Simultaneous state initialization and gyroscope bias calibration in visual inertial aided navigation

Kaiser, Jacques ; Martinelli, Agostino ; Fontana, Flavio ; Scaramuzza, Davide

Abstract: State of the art approaches for visual-inertial sensor fusion use filter-based or optimization-based algorithms. Due to the nonlinearity of the system, a poor initialization can have a dramatic impact on the performance of these estimation methods. Recently, a closed-form solution providing such an initialization was derived in [1]. That solution determines the velocity (angular and linear) of a monocular camera in metric units by only using inertial measurements and image features acquired in a short time interval. In this letter, we study the impact of noisy sensors on the performance of this closed-form solution. We show that the gyroscope bias, not accounted for in [1], significantly affects the performance of the method. Therefore, we introduce a new method to automatically estimate this bias. Compared to the original method, the new approach now models the gyroscope bias and is robust to it. The performance of the proposed approach is successfully demonstrated on real data from a quadrotor MAV.

DOI: https://doi.org/10.1109/LRA.2016.2521413

Simultaneous State Initialization and Gyroscope Bias Calibration in Visual Inertial aided Navigation

Jacques Kaiser[1], Agostino Martinelli[1], Flavio Fontana[2] and Davide Scaramuzza[2]

***Abstract*—State of the art approaches for visual-inertial sensor fusion use filter-based or optimization-based algorithms. Due to the nonlinearity of the system, a poor initialization can have a dramatic impact on the performance of these estimation methods. Recently, a closed-form solution providing such an initialization was derived in [1]. That solution determines the velocity (angular and linear) of a monocular camera in metric units by only using inertial measurements and image features acquired in a short time interval. In this paper, we study the impact of noisy sensors on the performance of this closed-form solution. We show that the gyroscope bias, not accounted for in [1], significantly affects the performance of the method. Therefore, we introduce a new method to automatically estimate this bias. Compared to the original method, the new approach now models the gyroscope bias and is robust to it. The performance of the proposed approach is successfully demonstrated on real data from a quadrotor MAV.**

***Index Terms*—Sensor Fusion, Localization, Visual-Based Navigation**

I. INTRODUCTION

AUTONOMOUS mobile robots navigating in unknown environments have an intrinsic need to perform localization and mapping using only on-board sensors. Concerning Micro Aerial Vehicles (MAV), a critical issue is to limit the number of on-board sensors to reduce weight and power consumption. Therefore, a common setup is to combine a monocular camera with an inertial measurements unit (IMU). On top of being cheap, these sensors have very interesting complementarities. Additionally, they can operate in indoor environments, where Global Positioning System (GPS) signals are shadowed. An open question is how to optimally fuse the information provided by these sensors.

Currently, most sensor-fusion algorithms are either filter-based or iterative. That is, given a current state and measurements, they return an updated state. While working well in practice, these algorithms need to be provided with an initial state. The initialization of these methods is critical. Due to nonlinearities of the system, a poor initialization can result into converging towards local minima and providing faulty states with high confidence.

Manuscript received: August, 31, 2015; Revised September, 29, 2015; Accepted January, 8, 2016.

This paper was recommended for publication by Editor Antonio Bicchi upon evaluation of the Associate Editor and Reviewers' comments. *This work was supported by the French National Research Agency ANR through the project VIMAD.

[1] INRIA Rhone Alpes, Grenoble, France. jacko.kaiser@gmail.com, agostino.martinelli@ieee.org

[2] Robotics and Perception Group, University of Zurich, Switzerland. ffontana,sdavideg@ifi.uzh.ch

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

In this paper, we demonstrate the efficiency of a recent closed-form solution introduced in [1, 2], which fuses visual and inertial data to obtain the structure of the environment at the global scale along with the attitude and the speed of the robot. By nature, a closed-form solution is deterministic and, thus, does not require any initialization.

The method introduced in [1, 2] was only described in theory and demonstrated with simulations on generic Gaussian motions, not plausible for an MAV. In this paper, we perform simulations with plausible MAV motions and synthetic noisy sensor data. Our simulations are therefore closer to the real dynamics of an MAV. This allows us to identify limitations of the method and bring modifications to overcome them. Specifically, we investigate the impact of biased inertial measurements. Although the case of biased accelerometer was originally studied in [1], here we show that a large bias on the accelerometer does not significantly worsen the performance. One major limitation of [1] is the impact of biased gyroscope measurements. In other words, the performance becomes very poor in presence of a bias on the gyroscope and, in practice, the overall method can only be successfully used with a very precise - and expensive - gyroscope. Here, we introduce a simple method that automatically estimates this bias. By adding this new method for the bias estimation to the original method [1], we obtain results that are equivalent to the ones in absence of bias. This method is suitable for dynamic take off and on-the-fly re-initialisation since it does not require a calibration step with the MAV sitting stationary. Compared to [1], the new method is now robust to the gyroscope bias and automatically calibrates the gyroscope.

II. RELATED WORK

The problem of fusing visual and inertial data has been extensively investigated in the past. However, most of the proposed methods require a state initialization. Because of the system nonlinearities, lack of precise initialization can irreparably damage the entire estimation process. In literature, this initialization is often guessed or assumed to be known [3–6]. Recently, this sensor fusion problem has been successfully addressed by enforcing observability constraints [7, 8] and by using optimization-based approaches [9–15]. These optimization methods outperform filter-based algorithms in terms of accuracy due to their capability of relinearizing past states. On the other hand, the optimization process can be affected by the presence of local minima. We are therefore interested in a deterministic solution that analytically expresses the state in terms of the measurements provided by the sensors during a short time-interval.

