Robust and Efficient State Estimation for Micro Aerial Vehicles

Logan Michael Ellis

August 14, 2018
CMU-RI-TR-18-57

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania

Thesis Committee:
Nathan Michael, *Chair*
William "Red" Whittaker
Eric Westman

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

Abstract

Autonomous robots provide excellent tools for information gathering in a wide variety of domains, from environmental management to infrastructure inspection and search and rescue. Micro aerial vehicles, in particular, offer a high degree of mobility that can further their effectiveness in such environments. Deployment of aerial robots deployment in remote environments can render human operation unfeasible, necessitating resilient autonomous systems. At the core of any autonomous mobile robot is the capability to produce a consistent belief of the robot's location with respect to its environment. The efficacy and efficiency of the state estimator affects the performance of nearly all other robotic systems, from high-level motion planners to feedback controllers. This thesis examines the challenges associated with providing an accurate state estimate for MAVs operating in diverse environments.

Operation in cluttered, indoor environments precludes the use of GPS and requires laser- and vision-based odometry methods for state estimation. Additionally, the added mobility of aerial platforms comes at the expense of size, weight, and power constraints that preclude more information-dense and computationally expensive observation modalities. This thesis addresses these challenges through the development and evaluation of two modern state estimation methodologies representing both primary sensing modalities. The approach to the methodologies are driven by computational efficiency and extensibility. A novel multi-modal framework is then formulated by combining both observational models to produce a consistent and accurate state estimator that is robust to environmental diversity. All three state estimation methods are implemented on an experimental platform and evaluated through a series of flights. Quantitative analysis is provided through flights in a motion capture arena

while qualitative evaluation is provided by traversals through challenging indoor environments. These evaluations demonstrate the ability to provide consistent and accurate state estimation in real-time on constrained aerial platforms operating in diverse environments.

Acknowledgements

The work presented in this thesis would not be possible without guidance and support from my advisor, Dr. Nathan Michael. Your illuminating knowledge in technical, professional, and academic domains was invaluable to my success as a researcher.

I would like to extend gratitude towards the members of the Resilient Intelligent Systems Laboratory. The closeness within the lab borders on familial and the ever-present discussions within the office provided frequent respites from the rigors of graduate school. I would like to thank John Yao who not only provided the foundational work on the laser-based state estimation covered in this thesis, but also met all of my inquiries and curiosities with oracle-like knowledge.

I would in particular like to thank Wennie Tabib. To summarize your contribution to my academic and professional development could fill a thesis on its own. From proof-reading nearly all of my work to providing key insights into my research, you have proven instrumental to my success at CMU.

Finally, I wish to thank my parents, Evelyn and Glen Cobb. Your boundless love, patience, and support produced a stable foundation on which to stand while I reached for the stars. I credit my successes, both personal and professional, to the path that you carved in front of me and the shining example of character and integrity that you provided.

We gratefully acknowledge the support of the Department of Energy Traineeship DEEM0004067.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Autonomous robotic systems serve as exceptional tools for information gathering, particularly in environments that may be difficult or dangerous for humans to access. The increased mobility of micro aerial vehicles (MAVs) make them especially well-suited for tasks ranging from environmental management [22, 65] to infrastructure inspection [49] and search and rescue [72]. Operations in the aforementioned roles often requires navigating challenging remote environments that make human operation of the robot unfeasible. Resilient autonomy is then crucial for the safety and effectiveness of the robotic systems.

At the core of autonomous capability is state estimation, the process by which a robot incorporates observations to localizes itself with respect to its environment. All levels of autonomous operations, e.g., planning, mapping, and control, are heavily dependent upon the accuracy of the state estimator. This dependency is made even more prominent on MAVs, which require high-rate feedback control to counter dynamic instability. It is therefore essential to provide an estimate of the state that is both accurate and consistent for successful autonomous operation. The focus of this thesis is to explore methods for providing robust state estimation for MAVs operating in challenging environments.

The added mobility of MAVs, relative to their legged and wheeled counterparts, comes at the cost of size, weight, and power constraints. Such constraints limit both the quantity and quality of sensors that may be carried, as well as the onboard computation required to process the observations. To further add to the difficulty of state estimation, indoor operations, such as those shown in Fig. 1.1, preclude the use of GPS and necessitate the use of localization and odometric methods.

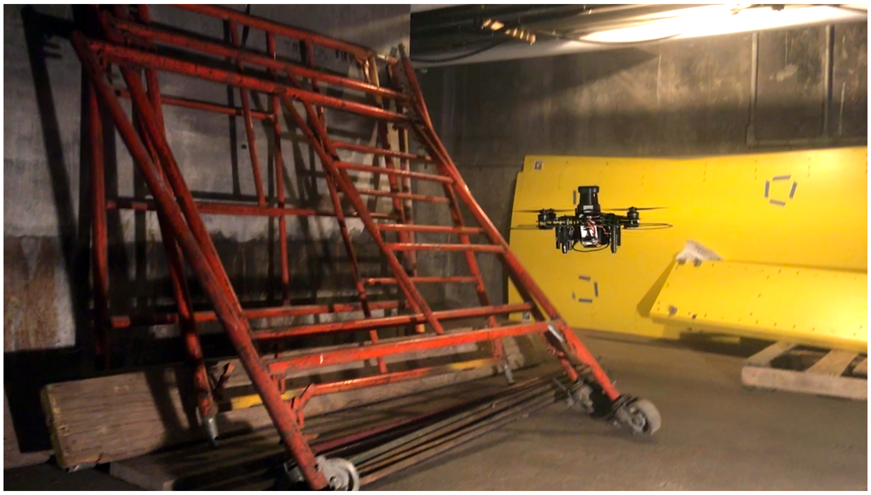

Figure 1.1: The experimental micro aerial vehicle navigating a challenging indoor environment with clutter and sub-optimal illumination.

Two primary classes of sensors are used on mobile robots: scanning laser rangefinders and camera-based vision systems. Each sensing methodology achieves acceptable performance in their optimal domains. However, both sensing methods begin to degrade when assumptions underlying their models, i.e. homogeneity in illumination or environmental structure, are violated. As MAVs and autonomous mobile robots continue to operate in increasingly challenging domains, it is necessary to develop state estimators that are both efficient enough for real-time performance and robust to environmental diversity.

1.1 Previous Work

State estimation for mobile robots fall under two primary categories: localization and odometry. Generally speaking, localization involves the coupling of observations with a known map to estimate *absolute* state [8, 13, 40], whereas odometry refers to the matching of consecutive observations to infer *relative* motion [1, 39, 53]. Fig. 1.2 provides a hierarchy of state estimation approaches.

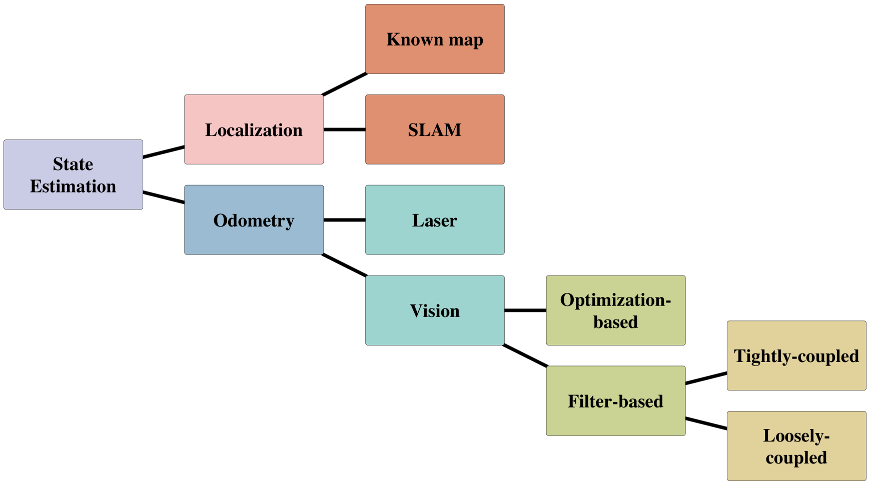

Figure 1.2: Hierarchical representation of state estimation approaches and methods.

The presence of an *a priori* map can provide a valuable resource for state estimation. With known initial conditions, Kalman Filtering methods may be applied [8], however particle filters are capable of generating estimates with arbitrary belief distributions on initial conditions or multi-modal estimates of state [35, 57, 58]. The necessity of an *a priori* map can be eliminated by performing Simulatenous Localization and Mapping (SLAM) [36, 44, 67]. Work within the field of SLAM includes a broad class of methods, ranging from vision-based [23, 28, 47, 47] to laser-based [21, 27, 44, 80]. The cornerstone work of Kaess, et al. [33] and Dellaert, et al. [14] developed a SLAM approach built around the probabilistic graphic models of factor graphs that produces impressive results. The field of SLAM research, while closely related to traditional state estimation, is an entire domain in and of itself. An excellent summary paper of the history and the future of SLAM is given in [9].

The work in this thesis primarily involves odometric [1, 6, 12, 41] methods rather than localization methods. While SLAM-based approaches are able to produce excellent and consistent estimates of vehicle state, the methods are inherently computationally expensive. Furthermore, at the core of most SLAM solutions is an odometric method of some form. Through focusing on efficient and robust state estimation methods, the benefits can propagate up to SLAM-based algorithms.

