Augmenting Ultra-Wideband Localization with Computer Vision for Accurate Flight

David Hoeller * **Anton Ledergerber** * **Michael Hamer** *
Raffaello D'Andrea *

* *Institute for Dynamic Systems and Control, ETH Zurich*
(e-mail: {dhoeller,antonl,hamerm,rdandrea}@ethz.ch).

Abstract: Ultra-wideband radio networks enable low-cost, low-computation robot localization in semi-structured environments; however, previous results have shown that these localization systems suffer from spatially-varying measurement biases, leading to a spatially-varying offset between the physical and the estimated position. In tasks where absolute positioning or high tracking accuracy is required, this offset can lead to failure of the task. This paper proposes augmenting ultra-wideband-based localization with visual localization to improve estimation accuracy for critical tasks. It also presents a control strategy that takes the camera measurement process into account, and allows the ultra-wideband system's measurement biases to be learned and compensated over multiple executions of the task. This bias compensation can be used to improve the accuracy of the task in the case of visual impairment. The effectiveness of the proposed framework is demonstrated by accurately flying a quadrocopter to a landing platform using on-board estimation and control.

Keywords: Flying robots, Perception and sensing, Information and sensor fusion, Motion control systems, Non-linear predictive control

1. INTRODUCTION

Recent progress in low-cost, low-computation indoor localization using ultra-wideband (UWB) radio networks has enabled the use of drones in indoor environments. However, UWB localization systems have been shown to suffer from spatially-varying measurement biases (Ye et al., 2010; Prorok and Martinoli, 2014), which can lead to a spatially-varying offset between the physical and the estimated position (Mueller et al., 2015; Ledergerber et al., 2015). In applications requiring drones to accurately perform maneuvers, for example, a maneuver to land on a charging-station or on a package in a warehouse, this offset can lead to failure.

In this paper, we augment a quadrocopter flying in a UWB localization system with an onboard camera and demonstrate that the camera enables the accurate execution of a landing maneuver. The combination of UWB localization and onboard vision is advantageous: absolute position, in our system with an accuracy of ±25cm, can be obtained from UWB measurements quickly and with a low computational cost; meanwhile, the onboard vision sensor provides improved local accuracy if the task requires it. This is in comparison with existing methods of indoor localization, such as visual localization, which require significant processing, and may not provide accurate positioning information in situations where the configuration and visual state of the environment is constantly changing, for example in modern, robotic warehouses (e.g. Wurman et al. (2007)).

Using computer vision to assist with drone maneuvers has been the focus of considerable research. One example

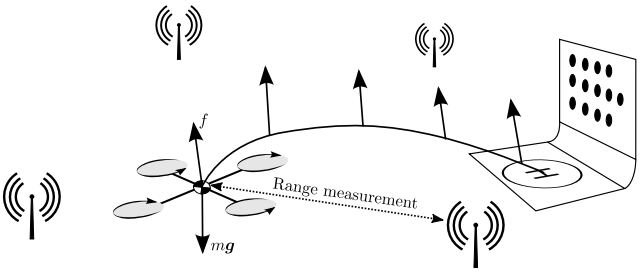

Fig. 1. An overview of the system presented in this paper. The quadrocopter localizes itself based on UWB range measurements with an accuracy of ±25cm. This is too inaccurate to successfully perform a landing maneuver. The quadrocopter therefore relies on camera measurements for additional accuracy during the landing maneuver. A control strategy was developed that takes into account the camera measurement process. This control strategy further allows the UWB biases to be learned over multiple maneuvers, allowing an accurate trajectory to be tracked by the UWB system in the case of temporary visual impairment.

of early work on this topic is Amidi et al. (1999), who applied visual odometry to improve the performance of autonomous helicopters. More recent publications, e.g. Faessler et al. (2014); Lange et al. (2009); Yang et al. (2013); Wenzel and Zell (2009) extend this work specifically for take-off and landing maneuvers. In addition, many recent consumer drones (e.g. the Parrot AR.Drone and the DJI Mavic drone) use computer vision to stabilize motion.

This paper leverages this research, employing an approach based on planar artificial markers for vision-based state

estimation during a landing maneuver. While such an approach was also chosen by Benini et al. (2013) to augment a UWB localization system with visual localization, the main contributions of this paper are found in the extensions of their work, namely the control strategy developed specifically for this kind of state estimation, as well as the framework used to compensate the biases of the UWB localization system during a maneuver. This bias compensation can be used to improve the accuracy of the landing maneuver in the case of visual impairment. Fig. 1 illustrates the landing procedure discussed in this paper, where trajectories are generated such that the onboard camera faces a known pattern next to the landing site and the UWB biases are learned along these trajectories.

The paper is structured as follows: Section 2 provides a brief overview of the camera-based state estimator and Section 3 lays out the dynamics of the quadrocopter. In Section 4, a control strategy is presented which takes the camera measurement process into account. Section 5 explains how the sensor measurements can be combined to learn and compensate for the biases of the UWB localization system, allowing a trajectory to be tracked in the case of temporary visual impairment. Finally, Section 6 presents details of the implementation and discusses experimental results.