In computer vision, several deterministic solutions have been introduced. These techniques, known as *Structure from Motion*, can recover the relative rotation and translation up to an unknown scale factor between two camera poses [16]. Such methods are currently used in state-of-the-art visual navigation methods for MAVs to initialize maps [6, 17, 18]. However, the knowledge of the absolute scale, and, at least, of the absolute roll and pitch angles, is essential for many applications ranging from autonomous navigation in GPS-denied environments to 3D reconstruction and augmented reality. For these applications, it is crucial to take the inertial measurements into consideration to compute these values deterministically.

A procedure to quickly re-initialize an MAV after a failure was presented in [19]. However, this method requires an altimeter to initialize the scale.

Recently, a closed-form solution has been introduced in [2]. From integrating inertial and visual measurements over a short time-interval, this solution provides the absolute scale, roll and pitch angles, initial velocity, and distance to 3D features. Specifically, all the physical quantities are obtained by simply inverting a linear system. The solution of the linear system can be refined with a quadratic equation assuming the knowledge of the gravity magnitude. This closed-form was improved in [20] to work with unknown camera-IMU calibration; however, since in this case the problem cannot be solved by simply inverting a linear system, a method to determine the six parameters that characterize the camera-IMU transformation was proposed. As a result, this method is independent of external camera-IMU calibration, hence, suitable for power-on-and-go systems.

A more intuitive expression of this closed-form solution was derived in [1]. While being mathematically sound, this closed-form solution is not robust to noisy sensor data. For this reason, to the best of our knowledge, it has never been used in an actual application. In this paper, we perform an analysis to find out its limitations. We start by reminding the reader the basic equations that characterize this solution (section III). In section IV, we show that this solution is resilient to the accelerometer bias but strongly affected by the gyroscope bias. We then introduce a simple method that automatically estimates the gyroscope bias (section V). By adding this new method for the bias estimation to the original method, we obtain results that are equivalent to the ones obtained in absence of bias. Compared to the original method, the new method is now robust to the gyroscope bias and also calibrates the gyroscope. In section VI, we validate our new method against real world data from a flying quadrotor MAV to prove its robustness against noisy sensors during actual navigation. Finally, we provide the conclusions in section VII.

III. CLOSED-FORM SOLUTION

In this section, we provide the basic equations that characterize the closed-form solution proposed in [1][1]. Let us refer to a short interval of time (e.g., of the order of 3 seconds). We assume that during this interval of time the camera observes simultaneously N point-features and we denote by $t_1,\ t_2, \cdots, t_{n_i}$ the times of this interval at which the camera provides an image of these points. Without loss of generality, we can assume that $t_1 = 0$. The following equation holds (see [1] for its derivation):

$$S_j = \lambda_1^i \mu_1^i - V t_j - G\frac{t_j^2}{2} - \lambda_j^i \mu_j^i \tag{1}$$

with:

- μ_j^i the normalized bearing of point feature i at time t_j in the local frame at time t_1;
- λ_j^i the distance to the point feature i at time t_j;
- V the velocity in the local frame at time t_1;
- G the gravity in the local frame at time t_1;
- S_j the integration in the interval $[t_1,\ t_j]$ of the rotated linear acceleration data (i.e., the integration of the inertial measurements).

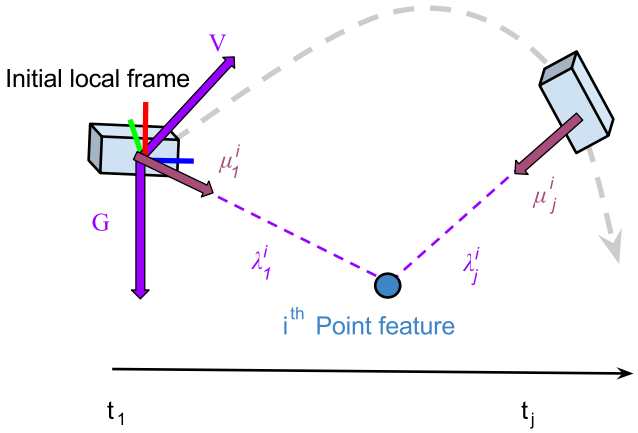


Fig. 1: Visual representation of Equation (1). The unknowns of the equation are colored in purple.

A visual representation of Equation (1) is provided in Fig. 1. The local frame refers to a frame of reference common to the IMU and the camera. In a real application, we would work in the IMU frame and have some additional constant terms accounting for the camera-IMU transformation. We do not express these constant calibration terms explicitly here for clarity reasons.

The unknowns of Equation (1) are the scalars λ_j^i and the vectors V and G. Note that the knowledge of G is equivalent to the knowledge of the roll and pitch angles. The vectors μ_j^i are fully determined by visual and gyroscope measurements [2], and the vectors S_j are determined by accelerometer and gyroscope measurements.

Equation (1) provides three scalar equations for each point feature $i = 1, ..., N$ and each frame starting from the second one $j = 2, ...n_i$. We therefore have a linear system consisting of $3(n_i - 1)N$ equations in $6 + Nn_i$ unknowns. Indeed, note that, when the first frame is taken at $t_1 = 0$, Equation (1) is always satisfied; thus does not provide information. We can

[1]Note that in this paper we do not provide a new derivation of this solution for which the reader is addressed to [1], section 3.