Odometric methods primarily rely on either vision- or laser-based sensors. While advances in computer vision have accelerated the use of visual odometry in recent years, the use of scanning laser rangefinders has been ubiquitous in robotics for both mapping [50, 69] and state estimation [57, 71]. State of the art methods involving scan matching of dense 3D point clouds can provide impressive results for motion inference [80], but the size of 3D laser scanners and the computational complexity of the associated algorithms are unfeasible for smaller MAV platforms. The work of Shen, et al. [62], provides an alternative method using, smaller, lighter 2D laser scanners.

1.1.1 Visual Odometry

Visual odometry (VO) methods are defined by two criteria: the metrics used for image matching and framework for estimating state from the resultant transforms. Image matching is performed through indirect or direct methods. Indirect methods [45, 48, 68] extract features of static points within the world and attempt to minimize a geometric reprojection error between images. Direct methods [3, 16, 18, 74] attempt to directly minimize the photometric error between images. Direct methods offer better accuracy and low-light performance, but tend to perform poorly on rolling shutter cameras.

With image matching metrics in place, state estimation can be performed performed through optimization-based techniques or filter-based techniques. Optimization-based methods [16, 18, 47, 74, 79] perform image matching and subsequent motion inference through iterative optimization. Such techniques are effective, but optimization-based methods require more computational resources than filter-based methods and offer little probabilistic guarantees on performance.

Filter-based methods perform state estimation by incorporating the VO estimate as the observation model in a filter (typically a Kalman Filter). Filter-based methods can then be distinguished by tightly-coupled or loosely-coupled filters. Loosely-coupled [63, 78] perform VO through external methods, then perform the Kalman Filter update using the resultant estimate. In contrast, tightly-coupled methods [7, 45, 68, 81, 83] formulate the observation update residuals directly on feature reprojection or photometric error. Such methods allow for observabililty analysis [25] that can be used to improve probabilistic correctness [26] or influence control strategies [2]. In addition to lower computational cost, filter-based methods also provide extensibility to incorporate other observation modalities.

1.2 Thesis Problem

This thesis focuses on the problem of providing an accurate and consistent state estimate for micro aerial vehicles operating challenging environments. Operations in confined environments coupled with the dynamic instability of aerial platforms mean that momentary failure or non-trivial degradation of the state estimator can result in catastrophic failure. This problem is further exacerbated in domains such as environmental management where recovery of the vehicle and subsequent continuance of operation may not be possible. The state estimator of a MAV must therefore be able to continuously provide an accurate estimate of its state despite challenging conditions.

This thesis addresses the challenges of accurate and consistent state estimation on MAVs through development of methodologies that:

- minimize size, weight, and power constraints through selection of sensors,
- are computationally tractable for real-time performance, and
- provide robustness to changing environmental structure and illumination.

1.3 Contributions and Outline

The thesis goals in 1.2 are addressed in three parts. The first is the development of a state estimator that uses an Unscented Filter with a laser-based observation model. Simplified state and environmental models are formulated to facilitate the use of smaller 2D laser scanners, which enables use on much smaller platforms. The second is the implementation of a vision-based Kalman Filter known as a Multi-state Constraint Kalman Filter (MSCKF) [45]. The MSCKF is more computationally efficient than other visual-inertial odometry (VIO) methods [68] due to tightly-coupled filter-based formulation and a null space projection to eliminate feature tracking from the state. An additional benefit of the MSCKF compared to other VIO methods is the extensibility that comes with the filtering framework. This extensibility is utilized in the third contribution: a new, novel multi-modal state estimator that combines both observations methods to provide a consistent and accurate state estimate that is robust to environmental diversity.

This thesis is organized as follows. The following section provides a brief summary of the notation used throughout this text. Chapter 2 develops an Unscented Kalman Filter that utilizes ICP-based odometry with a 2D laser scanner as the primary observation model. The simplified state and assumptions of environmental structure are lifted in Chapter 3 through implementation of the vision-based Multi-State Constraint Kalman Filter (MSCKF). Chapter 4 extends the MSCKF model to incorporate laser observation and altimeter model for a robust, multi-modal state estimator. Finally, a summary of the thesis and avenues for future research are provided in Chapter 5.

1.4 Overview of Notation

A brief word on the notation used throughout the thesis: vectors are expressed as a bold lower-case letter, $\mathbf{x}$ and matrices are expressed as bold upper-case letters, $\mathbf{X}$. Scalars are represented as unbolded lower-case letters, x. In later chapters quaternions are used to track orientation, and a quaternion that rotates a point from frame $\{B\}$ to frame $\{A\}$ is expressed as ${}^{A}\mathbf{q}_{B}$. Rotation matrices are represented as ${}^{A}\mathbf{R}_{B}$ and, unless otherwise noted, are assumed to be functions of the underlying quaternion corresponding to the same rotation. Positions are notated with a bold $\mathbf{p}$, and the position that describes the location of $\{B\}$ as expressed in the coordinates of $\{A\}$ is denoted as ${}^{A}\mathbf{p}_{B}$. State terms denoted by a hat $\hat{\mathbf{p}}$ represent estimates of the state. Finally, for all filter formulations, the body frame is assumed to be coincident with the IMU frame, therefore the terms *IMU frame* and *body frame* will be used interchangeably.

Chapter 2

Laser-based State Estimation With Unscented Kalman Filter

This chapter develops a loosely-coupled Unscented Kalman Filter that incorporates laser scan-matching as the primary sensing modality. A brief overview of Unscented Kalman Filtering is provided for reference in Section 2.1 and the state and process models are detailed in Section 2.2. In Section 2.3, the laser odometry observation model is formulated. The use of *relative* observations requires prior poses be incorporated into the state, which is described in Section 2.4. The state estimator is deployed on an aerial vehicle to evaluate real-time performance and estimator accuracy. The results are detailed in Section 2.5. The state estimator achieves accurate performance in structured environments, but performance suffers in 3D-rich environments.

2.1 Overview of Unscented Kalman Filter

The Unscented Kalman Filter (UKF) [32] is a non-linear Kalman Filter that utilizes state propagation techniques similar to a particle filter. The UKF linearizes process and observation models through propagation of sigma points. In simpler terms, it samples $2n + 1$ points (where n is the dimensionality of the state space) about the state mean and passes each point through the non-linear model. A new Gaussian distribution is then fit to the propagated points. The Unscented Kalman Filter is detailed in Alg. 1 [70].

Algorithm 1 Unscented Kalman Filter

```
 1: procedure UNSCENTED_KALMAN_FILTER($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$)
 2:     $\mathcal{X}_{t-1} = \left(\mu_{t-1} \quad \mu_{t-1} + \gamma\sqrt{\Sigma_{t-1}} \quad \mu_{t-1} - \gamma\sqrt{\Sigma_{t-1}}\right)$
 3:     $\bar{\mathcal{X}}_t = g(u_t, \mathcal{X}_{t-1})$
 4:     $\bar{\mu}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{X}}_t^{*[i]}$
 5:     $\bar{\Sigma}_t = \sum_{i=0}^{2n} (\bar{\mathcal{X}}_t^{*[i]} - \bar{\mu}_t)(\bar{\mathcal{X}}_t^{*[i]} - \bar{\mu}_t)^\top + R_t$
 6:     $\bar{\mathcal{X}}_t = \left(\bar{\mu}_t \quad \bar{\mu}_t + \gamma\sqrt{\bar{\Sigma}_t} \quad \bar{\mu}_t - \gamma\sqrt{\bar{\Sigma}_t}\right)$
 7:     $\bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$
 8:     $\hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$
 9:     $\bar{S}_t = \sum_{i=0}^{2n} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^\top + Q_t$
10:     $\bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^\top$
11:     $K_t = \bar{\Sigma}_t^{x,z} S_t^{-1}$
12:     $\mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$
13:     $\Sigma_t = \bar{\Sigma}_t - K_t S_t K_t^\top$
14:     return $\mu_t, \Sigma_t$
15: end procedure
```

Lines 1–5 of the algorithm represent the process update. Sigma points are generated in line 1, then passed through the non-linear function. The mean and covariance of the resultant points are formed in lines 4–5. In lines 6–12, new sigma points are generated from the process prior and passed through the observation update. It is noted that sigma points, rather than Jacobians as used in the Extended Kalman Filter, are used to propagate system uncertainty. The added filter complexity is offset by being derivative-free and offering superior performance in modeling uncertainty in non-linear systems [77].

2.2 State Representation and Process Model

Typical state representations for MAVs consist of vehicle position and full orientation through quaternions [73] or manifold representations with Lie Groups [5], however, the 2D planar scanner necessitates assumptions of the environment that limit observability of orientation. The assumptions are discussed below in Section 2.3. The main state is reduced to include only yaw rotation, as well as vehicle position and velocity. The state of the vehicle is centered at the IMU $\{I\}$ frame and is expressed with relationship to the world frame $\{G\}$.

With the coordinate frames defined and state assumption in place, the state $\mathbf{x} \in \mathbb{R}^{11}$ is defined as:

$$\mathbf{x} = \begin{bmatrix} {}^G\mathbf{p}_I & \psi_I & {}^G\mathbf{v}_I & b_{d\psi} & \mathbf{b}_a \end{bmatrix}^\top \tag{2.1}$$

where ${}^G\mathbf{p}_I = [p_x, p_y, p_z]$ is the position of the IMU frame in the world frame, ψ_I represents yaw about the gravity vector axis in the world frame, ${}^G\mathbf{v}_I = [v_x, v_y, v_z]$ is the velocity of the body frame and $\{b_{d\psi}, \mathbf{b}_a\}$ represent IMU biases for yaw and body-frame acceleration respectively.

