k}^\top & G v_{I_k}^\top & b_{a_k}^\top & G p_{I_k}^\top \end{matrix} ]^\top \quad (2)$$

$$x_C = [ \begin{matrix} C_{k-1} \bar{q}^\top & G p_{C_{k-1}}^\top & \cdots & C_{k-N} \bar{q}^\top & G p_{C_{k-N}}^\top \end{matrix} ]^\top, \quad (3)$$

where  ${}^I_G \bar{q}$  is the unit quaternion that represent the rotation from frame  $\{G\}$  to frame  $\{I\}$ ,  $b_w$  and  $b_a$  are the biases of gyroscope and accelerometer, and  ${}^G v_I$  and  ${}^G p_I$  are respectively IMU velocity and position with respect to  $G$ . Furthermore, Schmidt-MSCKF includes the keyframe poses into the state for loop closure as follows,

$$x_k = [ x_I^\top \quad x_C^\top \quad x_{S_1}^\top \quad \cdots \quad x_{S_n}^\top ]^\top = [ x_A^\top \quad x_S^\top ]^\top, \quad (4)$$

where the  $x_{S_i} = [ \begin{matrix} {}^G \bar{q}^\top & G p_{C_i}^\top \end{matrix} ]^\top$  is a keyframe pose to solve the problem of the drifts accumulating over time.

#### 3.1 Measurement Model

The measurement model of Schmidt-MSCKF is also derived from the standard MSCKF. In the framework of MSCKF, the camera observations are processed once the features have enough parallax in the sliding window. The same 3D pose point measurements are used to define a constraint equation, relating all the camera poses which are added when there is measurement. Therefore, the stacked nonlinear camera measurement model can be described as [Geneva *et al.*, 2019a]:

$$z_f = h(x_k, {}^G p_f) + n_f, \quad (5)$$

where  $n_f$  is the white Gaussian noise with covariance  $R_f$ , and  ${}^G p_f$  is the 3D position of the feature. The residual model can be obtained using the above measurement model:

$$r_f = z_f - h(\hat{x}_{k|k-1}, {}^G \hat{p}_f) \quad (6)$$

$$r_f = H_x \tilde{x}_{k|k-1} + H_f {}^G \tilde{p}_f + n_f, \quad (7)$$

where  $H_x$  and  $H_f$  are the Jacobians with respect to the state and the feature position, and  $\tilde{x}$  and  ${}^G \tilde{p}$  are the errors of state and feature position. However, standard EKF update cannot be conducted due to the correlation between  $\tilde{x}$  and  ${}^G \tilde{p}$ . To resolve this problem,  $r_f$  is projected to the left nullspace of  $H_f$ , which can transform into the residual model independent from the position of the feature as:

$$N^\top r_f = N^\top H_x \tilde{x}_{k|k-1} + N^\top H_f {}^G \tilde{p}_f + N^\top n_f \quad (8)$$

$$r'_f = H'_x \tilde{x}_{k|k-1} + n'_f, \quad (9)$$

where  $n'_f$  is white Gaussian noise with covariance  $R'_f = N^\top R_f N$ , and it can be updated as general EKF.

#### 3.2 Update

With the derived measurement model, the residual of Schmidt-MSCKF can be described as:

$$r'_f \simeq H_{A_k} \tilde{x}_{A_k|k-1} + H_{S_k} \tilde{x}_{S_k|k-1} + n'_f, \quad (10)$$

where  $H'_x = [H_{A_k} \ H_{S_k}]$ . However, due to the keyframes in the state vector,  $x_S$ , which can grow over time, update

process cannot be performed in real time. To solve this problem, Schmidt Kalman Filter [Schmidt, 1966] is implemented to avoid the high computational complexity. The intention of SKF update is to treat the keyframe state as static, which makes  $K_{S_k}$  as a zero. Therefore, using the general EKF update process, the estimation of state can be described as follows:

$$\hat{x}_{A_{k|k}} = \hat{x}_{A_{k|k-1}} + K_{A_k} \tilde{z}'_k \quad (11)$$

$$\hat{x}_{S_{k|k}} = \hat{x}_{S_{k|k-1}}, \quad (12)$$

where the covariance matrix can be updated keeping the cross-correlation term for consistency as