[2]The gyroscope measurements in the interval $[t_1,\ t_j]$ are needed to express the bearing at time t_j in the frame at time t_1

write our system using matrix formulation. Solving the system is equivalent to inverting a matrix of $3(n_i - 1)N$ rows and $6 + Nn_i$ columns.

In [1], the author proceeded to one more step before expressing the underlying linear system. For a given frame j, the equation of the first point feature $i = 1$ is subtracted from all other point feature equations $1 < i \leq N$ (Equation (7) in [1]). This additional step, very useful to detect system singularities, has the effect to corrupt all measurements with the first measurement, hence worsening the performance of the closed-form solution. Therefore, in this paper we discard this additional step.

The linear system in Equation (1) can be written in the following compact form:

$$\Xi X = S. \tag{2}$$

Matrix Ξ and vector S are fully determined by the measurements, while X is the unknown vector. We have:

$$S \equiv [S_2^T, ..., S_2^T, S_3^T, ..., S_3^T, ..., S_{n_i}^T, ..., S_{n_i}^T]^T$$
$$X \equiv [G^T, V^T, \lambda_1^1, ..., \lambda_1^N, ..., \lambda_{n_i}^1, ..., \lambda_{n_i}^N]^T$$

$$\Xi \equiv \left[\begin{array}{cc|ccc|ccc|ccc} T_2 & S_2 & \mu_1^1 & 0_3 & 0_3 & -\mu_2^1 & 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ T_2 & S_2 & 0_3 & \mu_1^2 & 0_3 & 0_3 & -\mu_2^2 & 0_3 & 0_3 & 0_3 & 0_3 \\ ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... \\ T_2 & S_2 & 0_3 & 0_3 & \mu_1^N & 0_3 & 0_3 & -\mu_2^N & 0_3 & 0_3 & 0_3 \\ ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... \\ ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... \\ T_{n_i} & S_{n_i} & \mu_1^1 & 0_3 & 0_3 & 0_3 & 0_3 & 0_3 & -\mu_{n_i}^1 & 0_3 & 0_3 \\ T_{n_i} & S_{n_i} & 0_3 & \mu_1^2 & 0_3 & 0_3 & 0_3 & 0_3 & 0_3 & -\mu_{n_i}^2 & 0_3 \\ ... & ... & ... & ... & ... & ... & ... & ... & ... & ... & ... \\ T_{n_i} & S_{n_i} & 0_3 & 0_3 & \mu_1^N & 0_3 & 0_3 & 0_3 & 0_3 & 0_3 & -\mu_{n_i}^N \end{array}\right],$$

where $T_j \equiv -\frac{t_j^2}{2} I_3$, $S_j \equiv -t_j I_3$ and I_3 is the identity 3×3 matrix, 0_3 is the 3×1 zero matrix. Note that matrix Ξ and vector S are slightly different from the ones proposed in [1]. This is due to the additional step that, as we explained in the previous paragraph, we discarded for numerical stability reasons (see [1] section 3 for further details).

The sensor information is completely contained in the above linear system. Additionally, in [1], the author added a quadratic equation assuming the gravitational acceleration is a priori known. Let us denote the gravitational magnitude by g. We have the extra constraint $|G| = g$ that we can express in matrix formulation:

$$|\Pi X|^2 = g^2, \tag{3}$$

with $\Pi \equiv [I_3, 0_3, ..., 0_3]$. We can therefore recover the initial velocity, the roll and pitch angles, and the distances to the point features by finding the vector X satisfying (2) and (3).

In the next sections, we will evaluate the performance of this method on simulated noisy sensor data. This will allow us to identify its weaknesses and bring modifications to overcome them.

IV. LIMITATIONS OF [1]

The goal of this section is to find out the limitations of the solution proposed in [1] when it is adopted in a real scenario. In particular, special attention will be devoted to the case of an MAV equipped with low-cost camera and IMU sensors. For this reason, we perform simulations that significantly differ from the ones performed in [1] (section 5.2). Specifically, they differ because of the following two reasons:

- The simulated motion is the one of an MAV;
- The values of the biases are significantly larger than the ones in [1].

This will allow us to evaluate the impact of the bias on the performance.

A. Simulation setup

We simulate an MAV as a point particle executing a circular trajectory of about 1m radius. We measure our error on the absolute scale by computing the mean error over all estimated distances to point features λ_j^i. We define the relative error as the euclidean distance between the estimation and the ground truth, normalized by the ground truth.

Synthetic gyroscope and accelerometer data are affected by a statistical error of 0.5 deg/s and 0.5 cm/s^2, respectively and they are also corrupted by a constant bias.

We set 7 simulated 3D point-features about 3m away from the MAV, which flies at a speed of around $2\ m\ s^{-1}$. We found that setting the frame rate of the simulated camera at 10Hz provides a sufficient pixel disparity with the following setup. In practice, increasing the frame rate above 30Hz decreases the pixel disparity and introduces numerical instability for this setup. The theoretical cases in which our system admits singularities are provided in [1, 2]. Reducing the number of considered frames also reduces the size of the matrices and, thus, speeds up the computations. As an example, over a time interval of 3 seconds, we obtain 31 distinct frames. When observing 7 features, solving the closed-form solution is equivalent to inverting a linear system of $3 \times 30 \times 7 = 630$ equations and $6 + 7 \times 31 = 223$ unknowns (see section III).

The method we use to solve the overconstrained linear system $\Xi X = S$ is a Singular Value Decomposition (SVD) since it yields numerically robust solutions.