Vehicle orientation is still required to transform body-frame accelerations into the world frame, and thus pitch and roll observations from the IMU are used as inputs to the system. The full input $\mathbf{u} \in \mathbb{R}^8$ is given as:

$$\mathbf{u} = \begin{bmatrix} \phi_m & \theta_m & \mathbf{a}_m^\top & \omega_m^\top \end{bmatrix}^\top \tag{2.2}$$

where $[\phi_m \quad \theta_m]$ are pitch and roll, $\mathbf{a}_m$ is body-frame acceleration and ω is the body-frame rotational velocity, all as measured by the IMU. The system dynamics are defined through a continuous-time model:

$$\begin{aligned} \dot{\mathbf{p}}_I &= {}^G\mathbf{v}_I \\ \dot{\psi}_I &= \frac{1}{\cos(\theta_m)}\left(\omega_y \sin(\psi_m) + \omega_z \cos(\psi_m)\right) - b_{d\psi} \\ {}^G\dot{\mathbf{v}}_I &= \hat{\mathbf{a}} - \mathbf{n}_a + {}^G\mathbf{g} \end{aligned} \tag{2.3}$$

where $\hat{\mathbf{a}} = \mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a$ is the IMU-measured body frame accelerations with noise and bias removed. IMU input noises $\mathbf{n}_I \in \mathbb{R}^7$ for acceleration, yaw bias, and accelerometer biases are given by:

$$\mathbf{n}_I = \begin{bmatrix} \mathbf{n}_a^\top & n_{bd\psi} & \mathbf{n}_{ba}^\top \end{bmatrix}^\top \tag{2.4}$$

The dimensionality of the state and noise input differ, therefore a mapping between noise and state values is required:

$$\mathbf{Q}_k = \mathbf{G}_Q \mathbf{Q} \mathbf{G}_Q^\top \tag{2.5}$$

where $\mathbf{Q}_k$ represents the process noise. The matrix $\mathbf{Q}$ is a diagonal matrix comprised of (2.4):

$$\mathbf{Q} = \begin{bmatrix} \sigma_a^2 \mathbf{I}_3 & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{1\times3} & \sigma_{bd\psi}^2 & \mathbf{0}_{1\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \sigma_{ba}^2 \mathbf{I}_3 \end{bmatrix} \tag{2.6}$$

The matrix $\mathbf{G}_Q$ is the mapping from IMU noise to process model noise determined by differentiating the continues-time dynamics with respect to input noise:

$$\mathbf{G}_Q = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{1\times3} & 1 & \mathbf{0}_{1\times3} \\ -\mathbf{I}_3 & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times1} & \mathbf{I}_3 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{I}_3 \end{bmatrix} \tag{2.7}$$

The state mean is propagated through the non-linear process dynamics of (**??**) using a first order integration for model simplicity. Covariance is propagated according to Alg. 1.

2.3 Observation Model

The primary sensing modality used for the filter is a planar scanning laser rangefinder. Dense 3D scanners can produce highly accurate estimates of state in 3D-rich environments [80] and 2D laser scanners can be used to localize an aerial vehicle in a known map [8]. However, operation in unknown environments necessitates necessitates assumptions about environmental structure. The state estimator assumes a 2.5D environment: the vehicle operates at a near-level orientation in environments with walls being purely vertical. If, for example, the robot experiences a vertical surfaced tilted away or towards the vehicle, a planar laser scanner will be unable to determine if the vehicle has moved or if the wall is simply angled. For most man-made environments, this assumption holds. A block diagram of the developed system is shown in Fig. 2.1.

The assumption of vertical walls allows for motion inference through matching of

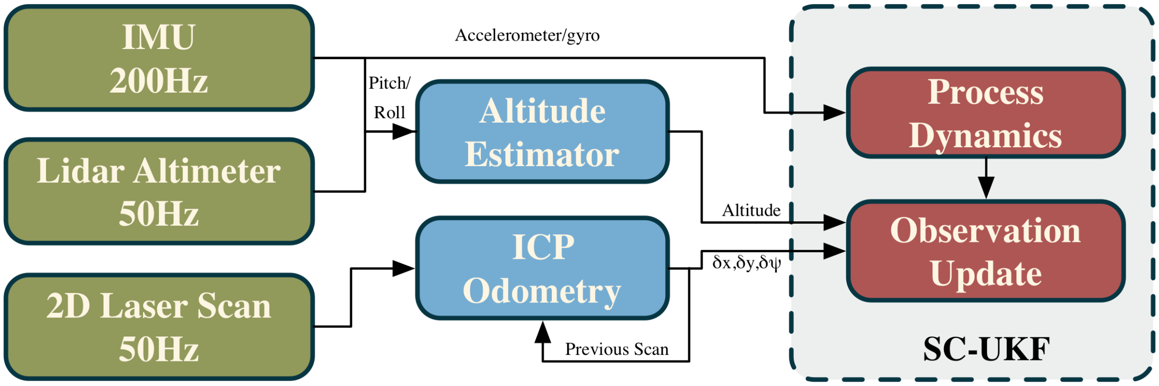

Figure 2.1: System Diagram of SC-UKF State Estimator

consecutive laser scans through the Iterative Closest Point (ICP) algorithm. ICP-based odometry, discussed in the following section, serves as the primary observation method for the Laser-based Unscented Kalman Filter. ICP odometry produces position and yaw estimates while altitude observations are provided by a downward-facing lidar rangefinder.

2.3.1 Odometry Through Iterartive Closest Point

Iterative Closest Point (ICP) is a ubiquitous algorithm in robotics that seeks to determine the optimal transformation to match two point clouds. The objective function is given by:

$$\mathbf{T}^* = \underset{\mathbf{T}}{\operatorname{argmin}} \sum_{i=1}^{N} \|\mathbf{T}\mathbf{p}_i - \mathbf{q}_i\|_2^2 \tag{2.8}$$

If the point clouds are noiseless and contain identical points (with known correspondences), the solution can be found in closed form. In practice, laser scans contain noisy and temporally heterogeneous observations, necessitating iterative optimization to solve (2.8). Algorithmic details, such as determination of point correspondences, are outside the scope of this thesis.For more information, the reader is directed to the excellent summary paper by Pomerleau, et al. [52].

A summary of the ICP odometry algorithm used in the laser-based UKF is shown in Alg. 2. The algorithm begins by projecting the body-frame laser scan onto the ground plane using the current state. The point cloud is processed with features such as random sampling of points and outlier rejection to improve robustness and decrease computational load. The projected and processed scan cloud is matched to the previous scan using the libpointmatcher library [52].

Algorithm 2

```
procedure ICP Odometry($\mathbf{T}_0, \mathbf{P}_0, \mathbf{Q}, {}^L\mathbf{R}_G,$)
    $\mathbf{T} \leftarrow \mathbf{T}_{est}$
    $\mathbf{P} \leftarrow$ ProjectScanToPlane($\mathbf{P}_0, {}^L\mathbf{R}_G$)
    $\mathbf{P} \leftarrow$ ProcessAndFilterScan($\mathbf{P}$)
    while not converged do
        for $i \leftarrow 1 : N$ do
            $\mathbf{q}_i \leftarrow$ FindClosestPointInQ($\mathbf{T} \cdot \mathbf{p}_i$)
            $\mathbf{n}_i \leftarrow$ GetNormalAtPoint($\mathbf{q}_i$)
        end for
        $\mathbf{T} \leftarrow \underset{\mathbf{T}}{\operatorname{argmin}} \sum_{i=1}^{N} \|\mathbf{n}_i \cdot (\mathbf{T}\mathbf{p}_i - \mathbf{q}_i)\|_2^2$
    end while
    $\mathbf{Q} \leftarrow \mathbf{P}$
    $\{\delta_x, \delta_y, \delta_\psi\} \leftarrow \mathbf{T}$
    return $\delta_x, \delta_y, \delta_\psi$
end procedure
```

Rather than using point-to-point matching as in (2.8), the point-to-plane objective function is used:

$$\mathbf{T}^* = \underset{\mathbf{T}}{\operatorname{argmin}} \sum_{i=1}^{N} \|\mathbf{n}_i \cdot (\mathbf{T}\mathbf{p}_i - \mathbf{q}_i)\|_2^2 \tag{2.9}$$

where $\mathbf{n}_i$ is the surface normal at $\mathbf{q}_i$. The points-to-plane formulation leverages assumptions of the local planarity of the surfaces being observed by the laser scanner to improve the robustness of matching [11]. More advanced methods of scan matching, such as plane-to-plane matching with Generalized ICP [60] and Normal ICP [61], have been shown to further improve scan matching performance. Given the assets available from the libpointmatcher library and the simpler structure of 2D scans, the point-to-plane implementation is used.

The resultant transformation from the ICP algorithm is an estimate of the *relative* motion between scans, given by $\mathbf{z} = [\delta_x, \delta_y, \delta_\psi]$. However, the state of the vehicle consists of global position and yaw estimates. To process these *relative* observations during the correction update, the filter must properly account for the uncertainty of prior poses. The process of augmenting the state with prior poses, referred to as Stochastic Cloning, is detailed in Section 2.4.