2. VISION-BASED POSE ESTIMATION

This section describes the estimation of the quadrocopter's pose from the image of a known pattern placed near the landing platform.

Camera model A pinhole model is used to describe the image formation process, as shown in Fig. 2. Assuming the camera has been calibrated (e.g. as described by Zhang (2000)), a point $\mathbf{p} = (x, y, z)$ in the world projected onto the camera's sensor plane results in the pixel measurement

$$\begin{bmatrix} u \\ v \end{bmatrix} = \Phi\left(\mathbf{p},\ \mathbf{T}_{\mathrm{CW}}\right), \tag{1}$$

where $\mathbf{T}_{\mathrm{CW}} \in \mathrm{SE}(3)$ is the pose of the camera's center relative to the inertial frame, and where $\Phi(\cdot,\cdot)$ is the calibrated, non-linear projection function as described in Hartley (2000) and Szeliski (2010).

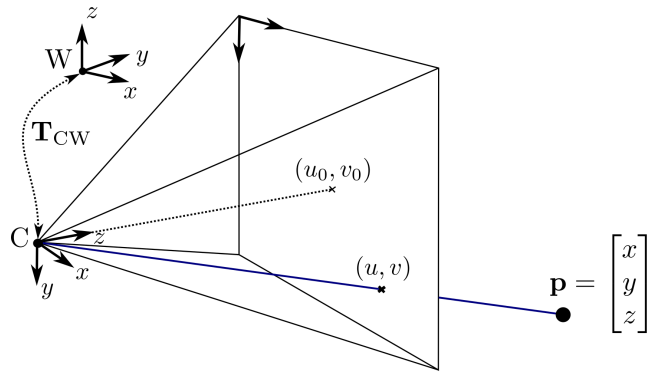

Fig. 2. Representation of the pinhole camera model. A point $\mathbf{p}$ is projected onto the sensor plane resulting in the pixel measurement (u, v). The pose of the camera is encoded via the transformation $\mathbf{T}_{\mathrm{CW}}$.

Pose estimation Given a set of points $\mathcal{P}$, where point $i \in \mathcal{P}$ has the known location $\mathbf{p}_i$ relative to the inertial frame, and measured pixel coordinates (u_i, v_i), the camera's pose $\mathbf{T}_{\mathrm{CW}}$ relative to the inertial frame can be estimated by solving the following non-linear least squares problem

$$\min_{\mathbf{T}_{\mathrm{CW}}} \sum_{i \in \mathcal{P}} \left\| \begin{bmatrix} u_i \\ v_i \end{bmatrix} - \Phi\left(\mathbf{p}_i,\ \mathbf{T}_{\mathrm{CW}}\right) \right\|^2, \tag{2}$$

where $\|\cdot\|$ is the Euclidean norm. If the points are in the same plane, an initial guess for the pose can be found by calculating the parameters of the homography relating the points in the world and the pixel measurements. The pose of the quadrocopter in the inertial frame can then be estimated as $\mathbf{T}_{\mathrm{BW}} = \mathbf{T}_{\mathrm{BC}}\mathbf{T}_{\mathrm{CW}}$, where $\mathbf{T}_{\mathrm{BC}} \in \mathrm{SE}(3)$ is the known transformation from the camera's center to the quadrocopter's body frame.

For simplicity, our implementation (Section 6) is based on a blob-detection approach, whereby the landing platform is marked by a pattern (as shown in Fig. 1), whose known layout and projection on the camera's sensor provides the information required for (2). The main limitation is that the pattern must be visible at all times and the method is therefore sensitive to occlusions. However, as we later discuss in Section 5, by learning and compensating for the UWB network's spatially-varying biases along the landing trajectory, position estimates derived from the UWB network can help ensure accurate tracking during periods of temporary visual impairment, e.g. due to marker occlusion or motion blur.

The accuracy of the resulting pose estimate can be quantified by means of backward transport of covariance (Hartley, 2000, pp. 138-150). For the planar pattern depicted in Fig. 1, the accuracy improves when the angle with respect to the pattern normal is increased.

3. DYNAMICS AND STATE ESTIMATION

We assume the quadrocopter is a rigid body of mass m able to produce a positive thrust f along its body's z-axis. The body's angular rates are defined in the body frame as $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3)$, where the notation (x, y, z) is used to succinctly express elements of a vector. We define the quadrocopter to have position $\mathbf{x}(t) \in \mathbb{R}^3$, relative to the inertial frame, and an orientation $\mathbf{R}_{\mathrm{WB}}(t) \in \mathrm{SO}(3)$, which expresses a rotation from the quadrocopter's body frame to the inertial frame. The continuous-time quadrocopter dynamics are then given by

$$\begin{aligned} \ddot{\mathbf{x}} &= \mathbf{R}_{\mathrm{WB}} \frac{\mathbf{e_3} f}{m} + \mathbf{g} \\ \dot{\mathbf{R}}_{\mathrm{WB}} &= \mathbf{R}_{\mathrm{WB}} [\![\boldsymbol{\omega}]\!]_\times, \end{aligned} \tag{3}$$

where $\mathbf{g} = (0, 0, -g)$ is the acceleration due to gravity, $\mathbf{e_3} = (0, 0, 1)$ and $[\![\boldsymbol{\omega}]\!]_\times$ is the matrix form of the cross product, defined as

$$[\![\boldsymbol{\omega}]\!]_\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \tag{4}$$

We use the angular rates $\boldsymbol{\omega}$ and the thrust f as control inputs (see Fig. 3) and assume that they can be controlled without delay. This assumption is reasonable, since the

motors' time constants are much smaller than those of the other dynamics.