$$P_{k|k} = P_{k|k-1} - \begin{bmatrix} K_{A_k} S_k K_{A_k}^\top & K_{A_k} H_k' & \begin{bmatrix} P_{AS_{k|k-1}} \\ P_{SS_{k|k-1}} \end{bmatrix} \\ \begin{bmatrix} P_{AS_{k|k-1}} \\ P_{SS_{k|k-1}} \end{bmatrix}^\top & H_k'^\top K_{A_k}^\top & 0 \end{bmatrix}. \quad (13)$$

As a result, the computational complexity can be reduced from  $O(n^2)$  to  $O(n)$ .

## 4 Compressed-MSCKF

As shown in the previous section, Schmidt-MSCKF implemented the nullspace technique, which enables to define the constraints without including the feature position to reduce the computational complexity. In addition, they selectively store key-frames for the loop closure using the Schmidt filtering techniques. It efficiently controls the unbounded localisation error and the computational cost by treating the key-frame as static, leading to the linear growth of computational complexity. However, even though it reduces the computational cost, the loss of information cannot be ignored while treating the key-frame state as a 'nuisance' through the trajectory. Finding the right balance between computing requirements and accuracy is a vital part of the real-time SLAM system. Therefore, we present the Compressed-MSCKF including loop closure in this section.

In a framework of standard Compressed SLAM [Kim *et al.*, 2017], the state vector is partitioned into a local and global map state. The main benefit of this system is the accumulation of local information gain while operating within the local boundary. This compressed information only propagates to the global state when the local boundary changes to the new boundary, and it allows to update the global map at a much lower rate. In Compressed SLAM, the key-frame states are further partitioned into:

$$x_S = \begin{bmatrix} x_{S_L}^\top & x_{S_G}^\top \end{bmatrix}^\top, \quad (14)$$

where  $x_{S_L}$  and  $x_{S_G}$  are the keyframe states in the local and global boundary, respectively. Therefore, we can

describe the state vector and the corresponding covariance of the Compressed-MSCKF based on [Geneva *et al.*, 2019a] as follow:

$$x_k = \begin{bmatrix} x_L \\ x_G \end{bmatrix} = \begin{bmatrix} x_A \\ x_{S_L} \\ x_{S_G} \end{bmatrix} \quad (15)$$

$$P = \begin{bmatrix} P_{LL} & P_{LG} \\ P_{GL} & P_{GG} \end{bmatrix} = \begin{bmatrix} P_{AA} & P_{AS_L} & P_{LG} \\ P_{S_L A} & P_{S_L S_L} & P_{GG} \\ P_{GL} & P_{GG} & P_{GG} \end{bmatrix}, \quad (16)$$

where  $x_A = \begin{bmatrix} x_I^\top & x_C^\top \end{bmatrix}^\top$ . In this way, since the keyframe need to be kept updated, compression method can be implemented to effectively reduce the computational cost. It can operate the system in  $O(L^2)$  of computational complexity with  $L$  being the local map size, which is much less than the total map size.

### 4.1 Propagation

Using a standard EKF, the estimation of IMU state can be propagated with the incoming IMU measurement, linear acceleration ( $a_m$ ) and angular velocity ( $\omega_m$ ), based on the following IMU kinematics [Mourikis and Roumeliotis, 2007]:

$$x_k = f(x_{k-1}, a_{m_{k-1}} - n_{a_{k-1}}, \omega_{m_{k-1}} - n_{\omega_{k-1}}), \quad (17)$$

where  $n_a$  and  $n_\omega$  are the zero mean white Gaussian noise of the IMU measurements. The propagation of the covariance matrix also propagates as:

$$P_{k|k-1} = \begin{bmatrix} J_{k-1} P_{AA} J_{k-1}^\top & J_{k-1} P_{AS} \\ (J_{k-1} P_{AS})^\top & P_{SS} \end{bmatrix} + \begin{bmatrix} Q_{k-1} & 0 \\ 0 & 0 \end{bmatrix}, \quad (18)$$

where  $J$  and  $Q$  are the system Jacobian and discrete noise covariance matrices for the active state. It can be seen that the correlation between the active and key-frame states can be expressed in a compressed form:

$$P_{AS}(k) = \left( \prod_{i=1}^k J_{k-1} \right) P_{AS}(0), \quad (19)$$

which can be compressed until the local boundary changes.