2.3.2 Altitude Observation

A downward-facing lidar rangefinder is used to produce *absolute* observations of the position of the vehicle along the z axis of the world frame. Observation updates of a Kalman Filter typically project the vehicle state into the observation space to form a residual between estimated observations and true observations. For the downward-facing lidar, this would consist of vehicle position and orientation. The state described in (2.1) only incorporates yaw observations so a simplified altitude model is used. The altitude observation from the rangefinder is rotated into the world frame using IMU-observed pitch and roll. The residual is then given by:

$$z = \mathbf{c}{}^{I}\hat{\mathbf{R}}_G^\top \mathbf{z}_{obs} - \hat{p}_z \tag{2.10}$$

with a slight abuse of notation, $\mathbf{z}_{obs} \in \mathbb{R}^3$ represents a vector with the observed altimeter observation expressed in the z axis, ${}^{I}\hat{\mathbf{R}}_G^\top$ is the rotation matrix formed by IMU-observed pitch and roll, and $\mathbf{c}$ represents the selection vector:

$$\mathbf{c} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \tag{2.11}$$

A fixed covariance value σ_{alt}^2 is used for the lidar altimeter. The observation update then proceeds as normal.

2.4 State Augmentation With Stochastic Cloning

In Section 2.3, the concept of Stochastic Cloning was briefly introduced. A complication in using odometric observations with a state representation that consists of global position and orientation is that the odometric methods only provide *relative* observations. That is, odometry observes the motion of vehicle between poses. Given that both poses are distributed with some non-zero uncertainty, deterministically integrating the observations is not probabilistically accurate. To incorporate the *relative* observations, and to properly account for the state uncertainty, the state is augmented with a clone of the pose at each scan time.

Cloning for Extended Kalman Filters is fairly straightforward [46, 55] and is covered in Chapter3. The lack of Jacobians in Unscented Kalman Filters make uncertainty propagation of augmented states during the observation update more complex. The following section summarizes the work of Shen, et. al. [63] in formulating stochastic cloning for the UKF, hereupon referred to as SC-UKF.

Consider a state vector given by $\mathbf{x} \sim \mathcal{N}(\hat{\mathbf{x}}, \mathbf{P}^{\mathbf{xx}}) \in \mathbb{R}^n$. The UKF generates sigma points and propagates them through a nonlinear function:

$$\mathcal{Y}_i = g(\mathcal{X}_i) \tag{2.12}$$

The mean and covariance of the propagated points, and cross-covariance between $\mathbf{x}$

and $\mathbf{y}$ are defined as:

$$\begin{aligned}\hat{\mathbf{y}} &= \sum_{i=0}^{2n} w_i^m \mathcal{Y}_i \\ \mathbf{P}^{\mathbf{yy}} &= \sum_{i=1}^{2n} w_i^c (\mathcal{Y}_i - \hat{\mathbf{y}})(\mathcal{Y}_i - \hat{\mathbf{y}})^\top \\ \mathbf{P}^{\mathbf{yx}} &= \sum_{i=1}^{2n} w_i^c (\mathcal{Y}_i - \hat{\mathbf{y}})(\mathcal{X}_i - \hat{\mathbf{x}})^\top \end{aligned} \tag{2.13}$$

Given I augmented states, the full state and covariance are represented as:

$$\begin{aligned}\breve{\mathbf{x}} &= \begin{bmatrix} \hat{\mathbf{x}} & \hat{\mathbf{x}}_1 & ... & \hat{\mathbf{x}}_I \end{bmatrix}^\top \\ \breve{\mathbf{P}} &= \begin{bmatrix} \mathbf{P}^{\mathbf{xx}} & \mathbf{P}^{\mathbf{xx_1}} & \cdots & \mathbf{P}^{\mathbf{xx_I}} \\ \mathbf{P}^{\mathbf{x_1x}} & \mathbf{P}^{\mathbf{x_1x_1}} & \cdots & \mathbf{P}^{\mathbf{x_1x_1}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}^{\mathbf{x_Ix}} & \mathbf{P}^{\mathbf{x_Ix_1}} & \cdots & \mathbf{P}^{\mathbf{x_Ix_I}} \end{bmatrix}. \end{aligned} \tag{2.14}$$

Consider a binary matrix $\mathbf{B}$ that selects the appropriate portions of the state to be augmented. State clones may be added and removed by:

$$\begin{aligned}\breve{\mathbf{x}}^+ = \mathbf{M}^+\breve{\mathbf{x}}, \quad \mathbf{M}^+ &= \begin{bmatrix} \mathbf{I}_{n+\sum_I n_i} \\ \mathbf{B_{I+1}} \end{bmatrix} \\ \breve{\mathbf{x}}^- = \mathbf{M} - +\breve{\mathbf{x}}, \quad \mathbf{M}^- &= \begin{bmatrix} \mathbf{I}_a & \mathbf{0}_{a\times n_j} & \mathbf{0}_{a\times b} \\ \mathbf{0}_{b\times n} & \mathbf{0}_{b\times n_j} & \mathbf{I}_b \end{bmatrix} \end{aligned} \tag{2.15}$$

where $a = n + \sum_{i=1}^{j-1} n_i$ and b=$\sum_{i=j+1}^{I} n_i$. The augmented state covariance is given by:

$$\breve{\mathbf{P}}^\pm = \mathbf{M}^\pm \breve{\mathbf{P}} \mathbf{M}^{\pm\top}. \tag{2.16}$$

The work of [37] detailed that the empirical approximation of a state linearization may be found by solving the objective function:

$$\min_{\mathbf{A},\mathbf{b}} \sum_{i=0}^{2n} w_i(\mathcal{Y}_i - \mathbf{A}\mathcal{X}_i - \mathbf{b})(\mathcal{Y}_i - \mathbf{A}\mathcal{X}_i - \mathbf{b})^\top \tag{2.17}$$

The optimal linear regression is given by:

$$\mathbf{A} = \mathbf{P^{yx}}\mathbf{P^{xx}}^{\mathbf{-1}}, \quad \mathbf{b} = \hat{\mathbf{y}} - \mathbf{A}\hat{\mathbf{x}} \tag{2.18}$$

The $\mathbf{A}$ matrix allows for propagation of covariance of a fully augmented state, similar to a Jacobian in an EKF. This linear regression matrix is used in both the process update and observation update.

2.4.1 Process Update

For the SC-UKF process update, the augmented and main states of (2.14) can be partitioned:

$$\breve{\mathbf{x}}_{t|t} = \begin{bmatrix} \hat{\mathbf{x}}_{t|t} \\ \hat{\mathbf{x}}_{I_{t|t}} \end{bmatrix}, \quad \breve{\mathbf{P}}_{t|t} = \begin{bmatrix} \mathbf{P}^{\mathbf{xx}}_{t|t} & \mathbf{P}^{\mathbf{xx_I}}_{t|t} \\ \mathbf{P}^{\mathbf{x_Ix}}_{t|t} & \mathbf{P}^{\mathbf{x_Ix_I}}_{t|t} \end{bmatrix} \tag{2.19}$$

A linear approximation to the nonlinear state dynamics can be written as:

$$\breve{\mathbf{x}}_{t+1|t} = f(\breve{\mathbf{x}}_{t|t}, \mathbf{u}_t, \mathbf{v_t}) = \begin{bmatrix} \mathbf{F}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{|I|} \end{bmatrix} \breve{\mathbf{x}}_{t|t} + \begin{bmatrix} \mathbf{J}_t & \mathbf{G}_t \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}_t \\ \mathbf{v}_t \end{bmatrix} + \mathbf{b}_t + \mathbf{e}_t \tag{2.20}$$

Noting the process update only affects the main state, sigma point propagation of the main state can be carried out in standard UKF fashion. Only the linearized process update $\mathbf{F}_t$ is needed to propagate the cross-variance terms in (2.19). To determine the linearized state dynamics using (2.18), the main state is first augmented with non-additive process noise and sigma points are generated from:

$$\bar{\mathbf{x}}_{t|t} = \begin{bmatrix} \breve{\mathbf{x}}_{t|t} \\ \mathbf{0} \end{bmatrix}, \quad \bar{\mathbf{P}}_{t|t} = \begin{bmatrix} \mathbf{P}^{\mathbf{xx}}_{t|t} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_t \end{bmatrix} \tag{2.21}$$

Sigma point propagation of the state produces both $\hat{\mathbf{x}}_{t+1|t}$ and $\mathbf{P^{xx}}_{t+1|t}$. An additional cross-covariance between prior and propagated states $\mathbf{P^{x\bar{x}}}_{t+1|t}$ can be computed. Following (2.18):

$$\mathbf{P}^{\mathbf{x\bar{x}}}_{t+1|t}\bar{\mathbf{P}}^{-1}_{t|t} = \begin{bmatrix} \mathbf{F}_t & \mathbf{G}_t \end{bmatrix} \tag{2.22}$$

The full state is updated using the resultant process Jacobian $\mathbf{F}_t$:

$$\bar{\mathbf{x}}_{t+1|t} = \begin{bmatrix} \breve{\mathbf{x}}_{t+1|t} \\ \hat{\mathbf{x}}_{I_{t|t}} \end{bmatrix}, \quad \breve{\mathbf{P}}_{t|t} = \begin{bmatrix} \mathbf{P}^{\mathbf{xx}}_{t+1|t} & \mathbf{F}_t\mathbf{P}^{\mathbf{xx_I}}_{t|t} \\ \mathbf{P}^{\mathbf{x_Ix}}_{t|t}\mathbf{F}_t^\top & \mathbf{P}^{\mathbf{x_Ix_I}}_{t|t} \end{bmatrix}. \tag{2.23}$$