In an approach similar to Augugliaro et al. (2012), the dynamics (3) are discretized for state estimation and trajectory generation (Section 4). We define the discrete-time state of the quadrocopter at discrete-time index k as $\mathbf{q}[k] = (\mathbf{x}[k], \mathbf{v}[k], \mathbf{R}_{\mathrm{WB}}[k])$, where $\mathbf{x}[k] \in \mathbb{R}^3$ and $\mathbf{v}[k] \in \mathbb{R}^3$ are respectively the position and velocity of the quadrocopter expressed in the inertial frame, and where $\mathbf{R}_{\mathrm{WB}}[k] \in \mathrm{SO}(3)$ expresses a rotation from the quadrocopter's body frame to the inertial frame. Note that the quadrocopter's body rates $\boldsymbol{\omega}$ are not included in the state, since they are used as inputs to the system. Discretization for the fixed sampling period Δt, assuming a zero-order input hold, then yields

$$\begin{aligned}
\mathbf{x}[k+1] \approx \;& \mathbf{x}[k] + \mathbf{v}[k]\Delta t \\
& + \mathbf{R}_{\mathrm{WB}}[k]\Delta t^2 \left(\frac{1}{2}\mathbf{I} + \frac{1}{6}\Delta t [\![\boldsymbol{\omega}[k]]\!]_\times\right) \mathbf{e_3} \frac{f[k]}{m} \\
& + \frac{1}{2}\mathbf{g}\Delta t^2, \\
\mathbf{v}[k+1] \approx \;& \mathbf{v}[k] + \mathbf{R}_{\mathrm{WB}}[k]\Delta t \left(\mathbf{I} + \frac{1}{2}\Delta t [\![\boldsymbol{\omega}[k]]\!]_\times\right) \mathbf{e_3} \frac{f[k]}{m} \\
& + \mathbf{g}\Delta t, \\
\mathbf{R}_{\mathrm{WB}}[k+1] = \;& \mathbf{R}_{\mathrm{WB}}[k]\mathbf{exp}([\![\boldsymbol{\omega}[k]]\!]_\times \Delta t),
\end{aligned} \tag{5}$$

where the first order approximation

$$\mathbf{exp}([\![\boldsymbol{\omega}[k]]\!]_\times \Delta t) \approx \mathbf{I} + [\![\boldsymbol{\omega}[k]]\!]_\times \Delta t \tag{6}$$

was used to discretize the updates of $\mathbf{x}$ and $\mathbf{v}$. The discrete-time acceleration $\mathbf{a}[k] \in \mathbb{R}^3$ in the inertial frame is therefore given by

$$\mathbf{a}[k] = \mathbf{R}_{\mathrm{WB}}[k] \frac{\mathbf{e_3} f[k]}{m} + \mathbf{g}. \tag{7}$$

An extended Kalman filter, similar to the filter described in Mueller et al. (2015), is used to track the state of the quadrocopter. Gyroscope and accelerometer measurements enter directly as inputs in the process update step, while camera pose estimates are used as measurement updates.

4. LANDING MANEUVER

As previously discussed, the state estimator relies on the camera's pose estimate, calculated based on the image of the blob pattern. This results in an additional constraint on the planned trajectory in order to ensure that the camera faces the pattern. In this section, we formulate the generation of landing maneuvers as an optimization problem including the quadrocopter's dynamics and the camera orientation constraint, and discuss how these trajectories can be tracked by means of a non-linear model predictive control (MPC) problem.

4.1 Trajectory Generation and Tracking

We express the trajectory generation problem as an optimization problem over the finite time-horizon N, and solve for the control inputs $\boldsymbol{\omega}[k]$ and $f[k]$ at each time step $k = 0, \ldots, N-1$. This formulation allows a smooth trajectory to be obtained by minimizing the jerk of the trajectory, while ensuring that the discrete-time dynamics (5) are fulfilled.

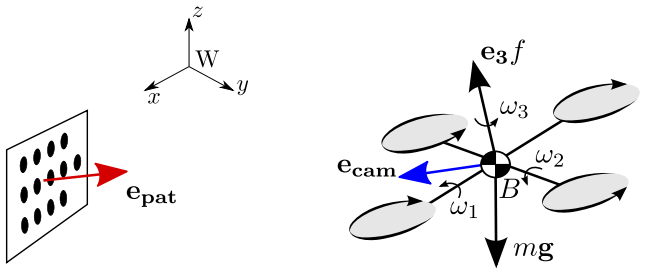

Fig. 3. Model of the quadrocopter. The inputs are the angular rates ω_1, ω_2, ω_3, and thrust f. The vector $\mathbf{e}_{\mathrm{cam}}$ (blue) describes the orientation of the rigidly-mounted camera with respect to the quadrocopter's body frame, and the vector $\mathbf{e}_{\mathrm{pat}}$ (red) is defined to point from the center of the pattern to the center of the camera.