In the next section, we will present the results obtained with the original closed-form solution on the simulated data mentioned, with different sensor bias settings. Our goal is to identify its performance limitations and introduce modifications to overcome them.

B. Performance without bias

The original closed-form solution described in Equation (2) will be used as a basis for our work. Moreover, we can also use the knowledge of the gravity magnitude to refine our results (Equation (3)). In this case, we are minimizing a linear objective function with a quadratic constraint. In Fig. 2, we display the performance of the original Closed-Form (CF) solution in estimating speed, gravity in the local frame, and distances to the features with and without this additional constraint.

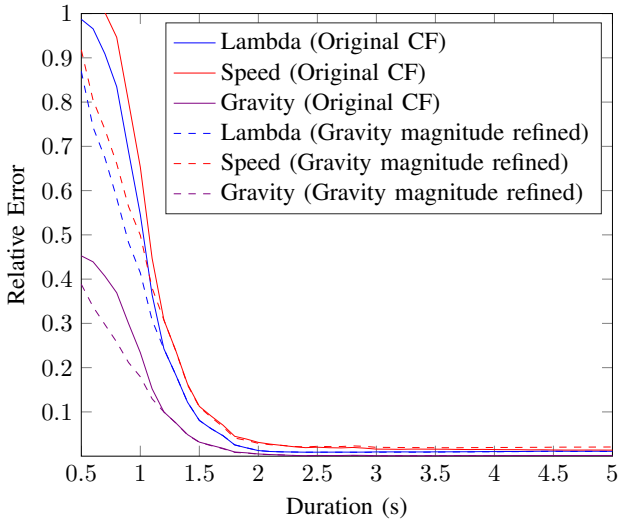
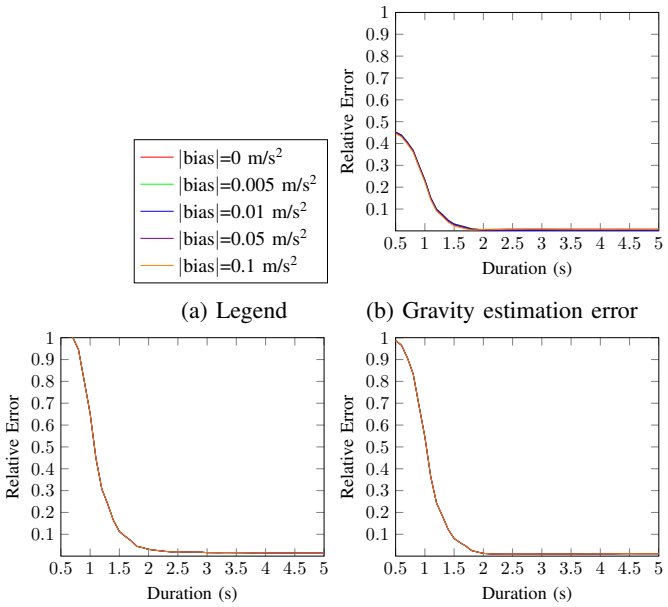


Fig. 2: Original closed-form solution estimations with and without using the knowledge of the gravity (3). We are observing 7 features over a variable duration of integration.



(a) Legend (b) Gravity estimation error

(c) Velocity estimation error (d) Lambda estimation error

Fig. 3: Impact of the accelerometer bias on the performance of the closed-form solution. We are observing 7 features over a variable duration of integration.

Note how the evaluations get better as we increase the integration time. Indeed, our equations come from an extended triangulation [2]. Therefore, it requires a significant difference in the measurements over time to robustly estimate the state.

Without sensor bias, the original closed-form robustly estimates all the properties (below 0.1% error) after 2 seconds of integration. Note that a robust estimation of the gravity requires a shorter duration of integration than the speed and the distance to the features. In general, we found that the gravity is well estimated with the original closed-form solution due to its strong weight in the equations (see section IV-C). Therefore, constraining its magnitude does not improve the performance much. In the following sections, we remove this constraint.

C. Impact of accelerometer bias on the performance

In order to visualize the impact of the accelerometer bias on the performance, we corrupt the accelerometer measurements by a bias (Fig. 3).

Despite a high accelerometer bias, the closed-form solution still provides robust results. As seen in Fig. 3, neither the estimation of the gravity, the velocity or the lambdas is impacted by the accelerometer bias. To explain this behavior we ran many simulations by also considering trajectories that are not plausible for an MAV and by changing the magnitude of the gravity.

We found the following conclusions. When the rotations are small, the effect of a bias is negligible even if its value is larger than the inertial acceleration. This is easily explained by remarking that, in the case of negligible rotations, a bias on the accelerometer acts as the gravity. Hence, its impact depends on the ratio between its magnitude and the magnitude of the gravity. If the rotations are important, the effect of a bias on the accelerometer is negligible when its magnitude is smaller than both the gravity and the inertial acceleration. Note that, for an MAV that accomplishes a loop of radius 1m and speed 2m s^{-1}, the inertial acceleration is 4m s^{-2}.

In [1], the author provides an alternative formulation of the closed-form solution including the accelerometer bias as an observable unknown of the system. However, the estimation of the accelerometer bias with that method is not robust since our system is only slightly affected by it[3].

D. Impact of gyroscope bias on the performance

To visualize the impact of the gyroscope bias on the performance, we corrupt the gyroscope measurements by an artificial bias (Fig. 4).