### 4.2 Update

In our Compressed-MSCKF framework, the observations are first used to update the local state, and the correlation with global state is accumulated. Observation matrix  $H$  is sparse and has values only for the local state, it can be expressed as  $H'_x = \begin{bmatrix} H_{L_k} & 0_{G_k} \end{bmatrix}$  and it can derive the residual measurement as:

$$r'_f \simeq H_{L_k} \tilde{x}_{L_{k|k-1}} + n'_f \quad (20)$$

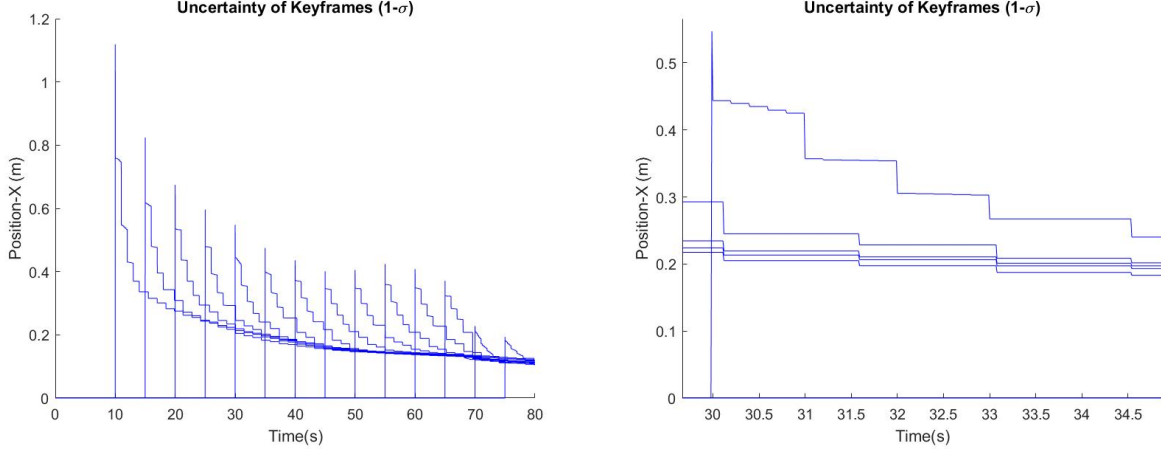


Figure 1: The evolution of uncertainty of keyframes and the enhanced view of keyframe number 5 showing the compressed updates during the simulation.

Using this model, the updating process of state estimate can be derived as follows:

$$\hat{x}_{L_{k|k}} = \hat{x}_{L_{k|k-1}} + K_{L_k} \tilde{z}'_k \quad (21)$$

$$\hat{x}_{G_{k|k}} = \hat{x}_{G_{k|k-1}} + K_{G_k} \tilde{z}'_k \quad (22)$$

The Kalman gain can be computed as

$$K_k = PH^T S^{-1} = \begin{bmatrix} P_{LL_{k|k-1}} H_{L_k}^T S_k^{-1} \\ P_{GL_{k|k-1}} H_{L_k}^T S_k^{-1} \end{bmatrix} = \begin{bmatrix} K_{L_k} \\ K_{G_k} \end{bmatrix},$$

where  $S_k = H'_k P_{k|k-1} H_k^T + R' = H_{L_k} P_{LL_{k|k-1}} H_{L_k}^T + R'$ , and therefore the form of updated covariance matrix is represented as:

$$\begin{aligned} P_{k|k} &= P_{k|k-1} - K_k S_k K_k^T \\ &= P_{k|k-1} - \begin{bmatrix} P_{LL_{k|k-1}} H_{L_k}^T S_k^{-1} \\ P_{GL_{k|k-1}} H_{L_k}^T S_k^{-1} \end{bmatrix} S \begin{bmatrix} P_{LL_{k|k-1}} H_{L_k}^T S_k^{-1} \\ P_{GL_{k|k-1}} H_{L_k}^T S_k^{-1} \end{bmatrix}^T \\ &= P_{k|k-1} - \begin{bmatrix} P_{LL_{k|k-1}} (H_{L_k}^T S_k^{-1} H_{L_k}) P_{LL_{k|k-1}} & \underbrace{(P_{LL_{k|k-1}} (H_{L_k}^T S_k^{-1} H_{L_k}))}_{\phi} P_{LG_{k|k-1}} \\ \left( (P_{LL_{k|k-1}} (H_{L_k}^T S_k^{-1} H_{L_k}) P_{LG_{k|k-1}})^T \right) & P_{GL_{k|k-1}} \underbrace{(H_{L_k}^T S_k^{-1} H_{L_k})}_{\Psi} P_{LG_{k|k-1}} \end{bmatrix}. \end{aligned} \quad (23)$$

During this process, the terms of  $K_{G_k} S_k K_{G_k}^T$  shows the largest computational cost in standard system and it can be computed in compressed form to effectively handle the computational complexity. Correlation and global state terms can be shortened using  $\phi, \Psi$ , and  $\Phi$  as follows:

$$P_{LG_{k|k}} = P_{LG_{k|k-1}} - \phi P_{LG_{k|k-1}} = \Phi P_{LG_{k|k-1}} \quad (24)$$

$$P_{GG_{k|k}} = P_{GG_{k|k-1}} - (P_{GL_{k|k-1}} \Psi P_{LG_{k|k-1}}) \quad (25)$$

$$\hat{x}_{G_{k|k}} = \hat{x}_{G_{k|k-1}} + (P_{GL_{k|k-1}} H_{L_k}^T S_k^{-1} \tilde{z}') \quad (26)$$

As a result, accumulated form of the global state, correlation term  $P_{LG}$ , and global state vector can be expressed as below:

$$P_{LG}(k) = \left( \prod \Phi \right) P_{LG}(0) = \Phi(k, 0) P_{LG}(0) \quad (27)$$

$$P_{GG}(k) = P_{GG}(0) - P_{GL}(0) \left( \sum \Phi(k, 0)^T \Psi \Phi(k, 0) \right) P_{LG}(0) \quad (28)$$

$$\hat{x}_G(k) = \hat{x}_G(0) + P_{GL}(0) \left( \sum \Phi(k, 0)^T H_L^T S^{-1} \tilde{z}' \right) \quad (29)$$

Using this compressed correlation term, the global map and covariance can be recovered at a much lower rate whenever the local map boundary changes.

## 5 Preliminary Results

A high-fidelity MATLAB simulator is used to verify the method. The simulator, called Compressed-Pseudo-SLAM, is for all-source navigation and utilises the landmark-based visual-inertial SLAM together as well as GPS pseudo-range and pseudo-range rate information. The sensor data of IMU (at 100Hz), vision (at 30Hz) and pseudo-ranges (at 1Hz) are generated following the simulated trajectory (a racehorse track in this work) using realistic sensor models.

To demonstrate the compressed-MSCKF filter, the simulator is modified to augment the key-frames of the vehicle poses into the state vector. The frequency of the augmentation is dependent on the camera's field-of-view and speed of the vehicle and a 5-second interval is used in this work which evenly covers the trajectory. Figure 1 (left) shows the uncertainty evolution of the keyframes showing total 14 keyframes registered. The compressed update is applied whenever the vehicle reaches the boundary of the local map, in which case the centre of the local map is re-centred at the current vehicle location. In the simulation, the local map is re-centred

at approximately 1.5–2 seconds. Figure 1 (right) shows the enhanced view of the key-frame number 5, showing the effects of sensor updates as well as the compressed updates. At around 30 seconds, the fifth key-frame is added to the state vector and continuously updated from the vision information. The correction at 31 seconds is due to the 1Hz GPS data, while the correction at around 34.5 seconds is due to the global update. After 2 global updates, the key-frame becomes out of the local region and becomes an inactive state, corrected only by the global compressed updates. In the Schmidt-MSCKF, the key-frames are initialised using the corresponding vehicle poses but without the covariance information. A more in-depth analysis is planned to understand the optimal trade-off between computational complexity and accuracy.