Furthermore, defining $\mathbf{e}_{\mathrm{cam}}$ as the unit-vector expressing the orientation of the rigidly-mounted camera with respect to the quadrocopter's body frame, and $\mathbf{e}_{\mathrm{pat}}$ as the unit vector pointing from the pattern to the camera's center (see Fig. 3), we additionally penalize deviation of the camera's orientation from the pattern's center by including the dot-product $\mathbf{R}_{\mathrm{WB}}[k]\mathbf{e}_{\mathrm{cam}}$ and $\mathbf{e}_{\mathrm{pat}}$. This term reaches its minimum of -1 when the vectors are facing each other. This results in the optimization problem

$$\begin{aligned}
\text{minimize} \quad & \alpha \sum_{k=0}^{N-2} \left|\left| \frac{\mathbf{a}[k+1] - \mathbf{a}[k]}{\Delta t} \right|\right|^2 \\
& + (1-\alpha) \sum_{k=1}^{N} \left(\mathbf{R}_{\mathrm{WB}}[k]\mathbf{e}_{\mathrm{cam}}\right)^T \mathbf{e}_{\mathrm{pat}}[k] \\
\text{subject to} \quad & f_{\min} \leq f[k] \leq f_{\max}, \; k = 0, \ldots, N-1 \\
& \boldsymbol{\omega}_{\min} \leq \boldsymbol{\omega}[k] \leq \boldsymbol{\omega}_{\max}, \; k = 0, \ldots, N-1 \\
& \mathbf{q}[0] = \mathbf{q}_{\mathrm{init}}, \; \mathbf{q}[N] = \mathbf{q}_{\mathrm{final}}
\end{aligned} \tag{8}$$

with $\alpha \in [0, 1]$ a weight factor and $\mathbf{q}_{\mathrm{final}}$ the desired final state defined up to the quadrocopter's yaw, which is left as a free state. The control inputs are subject to box constraints. This results in $4N$ optimization variables with nine equality constraints and $8N$ inequality constraints. Note that due to the problem formulation, the discrete dynamics of Section 3 are contained in the objective function and are not used as constraints. Also note that the initial conditions are fulfilled by definition.

The initial position of the maneuvers is chosen to ensure a sufficient angle with respect to the pattern normal. As described in the previous section, this results in more accurate camera pose estimates throughout the trajectory.

This non-linear optimization problem can be solved using sequential quadratic programming (Nocedal and Wright, 2006) where the non-convex objective function and constraints are iteratively approximated by quadratic and affine functions, resulting in a convex optimization problem that can be solved in real time by the quadrocopter's onboard processor (Section 6).

The resulting optimal control inputs define a trajectory which could be supplied as a reference to a standard trajectory-tracking controller. However, instead of focusing on the camera pointing constraint, too much control effort might be dedicated to tracking the trajectory, which

could result in the camera losing sight of the pattern in the case of a deviation from the planned trajectory.

In order to maintain camera-pattern alignment throughout the trajectory, a non-linear model predictive control approach was chosen, in which the optimization problem is solved every 20ms starting from the current state estimate. The resulting control inputs are then applied to the system until a new solution becomes available (as in, e.g. Hehn and D'Andrea (2011)). While the number of time steps N is held constant throughout the maneuver to keep the computational cost similar, the time allocated for each new trajectory is reduced by the time elapsed since the beginning of the maneuver resulting in a decreasing time step length Δt along the maneuver.

5. UWB BIAS ESTIMATION

While the camera provides a good pose estimate when facing the pattern, it is sensitive to illumination conditions and visual impairment. In the previous sections we have discussed the generation of landing maneuvers based on visual localization. In this section we derive a framework to learn the spatially-varying biases of the UWB measurements. These biases can be used to compensate the measurements during future maneuvers. Thus, the accuracy of the UWB localization system is improved, adding a layer of robustness against temporary visual impairment.

Mueller et al. (2015) present a UWB localization system capable of measuring the distance between an antenna on the quadrocopter and fixed antennas on the ground, known herein as anchors. Mueller et al. (2015) note that each range measurement appears to have a systematic, spatially varying bias; however, as this bias is unobservable without a secondary localization system, it is not included in their measurement model. Having augmented the quadrocopter with a visual localization system able to provide accurate and absolute pose estimates, this paper is able to extend the system of Mueller et al. (2015) by additionally considering the bias in the range measurement model.

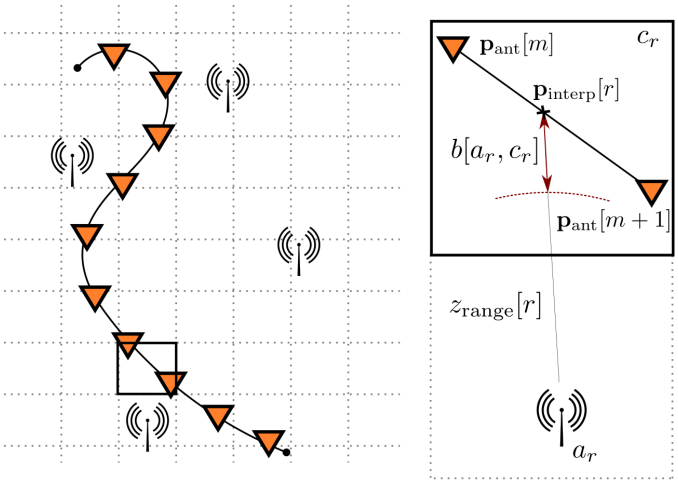

Fig. 4. Overview of the bias estimation framework. The UWB range measurements and the estimated positions of the antenna are logged along a trajectory. The bias to each anchor is assumed to be constant in each cube.