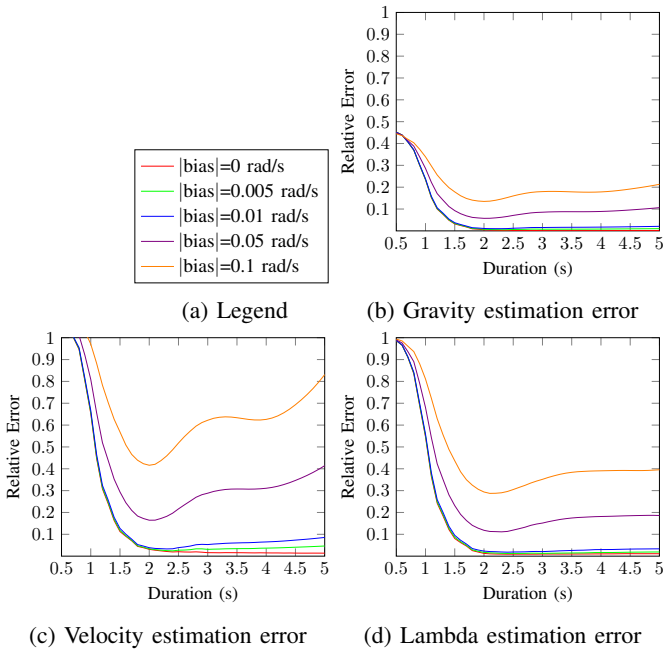
As seen in Fig. 4, the performance becomes very poor in presence of a bias on the gyroscope and, in practice, the overall method could only be successfully used with a very precise—and expensive—gyroscope.

Note that, in [1], the author evaluates the performance of the closed-form solution with a simulated gyroscope bias of magnitude $0.5deg/s \approx 0.0087rad/s$. In Fig. 4, this bias would yield a curve between the green and the blue ones, with relative error below 10%.

V. ESTIMATING THE GYROSCOPE BIAS

Previous work has shown that the gyroscope bias is an observable mode when using an IMU and a camera, which means that it can be estimated [2]. In this section, we propose an optimization approach to estimate the gyroscope bias using the closed-form solution.

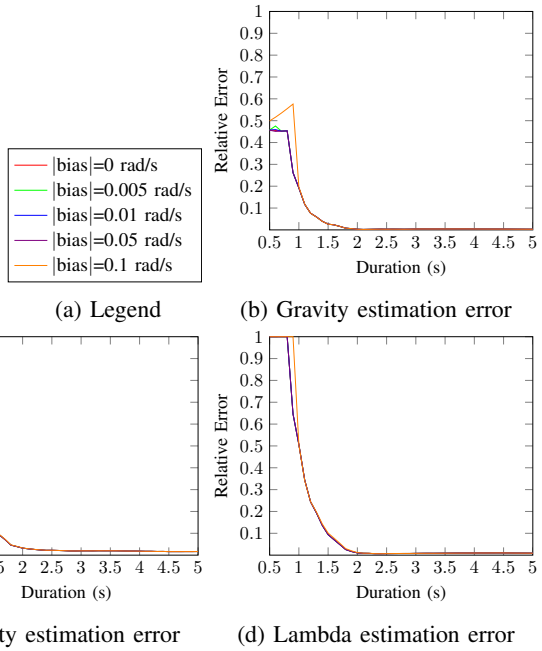
[3]Additionally, in [1] property 12, we prove that rotations must occur around at least two independent axes to determine the bias. In general, for a motion of a few seconds, an MAV accomplishes rotations around a single axis.



(a) Legend (b) Gravity estimation error

(c) Velocity estimation error (d) Lambda estimation error

Fig. 4: Impact of the gyroscope bias on the performance of the closed-form solution. We are observing 7 features over a variable duration of integration.



(a) Legend (b) Gravity estimation error

(c) Velocity estimation error (d) Lambda estimation error

Fig. 5: Impact of the gyroscope bias on the performance of the optimized closed-form solution. We are observing 7 features over a variable duration of integration.

A. *Nonlinear minimization of the residual*

Since our system of equations (1) is overconstrained, inverting it is equivalent to finding the vector X that minimizes the residual $||\Xi X - S||^2$. We define the following cost function:

$$cost(B) = ||\Xi X - S||^2, \tag{4}$$

with:

- B the gyroscope bias;
- Ξ and S computed by replacing the angular velocity provided by the gyroscope ω by $\omega - B$.

By minimizing this cost function, we recover the gyroscope bias B and the unknown vector X. Since our cost function requires an initialization and is non-convex (see Fig. 7), the optimization process can be stuck in local minima. However, by running extensive simulations we found that the cost function is convex around the true value of the bias. Hence, we can initialize the optimization process with $B = 0_3$ since the bias is usually rather small.

As seen in Fig. 6, this method can robustly estimate high values of the gyroscope bias (relative error of final bias estimate is below 2%). Fig. 5 displays the performance of the proposed method in estimating speed, gravity in the local frame, and distances to the features in presence of the same artificial gyroscope bias from Fig. 4. As seen in Fig. 5, after $1s$ of integration duration, the estimations agree no matter how high the bias is. In other words, given that the integration duration is long enough, this method is unaffected by the gyroscope bias. Using Levenberg-Marquardt algorithm, the optimization process reaches its optimal value after around 4 iterations and 20 evaluations of the cost function. Evaluating