## 6 Conclusions

In this work, we applied the compressed filtering framework to MSCKF-based Visual-Inertial SLAM. The existing Schmidt-filter based approach can reduce the computational cost by treating the pose key-frames as static variables and thus sacrificing the accuracy. We showed that we can still retain the pose key-frames in the SLAM state while limiting the computational complexity to  $\mathcal{O}(L)$  with  $L$  being the number of local key-frames. The future is on the trade-off analysis between the computational cost and accuracy and the implementation of the method for a real dataset collected from drones.

## References

- [Bloesch *et al.*, 2015] Michael Bloesch, Sammy Omari, Marco Hutter, and Roland Siegwart. Robust visual inertial odometry using a direct ekf-based approach. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 298–304, 2015.
- [Campos *et al.*, 2021a] Carlos Campos, Richard Elvira, Juan J. Gomez Rodriguez, Jose M. M. Montiel, and Juan D. Tardos. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam. *IEEE transactions on robotics*, pages 1–17, 2021.
- [Campos *et al.*, 2021b] Carlos Campos, Richard Elvira, Juan J. Gómez Rodríguez, José M. M. Montiel, and Juan D. Tardós. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam. *IEEE Transactions on Robotics*, pages 1–17, 2021.
- [Chang *et al.*, 2018a] Chen Chang, Zhu Hua, Li Meng-gang, and You Shaoze. A review of visual-inertial simultaneous localization and mapping from filtering-based and optimization-based perspectives. *Robotics (Basel)*, 7(3):45, 2018.
- [Chang *et al.*, 2018b] Chen Chang, Zhu Hua, Li Meng-gang, and You Shaoze. A review of visual-inertial simultaneous localization and mapping from filtering-based and optimization-based perspectives. *Robotics (Basel)*, 7(3):45, 2018.
- [Geneva *et al.*, 2019a] Patrick Geneva, Kevin Ekenhoff, and Guoquan Huang. A linear-complexity ekf for visual-inertial navigation with loop closures. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3535–3541, 2019.
- [Geneva *et al.*, 2019b] Patrick Geneva, James Maley, and Guoquan Huang. An efficient schmidt-ekf for 3d visual-inertial slam. 2019.
- [Kim *et al.*, 2017] Jonghyuk Kim, Jiantong Cheng, Jose Guivant, and Juan Nieto. Compressed fusion of gnss and inertial navigation with simultaneous localisation and mapping. *IEEE Aerospace and Electronic Systems Magazine*, 32(8):22–36, 2017.
- [Mourikis and Roumeliotis, 2007] Anastasios I. Mourikis and Stergios I. Roumeliotis. A multi-state constraint kalman filter for vision-aided inertial navigation. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3565–3572, 2007.
- [Mourikis and Roumeliotis, 2008] Anastasios I. Mourikis and Stergios I. Roumeliotis. A dual-layer estimator architecture for long-term localisation. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8, 2008.
- [Mur-Artal and Tardos, 2017] Raul Mur-Artal and Juan D. Tardos. Visual-inertial monocular slam with map reuse. *IEEE robotics and automation letters*, 2(2):796–803, 2017.
- [Qin *et al.*, 2018] Tong Qin, Peiliang Li, and Shaojie Shen. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018.
- [Qin *et al.*, 2019] Tong Qin, Jie Pan, Shaozu Cao, and Shaojie Shen. A general optimisation-based framework for local odometry estimation with multiple sensors, 2019.
- [Schmidt, 1966] Stanley F. Schmidt. *Application of State-Space Methods to Navigation Problems*, volume 3, pages 293–340. Elsevier, 1966.
- [Schneider *et al.*, 2018] Thomas Schneider, Marcin Dymczyk, Marius Fehr, Kevin Egger, Simon Lynen, Igor Gilitschenski, and Roland Siegwart. Maplab: An open framework for research in visual-inertial mapping and localisation. *IEEE Robotics and Automation Letters*, 3(3):1418–1425, 2018.