5.1 UWB Bias Modelling

As depicted in Fig. 4, in order to estimate the bias for each anchor in each cube, the camera, IMU, and UWB measurements should be logged when the quadrocopter performs a maneuver and an optimization run after completing the maneuver in order to estimate the biases that maximize the measurement likelihood along the trajectory. Due to the large amount of data, this high dimensional problem quickly becomes computationally intractable on the embedded platform. Forster et al. (2015) approach a similar problem in visual-inertial odometry by factorizing the IMU measurements, thus reducing the problem's dimensionality. In our case, dimensionality is reduced by logging the UWB range measurements when they arrive, as well as the estimated position of the quadrocopter's UWB antenna (derived from the fusion of camera and IMU measurements) whenever a measurement update with a camera pose estimate is performed (i.e. at a frequency of 60 Hz).

We denote the set of antenna position measurements as $\mathcal{M}$. These measurements mark the times and locations along the trajectory at which a camera update was performed. Given a measurement $m \in \mathcal{M}$, the position of the quadrocopter's UWB antenna in the inertial frame is measured as

$$\mathbf{z}_{\text{ant}}[m] = \mathbf{p}_{\text{ant}}[m] + \boldsymbol{\eta}_{\text{ant}}, \tag{9}$$

where $\mathbf{p}_{\text{ant}}[m]$ denotes the actual position of the antenna, and where the measurement of this position is corrupted by an assumed white and normally-distributed measurement noise $\boldsymbol{\eta}_{\text{ant}} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\text{ant}})$ with zero mean and covariance $\boldsymbol{\Sigma}_{\text{ant}}$.

As shown in Fig. 4, UWB range measurements to an anchor are corrupted by a systematic bias based on the quadrocopter's position, and the anchor to which the range is measured. The range bias for each anchor is assumed to be constant in time and spatially varying. In order to model the range biases, the world is segmented into cubes, in which the range bias for each anchor is assumed to be constant. This approach is similar to the approach of Prorok and Martinoli (2014), who use spatially anchored error models.

As the quadrocopter flies the trajectory, range measurements are collected at a rate of 200 Hz. Since this is more frequent than measurements of the antenna's position are recorded (60 Hz, since this occurs upon updating with a camera image), multiple range measurements will occur between two position measurements. For a range measurement r, we define $\mathbf{p}_{\text{interp}}[r]$ to be the position of the quadrocopter's antenna in the inertial frame at the time of recording the range measurement r, and relate it to the antenna position of the previous measurement $\mathbf{p}_{\text{ant}}[m]$ and next measurement $\mathbf{p}_{\text{ant}}[m+1]$ using linear interpolation based on the measurement times. Although it is possible to record the quadrocopter's state estimate upon the reception of every range measurement, by relating the position of a range measurement to the position of a camera measurement, we significantly reduce the number of optimization variables, hence making this problem computationally tractable on the embedded system.

We denote the set of all range measurements $\mathcal{R}$, and for a range measurement $r \in \mathcal{R}$ define: a_r to be the anchor index and $\mathbf{p}_{\text{anch}}[a_r]$ to be the known position of the anchor to which the range was measured; c_r to be the cube index in which the range was measured; and $b[a_r,\ c_r]$ to be the bias of a measurement to anchor a_r in cube c_r. Note that multiple range measurements made to the same anchor in the same cube will have the same bias. We further define the value of the range measurement to be

$$z_{\text{range}}[r] = \|\mathbf{p}_{\text{anch}}[a_r] - \mathbf{p}_{\text{interp}}[r]\| + b[a_r,\ c_r] + \eta_{\text{range}}, \tag{10}$$

where the measurement of this range is corrupted by an assumed white and normally-distributed measurement noise $\eta_{\text{range}} \sim \mathcal{N}(0, \Sigma_{\text{range}})$ with zero mean and covariance Σ_{range}.