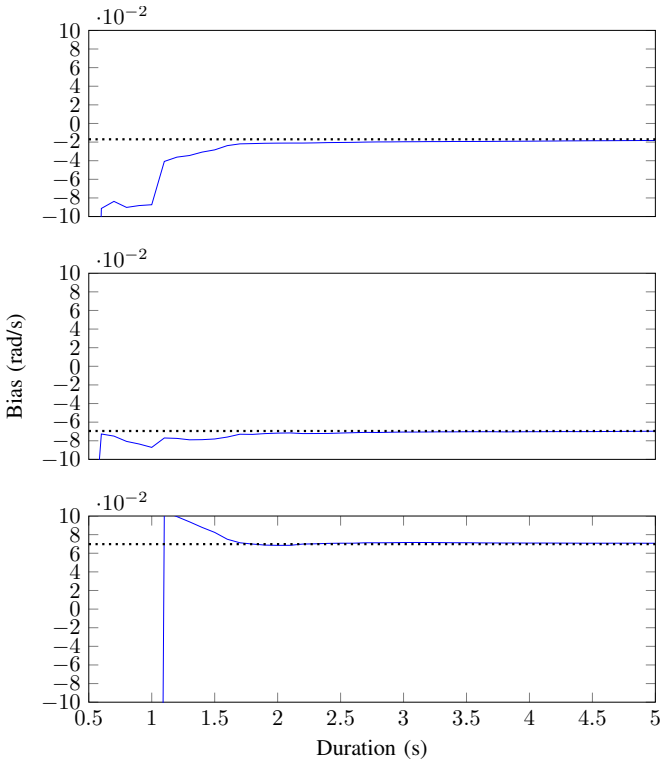
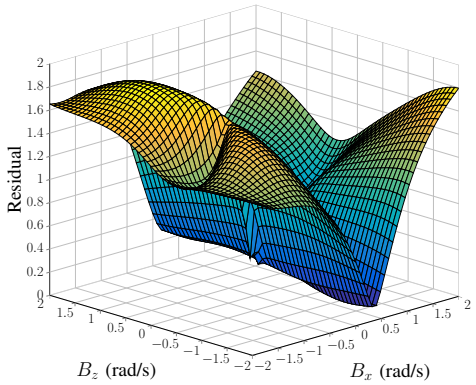
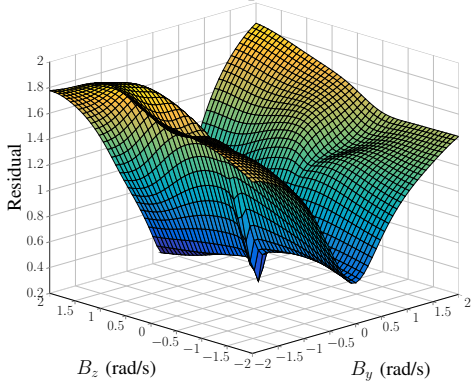


Fig. 6: Gyroscope bias estimation from nonlinear minimization of the residual. We are observing 7 features over a variable duration of integration. The true bias is $B = [-0.0170, -0.0695, 0.0698]$ with magnitude $||B|| = 0.1$ and the final bias estimate is $[-0.0183, -0.0697, 0.0708]$.



(a) Residual with respect to B_x and B_z



(b) Residual with respect to B_y and B_z

Fig. 7: Cost function (residual) with respect to the gyroscope bias for a small amount of available measurements (integration of 1 second while observing 7 features)

the cost function is equivalent to solving the linear system described in Equation (2).

For very short time of integration ($<$ 1 second), the cost function loses its local convexity and the proposed method can fail by providing a gyroscope bias much larger than the correct one. To understand this misestimation, in Fig. 7 we plot the residual with respect to the bias, which is the cost function we are minimizing. We highlight a misestimation of the gyroscope bias by setting the duration of integration to 1 second while observing 7 features. We refer to the components of the gyroscope bias by $B = [B_x, B_y, B_z]$. As we can see in Fig. 7, the cost function admits a symmetry with respect to B_z (and consequently it is not convex). This symmetry replicates the minima of the true gyroscope bias along B_z. The optimization process can therefore diverge from the true gyroscope bias. In the next section, we present a method to use a priori knowledge to guide the optimization process.

B. Removing the symmetry in the cost function

The symmetry in the cost function is induced by the strong weight of the gravity in the Equation (1). In general, the residual is almost constant with respect to the component of the gyroscope bias along the direction $\vec{u}$ when $\vec{u}$ is collinear with the gravity throughout the motion. Since an MAV normally operates in near-hover conditions, $\vec{u}$ is approximated to the vector pointing upward in the gyroscope frame when the MAV is hovering. If the MAV rotates such that $\vec{u}$ becomes noncollinear with the gravity, the cost function does not exhibit this symmetry anymore. In this case, the gyroscope bias is well estimated. A simple solution to avoid having that symmetry in our system would be to enforce that there is no such $\vec{u}$ by forcing our MAV to perform rotations while it is operating. Another way to artificially get rid of this symmetry is to tweak the cost function. Specifically, we can add a regularization term that penalizes high estimations of the component of the bias along $\vec{u}$:

$$cost(B) = ||\Xi X - S||^2 + \lambda(\vec{u} \cdot B)^2, \tag{5}$$

with $\vec{u}$ the direction collinear with the gravity throughout the motion and λ the coefficient given to how much we want to penalize this bias component.

For small values of λ, our cost function is similar to the previous one and the bias can grow arbitrarily high. Note that, instead of forcing this gyroscope bias component to be close to 0, we can easily force it to be close to any value. Therefore, we can use the a priori knowledge of a gyroscope bias approximation:

$$cost(B) = ||\Xi X - S||^2 + \lambda(\vec{u} \cdot (B - B^{approx}))^2,$$

with B^{approx} being the known approximate gyroscope bias. This methods allows us to reuse previously-computed gyroscope bias since it is known to slowly vary over time. The value of λ should be set starting from the knowledge about the range of change of the gyroscope bias. We can obtain this variation with previously-computed gyroscope bias.