2.4.2 Observation Update

Consider the state with m propagations between observations, with $\breve{\mathbf{x}}_{t+1|t}$ and $\breve{\mathbf{P}}_{t+1|t}$ being the latest mean and covariance. The estimated observation and linearized observation update for an ICP odometry observation that depends on the j^{th} augmented are given by:

$$\begin{aligned} \hat{\mathbf{z}}_{t+m|t} &= h_r(\hat{\mathbf{x}}_{t+m|t}, \mathbf{B}_j^\top, \mathbf{n}_{t+m}) \\ &= \mathbf{H}_{t+m|t}\breve{\mathbf{x}}_{t+m|t} + \mathbf{L}_{t+m}\mathbf{n}_{t+m} + \mathbf{b}_{t+m} + \mathbf{e}_{t+m} \qquad (2.24) \\ \mathbf{H}_{t+m|t} &= \begin{bmatrix} \mathbf{H}^{\mathbf{x}}_{t+m|t} & \mathbf{0} & \mathbf{H}^{\mathbf{x_j}}_{t+m|t} & \mathbf{0} \end{bmatrix} \end{aligned}$$

(2.25)

A new augmented state is formed using only the main state and involved j^{th} state:

$$\acute{\mathbf{x}}_{t+1|t} = \begin{bmatrix} \hat{\mathbf{x}}_{t+m|t} & \\ \hat{\mathbf{x}}_{j_{t+m|t}} & \mathbf{0} \end{bmatrix}, \quad \acute{\mathbf{P}}_{t|t} = \begin{bmatrix} \mathbf{P}^{\mathbf{xx}}_{t+m|t} & \mathbf{P}^{\mathbf{xx_j}}_{t+m|t} & \mathbf{0} \\ \mathbf{P}^{\mathbf{x_jx}}_{t+m|t} & \mathbf{P}^{\mathbf{x_jx_j}}_{t+m|t} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_{t+m} \end{bmatrix} \tag{2.26}$$

Note that $\acute{\mathbf{P}}_{t|t}$ may lose positive definiteness if consecutive observation updates are performed without a process update. Assuming a process update has been performed, it is safe to proceed with sigma point propagation for the observation update. The observation estimate $\hat{\mathbf{z}}_{t+m|t}$, observation covariance $\mathbf{P^{zz}}_{t+1|t}$, and observation-state cross-covariance $\mathbf{P}^{\mathbf{z}\acute{\mathbf{x}}}{}_{t+1|t}$ are obtained. The pseudo-Jacobian for the observation update is found:

$$\mathbf{P}^{\mathbf{z}\acute{\mathbf{x}}}_{t+m|t}\acute{\mathbf{P}}^{\mathbf{-1}}_{t+m|t} = \begin{bmatrix} \mathbf{H}^{\mathbf{x}}_{t+m|t} & \mathbf{H}^{\mathbf{x_j}}_{t+m|t} & \mathbf{L}_{t+m} \end{bmatrix}. \tag{2.27}$$

The observation update is then applied in a similar fashion to an EKF:

$$\begin{aligned} \breve{\mathbf{K}}_{t+m} &= \breve{\mathbf{P}}_{t+m|t}\mathbf{H}^{\top}_{t+m|t}\mathbf{P}^{\mathbf{xx}^{-1}}_{t+m|t} \\ \breve{\mathbf{x}}_{t+m|t+m} &= \breve{\mathbf{x}}_{t+m|t} + \breve{\mathbf{K}}_{t+m}(\mathbf{z}_{t+m} - \hat{\mathbf{z}}_{t+m|t}) \\ \breve{\mathbf{P}}_{t+m|t+m} &= \breve{\mathbf{P}}_{t+m|t} - \breve{\mathbf{K}}_{t+m}\mathbf{H}_{t+m|t}\breve{\mathbf{P}}_{t+m|t} \end{aligned} \tag{2.28}$$

2.5 Results

Performance of the laser-based SC-UKF (and all subsequent state estimation methods presented in this thesis) was evaluated through three trials in a motion capture arena that were designed to mimic varying flight conditions. The first of the three trials is a near-hover flight with environmental clutter. The second trial, shown in Fig. 2.2, was performed with the curtains of the motion capture arena drawn and clutter removed to produce a highly-structured environment. Finally, a third flight involves significant clutter and sloped vertical surfaces that violate the environmental assumptions made by the ICP-based observation model. The highly accurate motion capture estimate is regarded as ground truth for the experiments.

The vehicle is controlled using non-linear model predictive control (MPC) with state estimation provided by the Vicon Motion Capture arena for the near-hover flight. Subsequent flights in cluttered and structured environments were performed through teleoperation with a PID controller due to performance issues using the non-linear MPC. For all three trials, the proposed state estimator was not used within the control loop but was rather processed in parallel to demonstrate state estimator performance without the technical challenges involved in integrating state estimation into closed-loop control. While real-time performance is assessed and verified, most of the results were generated by processing sensor data offline using ROS bagfiles on a MacBook Pro with a quad-core Intel Core i7 Processor running at 2.7GHz. The platform is detailed in the following section.

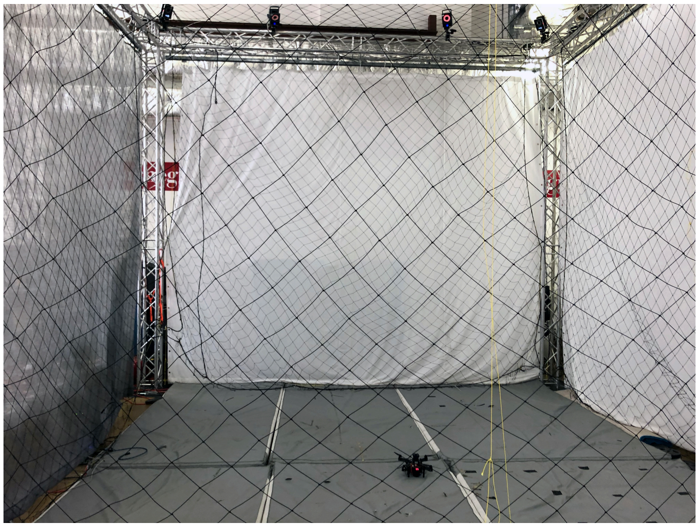

Figure 2.2: Motion capture arena in the Gates Highbay at Carnegie Mellon University

2.5.1 Experimental Platform

The vehicle used for all experimental testing throughout this thesis is shown in Fig. 2.3. The platform is a custom-built quadrotor. The fully-loaded flight mass without battery is 1.5kg and the flight time is approximately 8 minutes. Onboard computation is provided through an Intel Compute Stick CS525 with 4GB of RAM and a dual-core Intel Core m5 proccesor running at 2.80GHz. The compute stick is running Ubuntu 16.04 and ROS Kinetic. Fixed atop the vehicle is a Hokuyo UTM-30LX-EW scanning laser rangefinder. The RGB Camera is an ELP USB130W01MT-L21, which is a rolling shutter camera collecting 640 × 480 images at 30Hz. The IMU is a PX4 PixRacer broadcasting updates at 200MHz. The downward-facing lidar beam rangefinder is a Lightware SF20 operating at 50Hz. Camera intrinsics and camera-to-IMU extrensic calibrations were performed using the Kalibr toolbox [20].

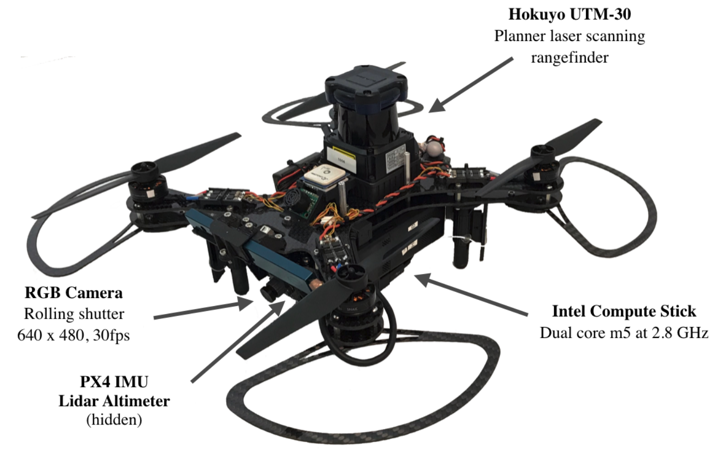

Figure 2.3: Micro Aerial Vehicle used for flight testing. Platform includes onboard compute stick, 2D Laser Scanner, RGB Camera, IMU, and Lidar altimeter

2.5.2 Motion Capture Results

The first flight conducted in the motion capture arena involved near-hover flight with mild translational motion and negligible yaw. The test was designed to evaluate baseline performance of the laser-based SC-UKF at hover conditions. The resultant position estimates are shown in Fig. 2.4 with ground truth is provided for comparison. The position error with respect to ground truth for the near-hover flight is shown in Fig. 2.5.

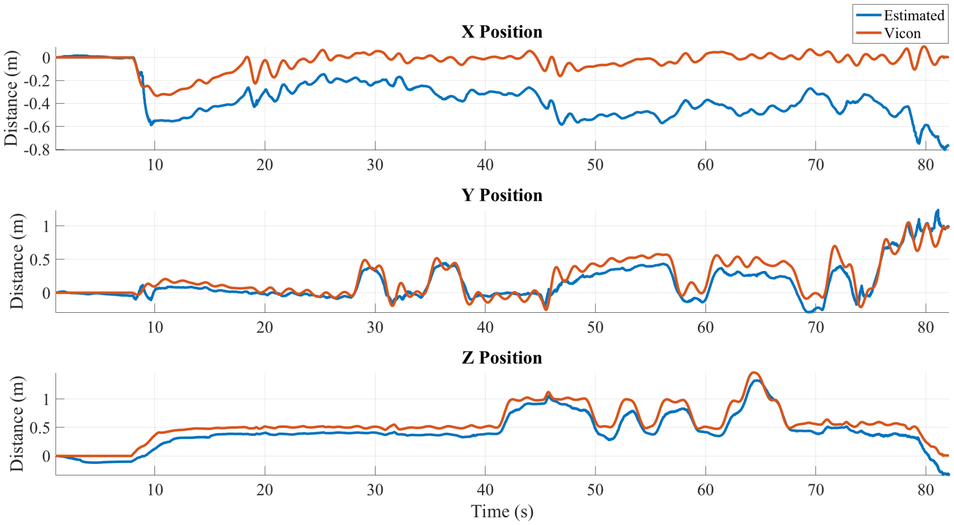

Figure 2.4: Estimated position produced by Laser UKF with respect to ground truth for flight with minimal translation and rotation.