The problem of estimating the biases and path flown during a maneuver can be expressed as the following non-linear least squares minimization problem over $\mathbf{p}_{\text{ant}}[m]\ \ \forall m \in \mathcal{M}$ and $b[a_r,\ c_r]\ \ \forall r \in \mathcal{R}$:

$$\min \sum_{r\in\mathcal{R}} \left\|\Sigma_{\text{range}}^{-1}\rho_{\text{range}}[r]\right\|^2 + \sum_{m\in\mathcal{M}} \left\|\boldsymbol{\Sigma}_{\text{ant}}^{-1}\boldsymbol{\rho}_{\text{ant}}[m]\right\|^2 \tag{11}$$

where

$$\begin{aligned}\rho_{\text{range}}[r] &= z_{\text{range}}[r] - (\|\mathbf{p}_{\text{anch}}[a_r] - \mathbf{p}_{\text{interp}}[r]\| + b[a_r,\ c_r])\\ \boldsymbol{\rho}_{\text{ant}}[m] &= \mathbf{z}_{\text{ant}}[m] - \mathbf{p}_{\text{ant}}[m],\end{aligned} \tag{12}$$

and recalling that $\mathbf{p}_{\text{interp}}[r]$ is a function of the optimization variables $\mathbf{p}_{\text{ant}}[m]$. This problem formulation is attractive: it is computationally tractable on the embedded system (e.g. it can be solved quickly using the Levenberg-Marquardt algorithm (Moré, 1978)); and, under the prior assumption of Gaussian measurement noise ($\boldsymbol{\eta}_{\text{ant}}$ and η_{range}), results in the maximum likelihood estimate (Bishop, 2006). To maintain reasonable solving times on the embedded system, longer trajectories are segmented into sub-trajectories and the problem is solved for each portion.

The main disadvantage of this technique is that only the spatially varying bias is learned; whereas, as shown by Ye et al. (2010), the bias also appears to depend on the orientation of the quadrocopter. However, since the maneuvers traverse similar cubes with a similar orientation, this does not appear to have a significant effect on the situation described in this paper, as is shown in Section 6.

6. EVALUATION

This section describes the implementation details and the experimental results assessing the camera-based estimator, the bias estimation framework, and the landing maneuvers.

6.1 System Architecture

The quadrocopter used for the experiments is an Ascending Technologies Hummingbird that was modified to run with a Snapdragon Flight computer, which is equipped with both a Quad-core 2.26 GHz Krait CPU and a DSP. The camera used to record images at a resolution of 1920x1080 (1080p) is a Sony IMX135 camera. The IMU is a low-cost Invensense MPU-9250 and the UWB radios are Decawave DWM1000 modules.

The algorithms presented in this paper are implemented based on the Flying Machine Arena framework (see Lupashin et al. (2014)), which is used to issue high level commands, and additionally to provide accurate motion capture data for comparison purposes. Every algorithm presented in this paper runs onboard the quadrocopter and without the help of the external motion capture system.

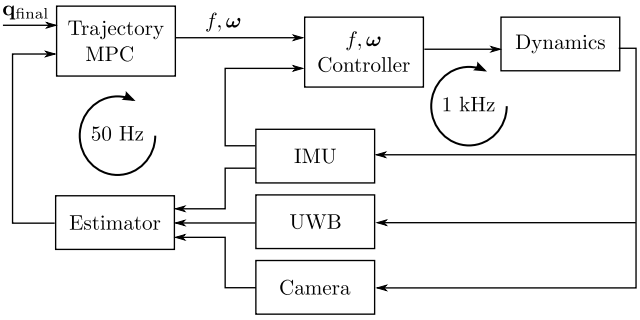

Fig. 5. Block diagram of the system. The estimator obtains measurements from the IMU, UWB radios, and camera. Based on the state estimate and the desired end state and time, the MPC generates a control input that is tracked by the angular rate and thrust controller.

The non-linear optimization problem of the trajectory generation step is solved with the open source library NLopt (Johnson (2007)) implementing the algorithm described in Kraft (1988) and Kraft (1994).

The block diagram of the system is depicted in Fig. 5. The estimator obtains measurements from the IMU at 1 kHz, from the UWB anchors at 200 Hz, and from the camera at 60 Hz. Given the current state estimate, the desired end state and the time to reach it, the MPC calculates a series of control inputs at a frequency of 50 Hz. These commands are tracked with an angular rate and thrust controller which runs at 1 kHz and uses the gyroscopic sensor for feedback. This controller is further described in Appendix B of Lupashin et al. (2014).

6.2 Setup

The experimental area was equipped with four UWB anchors arranged in a rhombus at heights varying from 0 to 3m. The landing platform is a 0.38 m × 0.38 m square whose center is placed 0.7 m in front of the pattern.

In order to estimate UWB biases along the trajectory, the space was divided into cubes with a side length of 0.2 m.

6.3 Results

Slow Triangular Maneuver The first experiment was designed to assess the performance of the camera-based estimator and the bias estimation framework. A slow, horizontal trajectory at the same height as the pattern's center was repeatedly flown, after which the bias estimation framework was run. In order to assess the impact of the bias compensation step, the trajectory was then reflown and two estimators fusing only the IMU and UWB

measurements were run simultaneously. The first used raw UWB range measurements, while the second also corrected the range with the estimated UWB bias. This mimics the impact of an occlusion in the vision system during a trajectory. Fig. 6 shows range measurements in the reflown trajectory to two different anchors with and without bias compensation. Spikes in the range measurements to the first anchor (as seen in Fig. 6) arise when the UWB radio signal has to traverse the quadrocopter. Due to the assumption that the bias is constant in a cube, the estimator cannot handle such outliers well.

The mean absolute error of the IMU-Camera EKF position is 0.036 m in the reflown trajectory.