VI. EXPERIMENTS ON REAL DATA

We validate our method on a real dataset containing IMU and camera measurements from a flying quadrotor along with ground truth.

A. Experimental setup

For our evaluation, we consider an MAV flying in a room equipped with a motion-capture system. This allows us to compare the estimations of the velocity along with the roll and pitch angles against ground truth.

We use the same MAV used in [18], Section 3.4. Specifically, our quadrotor relies on the frame of the Parrot AR.Drone 2.0 including their motors, motor controllers, gears, and propellers. It is equipped with a PX4FMU autopilot and a PX4IOAR adapter board. The PX4FMU includes a 200Hz IMU. The MAV is also equipped with a downward-looking MatrixVision mvBlueFOX-MLC200w (752×480-pixel) monochrome camera with a 130-degree field-of-view lens (Fig. 8a). The data are recorded using an Odroid-U3 single-board computer. The MAV flies indoors at low altitude (1.5m) (Fig. 8b). The feature extraction and matching is done via the FAST corners [21, 22].

(a) A closeup of our quadrotor: 1) down-looking camera, 2) Odroid U3 quad-core computer, 3) PIXHAWK autopilot.

(b) Our flying arena equipped with an OptiTrack motion-capture system (for ground-truth recording).

Fig. 8: Experimental setup for identifying the limitations of the performance. The drone is equipped with an IMU and a down-looking camera.

B. Results

We compare the performance on the estimations of the gravity and velocity obtained with three methods:

- The original closed-form solution [1] (Eq. (2));
- Our modified closed-form solution (Eq. (4));
- The loosely-coupled visual-inertial algorithm (MSF) [23] using pose estimates from the Semi-direct Visual Odometry (SVO) package [6] (how to combine MSF with SVO can be found in [18]).

The reason we included SVO+MSF in the validation is to have a reference state-of-the-art pose estimation method. However, MSF requires to be initialized with a rough absolute scale, whereas our method works without initialization. We set the integration duration for the closed-form solution to 2.8 seconds, since it is sufficient to obtain robust results (see Fig. 6). The camera provides 60fps, but we discard most of the frames and consider only 10Hz (this is discussed in section IV-A).

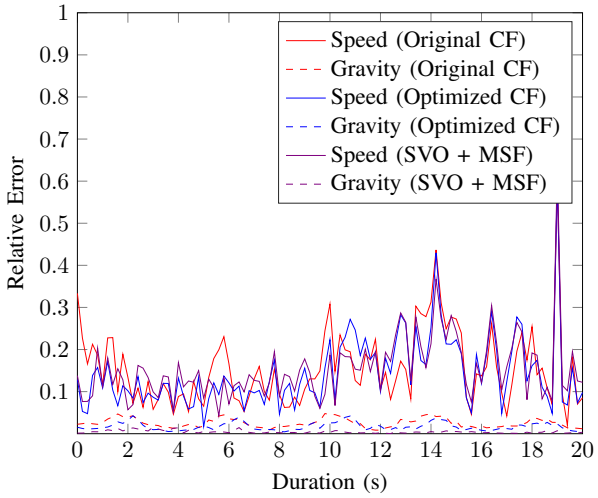
As seen in Fig. 9a, the performance obtained by our method is similar than the performance obtained by a well-initialized MSF. We remind the reader that unlike MSF, the closed-form solution does not require the knowledge of the absolute scale to be provided. Moreover, the original closed-form solution and the optimized closed-form solution have similar performance. Indeed, for this dataset the gyroscope bias was estimated to $B = [0.0003, 0.009, 0.001]$, which is very small $(||B|| = 0.0091)$.

To prove the robustness of our method compared to the original closed-form, we corrupt the gyroscope measurements provided by the dataset with an artificial bias in Fig. 9b and Fig. 9c.

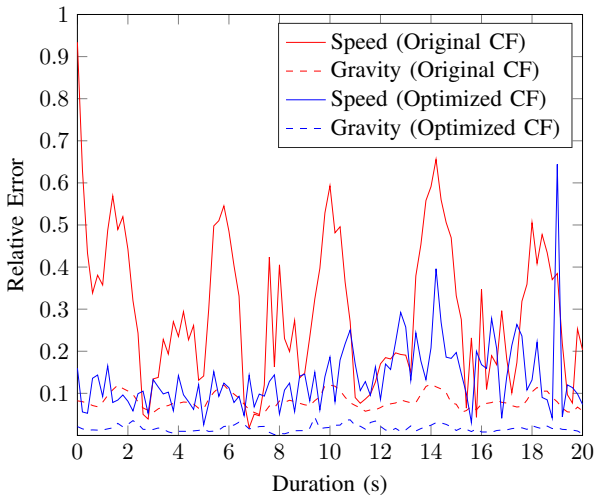
As seen in these figures, our method is robust against gyroscope bias whereas the original closed-form is not.