The second flight wass designed to mimic highly-structured environments. As is visible in Fig 2.2, the curtains of the motion capture arena were drawn to provide consistent, homogeneous vertical surfaces. The resultant position with respect to ground truth is shown in Fig. 2.6. The error of the 3D position with respect to ground truth can be seen in Fig. 2.7.

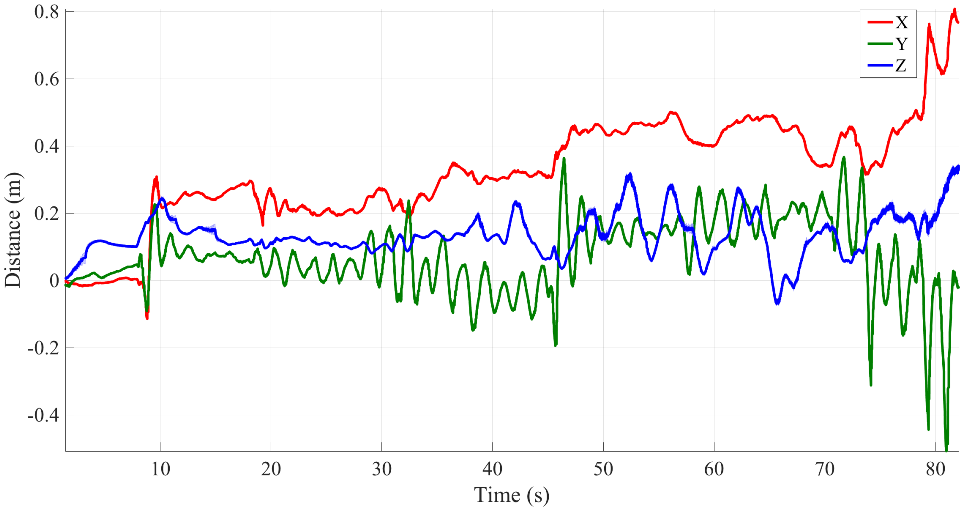

Figure 2.5: Position error with respect to ground truth and corresponding estimator variance during flight with minimal translation and rotation.

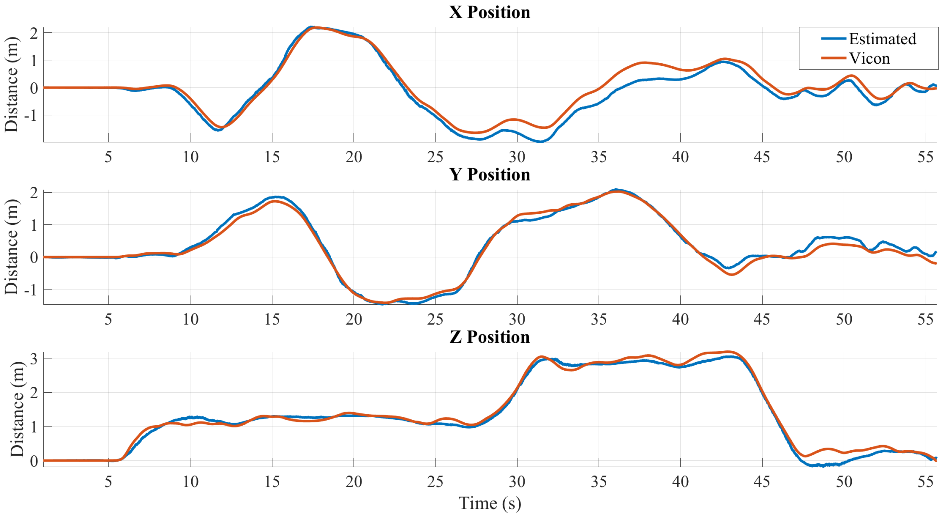

Figure 2.6: Estimated position produced by Laser UKF with respect to ground truth in an environment with vertical walls.

The final flight was designed to mimic cluttered, 3D rich environments. The motion capture arena was strewn with large boxes and several planes placed at varying angles. The test was meant to challenge the structural assumptions made by the SC-UKF. The resultant position with respect to ground truth is shown in Fig. 2.8. The

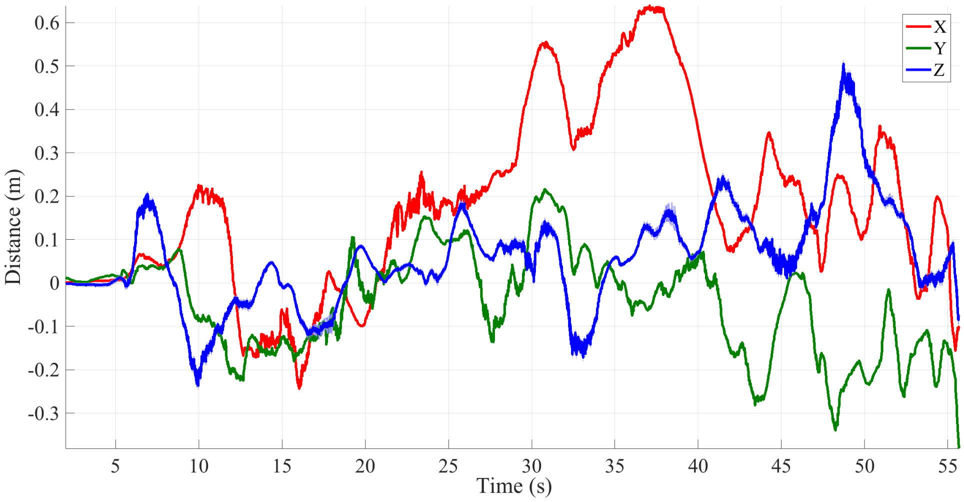

Figure 2.7: Position error with respect to ground truth and corresponding estimator variance during flight in highly structured environment.

position error with respect to ground truth can be seen in Fig. 2.9.

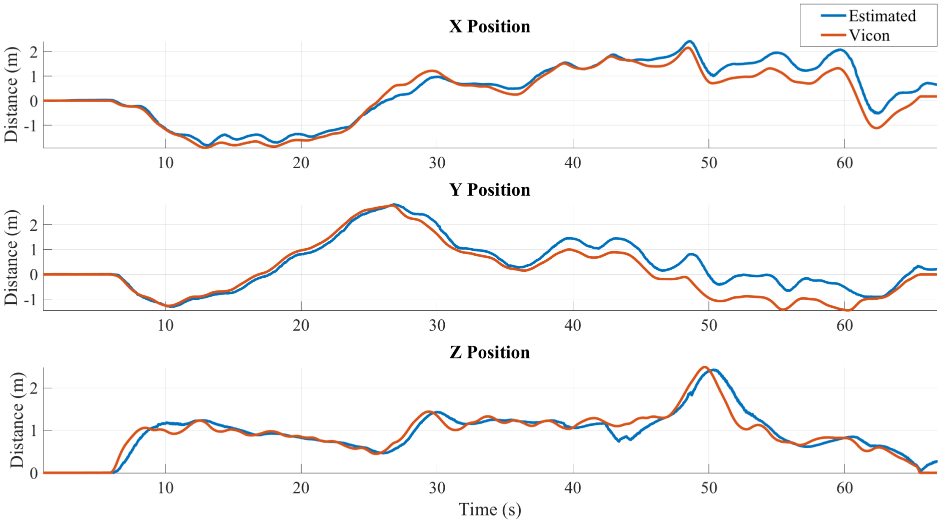

Figure 2.8: Estimated position produced by Laser UKF with respect to ground truth in an unstructured environment

These trials demonstrate the strengths and weaknesses of laser-based state estimation. Flights in highly-structured environments that adhere to the assumptions

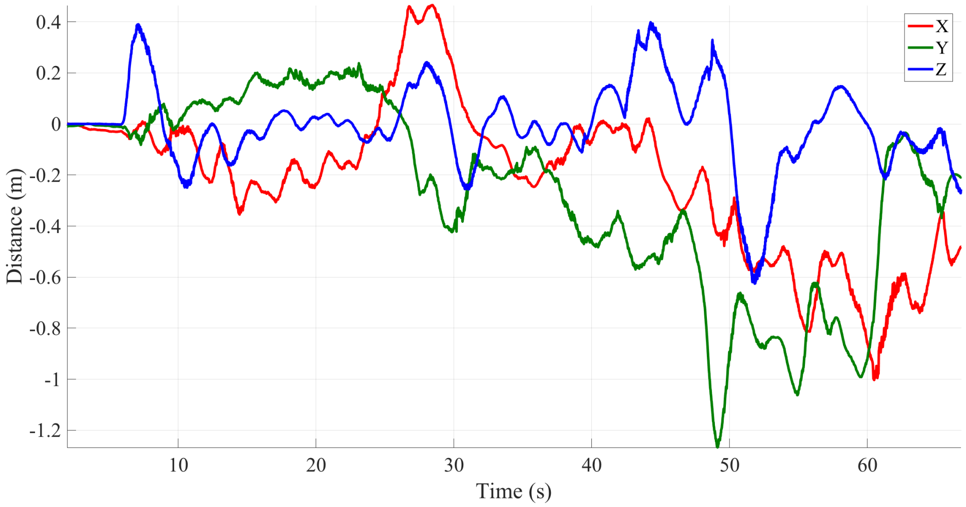

Figure 2.9: Position error with respect to ground truth and corresponding estimator variance during flight in unstructured environment

of the observation achieved far superior performance relative to the other flights. The RMSE nearly doubled between the structured and unstructured flights. The clutter in the near-hover trial hindered performance, despite little translational motion. Surprisingly, the state estimator still achieved respectable performance in the unstructured environments, demonstrating the robustness of the formulation.