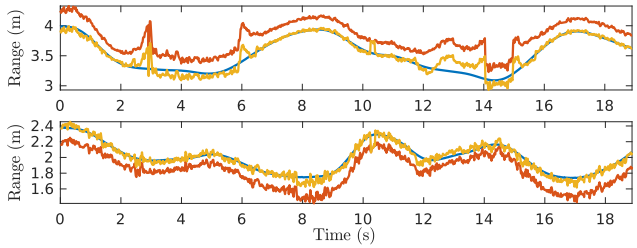

Fig. 6. Range measurements for two anchors in the first experiment. The ground truth measurements are in blue, the raw range measurements in red, and the bias compensated measurements in yellow.

The position estimates of the two estimators in the second trajectory are presented in Fig. 7. The mean absolute error of the position estimate is reduced from 0.30 m to 0.12 m when compensating for range biases.

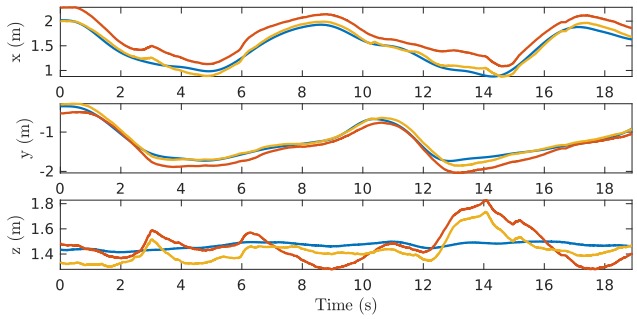

Fig. 7. This figure shows the effect of bias compensation on the position estimate. Shown in red, the position is estimated based on raw UWB ranges, while in yellow, the position is estimated based on bias-compensated range measurements. The position as measured by the motion capture system is shown in blue.

Landing Maneuver The goal of the second experiment is to assess the landing maneuvers. For that purpose, 3 m long maneuvers with $T = 3.4\,\text{s}$, $N = 8$ (see Section 4) were performed. The optimization problem (8) used to generate the control inputs for these maneuvers can be solved for different values of the weighting factor α, resulting in different trajectories. Two such trajectories for $\alpha = 0$ and $\alpha = 0.1$ respectively are shown in Fig. 8. While an $\alpha = 0$ leads to a better alignment of the camera and the pattern, it is not clear that this also leads to a better camera pose estimate. Weighting also the jerk leads to smoother trajectories with lower peak velocities, hence this also reduces the motion blur impairing the pose estimates. Therefore, the value of α should be adapted to the time available for a maneuver and the exposure time. For our landing maneuver a value of $\alpha = 0.1$ was chosen.

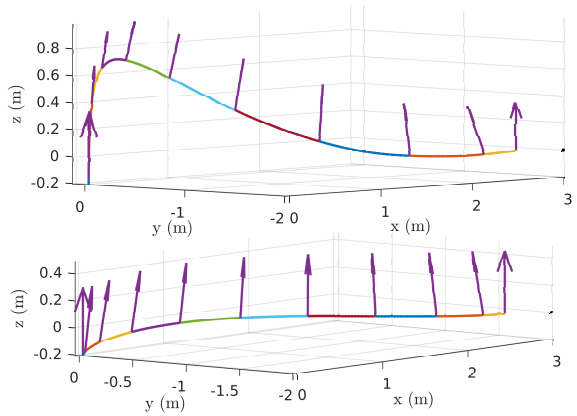

Fig. 8. Two trajectories generated by solving the optimization problem (8). The upper one with $\alpha = 0$ better keeps the camera aligned whereas the lower one with $\alpha = 0.1$ leads to smoother trajectories and hence also less motion blur.

The landing maneuver was flown multiple times, resulting in a mean absolute position estimation error across all maneuvers of 0.063 m. Very similar results are obtained when including bias-compensated UWB measurements, due to the accuracy of the camera's pose estimates in comparison to the estimates from the UWB system.

To simulate the effect of occlusions, UWB range biases were learned based on the initial camera-based flights, after which the maneuver was flown multiple times with the camera disabled and the estimator fusing only the IMU and UWB measurements. When compensating for UWB biases, the absolute position error across all maneuvers was 0.14 m. Without bias estimation, the position error was 0.21 m.

These results support the idea that bias estimation is necessary for accurate flight of dynamic maneuvers without a camera. It should be noted that these experiments correspond to the extreme case of visual impairment, where the camera is unable to provide pose estimates for the entire duration of the trajectory.

Fig. 9 shows the first trajectory that is calculated by the MPC in blue along with the ground truth trajectory finally flown in green. The red rectangles are the pattern and the landing platform. The discrepancy between the first calculated trajectory and the one that was performed highlights the limitations of the simple motion model used for the quadrocopter. This model does not include the propeller dynamics and also neglects drag. The advantage of such a formulation is its simplicity, making it usable for most multirotors since it does not depend on the vehicle's geometry.