VII. CONCLUSION

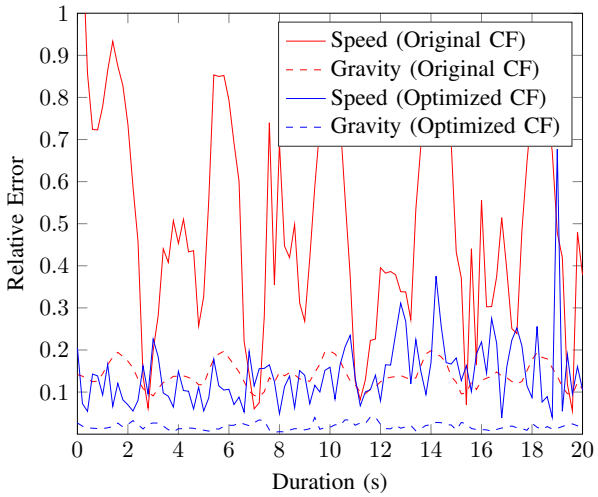
In this paper, we studied the recent closed-form solution proposed by [1] which performs visual-inertial sensor fusion without requiring an initialization. We implemented this method in order to test it with plausible MAV motions and synthetic noisy sensor data. This allowed us to identify its performance limitations and bring modifications to overcome them.



(a) No artificial bias.



(b) Artificial bias of magnitude $0.05 rad/s$.



(c) Artificial bias of magnitude $0.1 rad/s$.

Fig. 9: Estimation error of the optimized closed-form solution against the original closed-form solution [1] and SVO [18]. The duration of integration is set to 2.8 seconds, and 10 point features are observed throughout the whole operation. In Fig. 9b and Fig. 9b, we corrupted the gyroscope measurements with an artificial bias.

We investigated the impact of biased inertial measurements. Although the case of biased accelerometer was originally studied in [1], we showed that the accelerometer bias does not significantly worsen the performance. One major performance limitation of this method was due to the impact of biased gyroscope measurements. In other words, the performance becomes very poor in presence of a bias on the gyroscope and, in practice, the overall method could only be successfully used with a very precise (and expensive) gyroscope. We then introduced a simple method that automatically estimates this bias.

We validated this method by comparing its performance against state-of-the-art pose estimation approach for MAV. For future work, we see this optimized closed-form solution being used on an MAV to provide accurate state initialization. This would allow aggressive take-off maneuvers, such as hand throwing the MAV in the air, as already demonstrated in [19] with a range sensor. With our technique, we could get rid of the range sensor.

REFERENCES

[1] A. Martinelli, "Closed-form solution of visual-inertial structure from motion," *International Journal of Computer Vision (IJCV)*, vol. 106, no. 2, pp. 138–152, 2014.

[2] ——, "Vision and imu data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination," *Transactions on Robotics (T-RO)*, vol. 28, no. 1, pp. 44–60, 2012.

[3] L. Armesto, J. Tornero, and M. Vincze, "Fast ego-motion estimation with multi-rate fusion of inertial and vision," *The International Journal of Robotics Research (IJRR)*, vol. 26, no. 6, pp. 577–589, 2007.

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

[5] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "On the complexity and consistency of ukf-based slam," in *International Conference on Robotics and Automation (ICRA)*, 2009, pp. 4401–4408.

[6] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *International Conference on Robotics and Automation (ICRA)*, 2014.

[7] J. Hesch, D. Kottas, S. Bowman, and S. Roumeliotis, "Consistency analysis and improvement of vision-aided inertial navigation," *Transactions on Robotics (T-RO)*, vol. 30, no. 1, pp. 158–176, 2014.

[8] G. Huang, M. Kaess, and J. J. Leonard, "Towards consistent visual-inertial navigation," in *International Conference on Robotics and Automation (ICRA)*, 2015.

[9] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial odometry using nonlinear optimization," *International Journal of Robotics Research (IJRR)*, vol. 34, no. 3, pp. 314–334, 2014.

[10] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," in *Robotics: Science and Systems XI*, 2015.

[11] A. Mourikis, S. Roumeliotis, *et al.*, "A dual-layer estimator architecture for long-term localization," in *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2008.

[12] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *Transactions on Robotics (T-RO)*, vol. 28, no. 1, pp. 61–76, 2012.

[13] G. P. Huang, A. Mourikis, S. Roumeliotis, *et al.*, "An observability-constrained sliding window filter for slam," in *Intelligent Robots and Systems (IROS)*, 2011, pp. 65–72.

[14] A. Mourikis, S. Roumeliotis, *et al.*, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *International Conference on Robotics and Automation (ICRA)*, 2007, pp. 3565–3572.

[15] V. Indelman, S. Williams, M. Kaess, and F. Dellaert, "Information fusion in navigation systems via factor graph based incremental smoothing," *Robotics and Autonomous Systems*, pp. 721–738, 2013.

[16] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Second. Cambridge, 2004.

[17] S. M. Weiss, "Vision based navigation for micro helicopters," PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 2012.

[18] M. Faessler, F. Fontana, C. Forster, E. Mueggler, M. Pizzoli, and D. Scaramuzza, "Autonomous, vision-based flight and live dense 3d mapping with a quadrotor micro aerial vehicle," *Journal of Field Robotics (JFR)*, 2015.

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

[20] T.-C. Dong-Si and A. Mourikis, "Estimator initialization in vision-aided inertial navigation with unknown camera-imu calibration," in *Intelligent Robots and Systems (IROS)*, 2012.

[21] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking.," in *International Conference on Computer Vision (ICCV)*, 2005.

[22] ——, "Machine learning for high-speed corner detection," in *European Conference on Computer Vision (ECCV)*, vol. 1, 2006, pp. 430–443.

[23] S. Lynen, M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, "A robust and modular multi-sensor fusion approach applied to mav navigation," in *Intelligent Robots and Systems (IROS)*, 2013.