2.6 Chapter Summary

Chapter 2 developed a computationally efficient state estimator with a planar scanner as the core sensing modality. Estimates of vehicle motion were produced from ICP scan matching and fused with lidar altimeter observations using an Unscented Kalman Filter (UKF).

The 2D scanning laser rangefinder offers accurate observations in a form factor sufficient for use on MAVs. The lack of rich 3D information from the scanner precludes the use of more robust scan-matching methods. A strong assumption was made that the vehicle will be operating near-hover attitude in environments with purely vertical walls. To account for the lack of observability over vehicle pitch and roll, a simplified state model was used, based primarily around global position and yaw. The assumptions of environmental structure allowed the use of ICP scan matching to provide estimates of vehicle motion which is outlined in Section 3.2. A lidar altimeter provided *absolute* observations, however the *relative* observations from the ICP odometry complicate the observation update of the UKF.

Section 2.4 addressed the issue of *relative* observations through use of Stochastic Cloning. Stochastic Cloning augments the vehicle state with prior poses to properly account for uncertainty of prior states when performing *relative* observation updates. Stochastic Cloning is common and fairly straightforward in EKFs but is made more difficult in UKFs due to sigma point propagation replacing Jacobians. The formulation empirical pseudo-Jacobion was detailed using a technique based on a Linear Regression Kalman Filter. The pseudo-Jacobian allowed for augmented cross-covariance portions of the state covariance matrix to be updated during process and observation updates.

The state estimator was implemented and deployed on to a MAV for evaluation. The specifics of the vehicle and experimental results were detailed in Section 2.5. The performance of the state estimator was quantitatively assessed through flights in a motion capture arena. The vehicle was able provide a consistent and accurate estimate of its state through various environments. As expected, state estimator performance is reduced in cluttered environments where assumptions of environmental structure are violated. The impact of environmental assumptions are softened in Chapter 3 with the development of a vision-based state estimator and are reduced further through a multi-modal implementation in Chapter 4.

Chapter 3

Visual-Inertial Odometry with Multi-State Constrained Kalman Filter

The previous chapter detailed the development of a laser-based UKF state estimator. The estimator was able to produce accurate estimates of the state but the results highlighted shortcomings of the strong assumptions of environmental structure. Visual odometry offers an attractive alternative that makes no such assumptions. In this chapter, a state of the art implementation of a tightly-coupled, filter-based visual odometry method is detailed.

Recent advances in computer vision have produced veritable bevy of works on vision-based state estimators for robotic systems [7, 15, 16, 18, 48]. Direct methods infer motion by reducing photometric error between images and offer strong performance [6, 16], even in low-light environments [3]. Photometric matching performs well on global shutter cameras, but performance suffers on lower-cost, rolling shutter cameras (such as the one used on the experimental platform in this thesis). Optimization-based frameworks can offer accurate and robust performance [19, 47, 48] at the cost of increased computational requirements.

An ideal visual odometry methodology would offer computational efficiency and acceptable performance with lower-cost, rolling shutter cameras. The approach detailed in this chapter is specifically tailored to those two requirements. The approach is based primarily on the Multi-State Constraint Kalman Filter (MSCKF) developed by Mourikis, et al. [46]. The MSCKF is a tightly-coupled visual-inertial Extended Kalman Filter that formulates the observation model as a constraint between multiple camera poses. The efficiency and performance comes through a variety of steps, including state augmentation and a null space projection that eliminates global features from the state. As an additional benefit, the computational efficiency and extensibility that comes with the filter-based approach is ideal for the multi-modal extension formulated in Chapter 4. Other modern implementations of MSCKF can be found in [68, 81].

At the core of MSCKF is the Extended Kalman Filter (EKF) which uses Jacobians, rather than sigma points described in Section 2.1, to propagate uncertainty. A brief summary of the EKF is included in Section 3.1 alongside the state and process models for the MSCKF. The constraint-based observation model and techniques for reducing computational cost are formulated in Section 3.2. The image processing backend for feature detection and tracking, based on work by Sun, et al. [68], is shown in Section 3.3. The state estimator is evaluated through in-flight experimentation and the results are presented in Section 3.4.

3.1 Filter Overview

3.1.1 Extended Kalman Filter Formulation

The Extended Kalman Filter (EKF) is an extension of a Linear Kalman filter to non-linear systems. Whereas the UKF performs linear regression through sigma point propagation, the EKF uses first order derivatives in the form of Jacobians. The standard EKF algorithm is shown in Alg. 3 [70], where **F** and **H** are the Jacobians related to the process and observation model, respectively. For example, the matrix $\mathbf{F}_t$ is of the form:

Algorithm 3 Extended Kalman Filter

```
procedure EXTENDED_KALMAN_FILTER($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$)
    $\bar{\mu}_t = f(u_t, \mu_{t-1})$
    $\bar{\Sigma}_t = F_t \Sigma_{t-1} F_t^\top + R_t$
    $K_t = \bar{\Sigma}_t H_t^\top (H_t \bar{\Sigma}_t H_t^\top + Q_t)^{-1}$
    $\mu_t = \bar{\mu}_t + K(z_t - h(\bar{\mu}_t))$
    $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$
    return $\mu_t, \Sigma_t$
end procedure
```

$$\mathbf{F}_t = \begin{bmatrix} \frac{\partial x_1}{\partial \dot{x}_1} & \frac{\partial x_1}{\partial \dot{x}_2} & \cdots & \frac{\partial x_1}{\partial \dot{x}_N} \\ \frac{\partial x_2}{\partial \dot{x}_1} & \frac{\partial x_2}{\partial \dot{x}_2} & \cdots & \frac{\partial x_2}{\partial \dot{x}_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial x_N}{\partial \dot{x}_1} & \frac{\partial x_N}{\partial \dot{x}_2} & \cdots & \frac{\partial x_N}{\partial \dot{x}_N} \end{bmatrix} \tag{3.1}$$

At first glance, the EKF appears far simpler than the UKF from Alg.1, however, formulation of the Jacobians is not trivial, particularly with more complex models. The MSCKF makes use of a null space projection for the observation model, which is covered in detail in Section 3.2, to eliminate the requirement of incorporating features into the state. The null space projection requires Jacobians and is thus more suited to the EKF. It is worth nothing that while the MSCKF can be implemented with an UKF using the stochastic cloning technique outlined in Chapter 2, the marginal performance gain [7] is outweighed by the additional filter complexity.

3.1.2 State Definition

One major benefit to visual odometry relative to 2D laser-based odometry is that strong assumptions of environmental structure are no longer necessary. The use of feature tracking for odometry allows for observability of both rotation and relative motion in 3D-rich environments. The increased observability allows for a state model that that incorporates the full rotation of the body frame and the camera-to-IMU extrinsics. An illustration of the involved coordinate frames is provided for reference in Fig. 3.1.

The expanded, time-varying state of the IMU is defined as:

$$\mathbf{x} = \begin{bmatrix} {}^{I}\mathbf{q}_{G}^{\top} & \mathbf{b}_{g}^{\top} & {}^{G}\mathbf{v}_{I}^{\top} & \mathbf{b}_{a}^{\top} & {}^{G}\mathbf{p}_{I}^{\top} & {}^{C}\mathbf{q}_{I}^{\top} & {}^{I}\mathbf{p}_{C}^{\top} \end{bmatrix}^{\top} \tag{3.2}$$

where ${}^{I}\mathbf{q}_{G}$ is a unit quaternion representing the rotation from the inertial from to the body frame, and $\mathbf{b}_g$ and $\mathbf{b}_a$ are the IMU biases for gyroscope and accelerometer, respectively. The terms ${}^{G}\mathbf{p}_{I}, {}^{G}\mathbf{v}_{I} \in \mathbb{R}^3$ are the position and velocity of the body frame with respect to the fixed global frame. It is assumed that the IMU frame and body frame are coincident. The final two terms, ${}^{C}\mathbf{q}_{I}, {}^{I}\mathbf{p}_{C}$ represent the camera extrensics.