7. CONCLUSION

This paper first discusses a suitable control strategy for a quadrocopter estimating its state by taking images of a

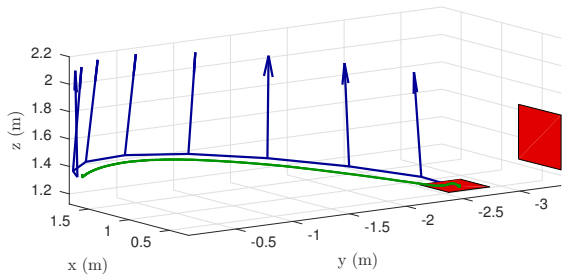

Fig. 9. Trajectory generated by the MPC during a landing maneuver. The red rectangles represent the pattern and the landing platform. The blue curve corresponds to the first trajectory calculated by the MPC, where the vectors represent the thrust at each time step. The green trajectory is the ground truth trajectory.

known pattern with a rigidly mounted onboard camera. While we only applied this control strategy to land the quadrocopter on a landing platform, the general problem formulation also allows its usage for tracking objects, e.g. for outdoor filming. The second part of the paper shows how the precise state estimate resulting from this control strategy can be used to learn the biases of the UWB localization system such that also under visual impairment precise maneuvers can be flown.

ACKNOWLEDGEMENTS

The authors would like to thank Marc-Andrè Corzillius for the design of the electrical hardware and Michael Egli for construction of the platform. Many people have contributed to the Flying Machine Arena in which this research was conducted. A list of these people can be found at: `http://flyingmachinearena.org/people`.

This work was supported by the Swiss National Science Foundation (SNSF).

REFERENCES

Amidi, O., Kanade, T., and Fujita, K. (1999). A visual odometer for autonomous helicopter flight. *Journal of Robotics and Autonomous Systems*, 28, 185 – 193.

Augugliaro, F., Schoellig, A.P., and D'Andrea, R. (2012). Generation of collision-free trajectories for a quadrocopter fleet: A sequential convex programming approach. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1917–1922.

Benini, A., Mancini, A., and Longhi, S. (2013). An IMU/UWB/vision-based extended Kalman filter for mini-UAV localization in indoor environment using 802.15.4a wireless sensor network. *Journal of Intelligent & Robotic Systems*, 70(1-4), 461–476.

Bishop, C.M. (2006). *Pattern recognition and machine learning*. Springer.

Faessler, M., Mueggler, E., Schwabe, K., and Scaramuzza, D. (2014). A monocular pose estimation system based on infrared leds. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 907–913.

Forster, C., Carlone, L., Dellaert, F., and Scaramuzza, D. (2015). On-manifold preintegration theory for fast and accurate visual-inertial navigation. *CoRR*.

Hartley, R. (2000). *Multiple view geometry in computer vision*. Cambridge University Press.

Hehn, M. and D'Andrea, R. (2011). Quadrocopter trajectory generation and control. *IFAC Proceedings Volumes*, 18th IFAC World Congress.

Johnson, S.G. (2007). The NLopt nonlinear-optimization package. URL `http://ab-initio.mit.edu/nlopt`.

Kraft, D. (1988). A software package for sequential quadratic programming. Technical report, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen.

Kraft, D. (1994). Algorithm 733: TOMP-Fortran modules for optimal control calculations. *ACM Transactions on Mathematical Software*, 20(3), 262–281.

Lange, S., Sunderhauf, N., and Protzel, P. (2009). A vision based onboard approach for landing and position control of an autonomous multirotor UAV in GPS-denied environments. In *2009 International Conference on Advanced Robotics (ICAR)*, 1–6.

Ledergerber, A., Hamer, M., and D'Andrea, R. (2015). A robot self-localization system using one-way ultra-wideband communication. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3131–3137.

Lupashin, S., Hehn, M., Mueller, M.W., Schoellig, A.P., Sherback, M., and D'Andrea, R. (2014). A platform for aerial robotics research and demonstration: The Flying Machine Arena. *Mechatronics*, 24(1), 41 – 54.

Moré, J.J. (1978). The Levenberg-Marquardt algorithm: Implementation and theory. In *Numerical analysis*, 105–116. Springer.

Mueller, M.W., Hamer, M., and D'Andrea, R. (2015). Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation. In *IEEE International Conference on Robotics and Automation, (ICRA)*, 1730–1736.

Nocedal, J. and Wright, S. (2006). *Numerical optimization*. Springer Science & Business Media.

Prorok, A. and Martinoli, A. (2014). Accurate indoor localization with ultra-wideband using spatial models and collaboration. *International Journal of Robotics Research*, 33(4), 547–568.

Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer.

Wenzel, K.E. and Zell, A. (2009). *Automatic take off, hovering and landing control for miniature helicopters with low-cost onboard hardware*, 73–80. Springer Berlin Heidelberg.

Wurman, P.R., D'Andrea, R., and Mountz, M. (2007). Coordinating hundreds of cooperative, autonomous vehicles in warehouses. In *2007 National Conference on Innovative Applications of Artificial Intelligence*, volume 2, 1752–1759.

Yang, S., Scherer, S.A., and Zell, A. (2013). An onboard monocular vision system for autonomous takeoff, hovering and landing of a micro aerial vehicle. *Journal of Intelligent & Robotic Systems*, 69(1), 499–515.

Ye, R., Redfield, S., and Liu, H. (2010). High-precision indoor UWB localization: Technical challenges and method. In *2010 IEEE International Conference on Ultra-Wideband*, volume 2, 1–4.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11), 1330–1334.