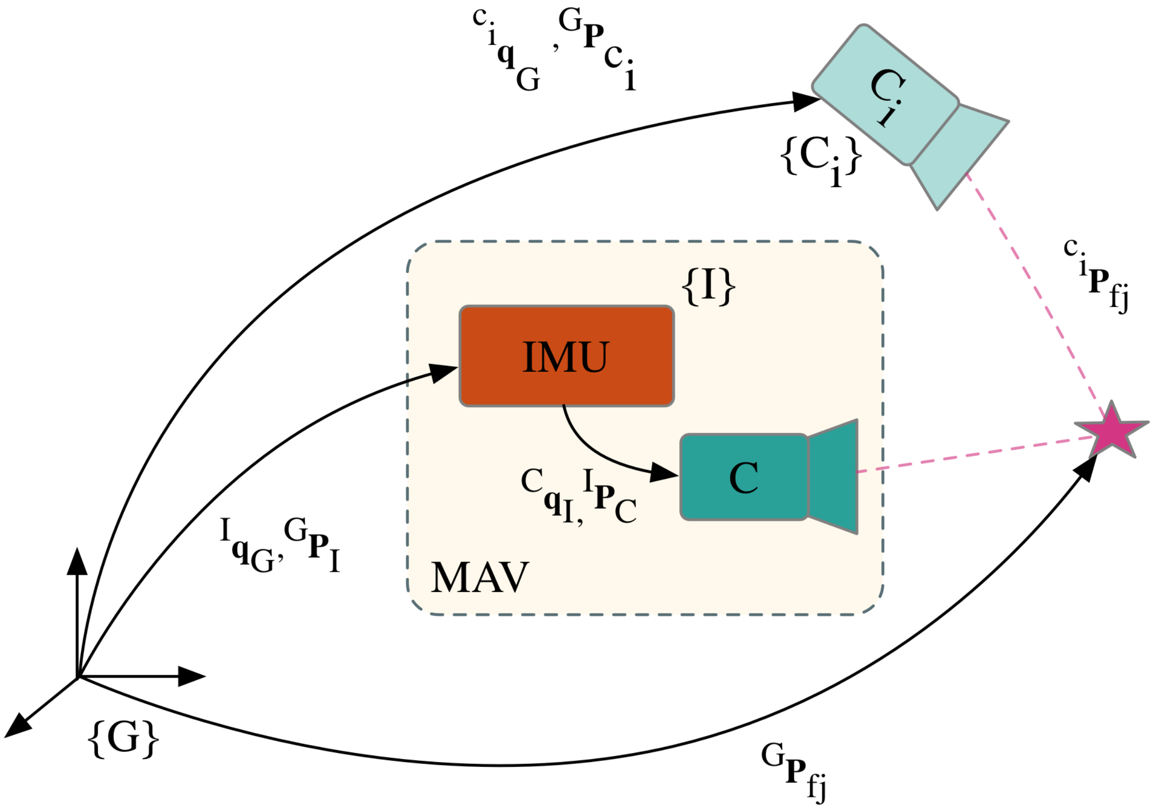

Figure 3.1: Coordinate frames for MSCKF.

The state propagation is modeled with continuous-time dynamics:

$$\begin{aligned} {}^I\dot{\hat{\mathbf{q}}}_G = \frac{1}{2}\Omega(\hat{\omega})^I\hat{\mathbf{q}}_G, \quad \dot{\hat{\mathbf{b}}}_a = \mathbf{0}_{3\times1} \\ {}^G\dot{\hat{\mathbf{v}}}_I =^I \hat{\mathbf{R}}_G^\top \cdot \hat{\mathbf{a}} +^G \mathbf{g} \\ \dot{\hat{\mathbf{b}}}_g = \mathbf{0}_{3\times1}, \quad {}^G\dot{\hat{\mathbf{p}}}_I =^G \hat{\mathbf{v}} \\ {}^I_C\dot{\hat{\mathbf{q}}} = \mathbf{0}_{3\times1}, \quad {}^I\dot{\hat{\mathbf{p}}}_C = \mathbf{0}_{3\times1} \end{aligned} \tag{3.3}$$

where $\hat{\omega}$ and $\hat{\mathbf{a}}$ are the bias compensated angular velocities and linear accelerations from the IMU:

$$\hat{\mathbf{w}} = \mathbf{w}_m - \hat{\mathbf{b}}_g, \hat{\mathbf{a}} = \mathbf{a}_m - \hat{\mathbf{b}}_a \tag{3.4}$$

and $\Omega(\hat{\mathbf{w}})$ maps body-frame angular velocities to global-frame angular velocities:

$$\Omega(\omega) = \begin{bmatrix} -[\omega]_x & \omega \\ -\omega^\top & 0 \end{bmatrix} \tag{3.5}$$

$$[\omega]_x = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{3.6}$$

The use of the unit quaternion for an orientation representation prevents direct use of the vehicle state for residual and uncertainty representation. The unit length constraint can lead to a singular covariance matrix [73]. Therefore, a separate error state is required. The error state for the body frame is defined as:

$$\tilde{\mathbf{x}}_I = \begin{bmatrix} {}^I_G\tilde{\theta} & \tilde{\mathbf{b}}_g & {}^G\tilde{\mathbf{v}}_I & \tilde{\mathbf{b}}_a & {}^G\tilde{\mathbf{p}}_I \end{bmatrix}^\top \tag{3.7}$$

The standard additive error is used for position, velocity, and biases (i.e. $\tilde{x} = x - \hat{x}$). As mentioned above, the unit length constraint on quaternion requires the use of the error quaternion $\delta\mathbf{q} = \mathbf{q} \otimes \hat{\mathbf{q}}^{-1}$, where $\otimes$ represents quaternion multiplication. Given the general formula for a quaternion:

$$\mathbf{q} = \begin{bmatrix} \sin\left(\frac{\theta}{2}\right) v_x & \sin\left(\frac{\theta}{2}\right) v_y & \sin\left(\frac{\theta}{2}\right) v_z & \cos(\theta) \end{bmatrix}^\top \tag{3.8}$$

and using a small-angle approximation, the error quaternion for small rotational errors can be expressed as:

$$\tilde{\mathbf{q}} \approx \begin{bmatrix} \frac{1}{2}\tilde{\theta} & 1 \end{bmatrix}^\top \tag{3.9}$$

which allows attitude errors to be expressed as $\tilde{\theta}$ in a minimal representation with no unit length constraint.

The linearized continuous-time error state propagation is defined as:

$$\dot{\tilde{\mathbf{X}}} = \mathbf{F}\tilde{\mathbf{X}}_I + \mathbf{G}\mathbf{n}_I \tag{3.10}$$

where the IMU noise gyro, accelerometer, and bias noises are given by $\mathbf{n}_I = \begin{bmatrix}\mathbf{n}_g & \mathbf{n}_{bg} & \mathbf{n}_a & \mathbf{n}_{ba}\end{bmatrix}^\top$, $\mathbf{G}$ maps IMU noise to state noise, and $\mathbf{F}$ is a Jacobian representing the linearized continuous-time dynamics given in (3.3). For reference, matrices $\mathbf{F}$ and $\mathbf{G}$ are given by:

$$\mathbf{F} = \begin{bmatrix} -\left[\hat{\omega}\right]_\times & -\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ -{}^I\hat{\mathbf{R}}_G^\top\left[\hat{a}\right]_\times & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & {}^I\hat{\mathbf{R}}_G^\top & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} \end{bmatrix} \tag{3.11}$$

$$\mathbf{G} = \begin{bmatrix} -\mathbf{I}_3 & -\mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -{}^I\hat{\mathbf{R}}_G^\top & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 \\ \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} \end{bmatrix} \tag{3.12}$$

A point of clarity: the *error state* is not an augmented state nor is it explicitly tracked over time like the main state. Rather, it is a mathematical device for which to define the structure of and to propagate the state uncertainty covariance matrix [56].

3.1.3 State Augmentation

The MSCKF observation model is formulated as a constraint between multiple keyframes observing global features. By its nature, these are *relative* observations similar to the ICP odometry seen in Chapter 2. State augmentation is again required to adequately account for the uncertainty of prior states. At a given time, N camera poses $C_i, i \in \{1...N\}$ are augmented to the main state. The i^{th} camera state and its corresponding error state are defined as:

$$\begin{aligned}\hat{\mathbf{x}}_{C_i} &= \begin{bmatrix} {}^{C_i}\hat{\mathbf{q}}_G & {}^{G}\hat{\mathbf{p}}_{C_i} \end{bmatrix}^\top \\ \tilde{\mathbf{x}}_{C_i} &= \begin{bmatrix} \tilde{\theta}_{c_i} & {}^{G}\tilde{\mathbf{p}}_{C_i} \end{bmatrix}^\top \end{aligned} \tag{3.13}$$

where ${}^{C_i}\hat{\mathbf{q}}_G$ and ${}^{G}\hat{\mathbf{p}}_{C_i}$ are given by:

$${}^{C_i}\hat{\mathbf{q}}_G = {}^{C}\hat{\mathbf{q}}_I \otimes {}^{I}\hat{\mathbf{q}}_G, \quad {}^{G}\hat{\mathbf{p}}_{C_i} = {}^{G}\hat{\mathbf{p}}_I + {}^{I}\mathbf{R}_G^{\top I}\hat{\mathbf{p}}_C \tag{3.14}$$

The total error state for the system then is defined as:

$$\tilde{\mathbf{x}} = \begin{bmatrix} \tilde{\mathbf{x}} & \tilde{\mathbf{x}}_{C_1} & ... & \tilde{\mathbf{x}}_{C_N} \end{bmatrix}^\top \tag{3.15}$$

The covariance is augmented with the new camera error state by the following:

$$\mathbf{P}_{k|k} \leftarrow \begin{bmatrix} \mathbf{I}_{6N+15} \\ \mathbf{J} \end{bmatrix} \mathbf{P}_{k|k} \begin{bmatrix} \mathbf{I}_{6N+15} \\ \mathbf{J} \end{bmatrix}^\top \tag{3.16}$$

where $\mathbf{J}$ is found by taking the derivative of (3.13) with respect to (3.9):

$$\mathbf{J} = \begin{bmatrix} {}^I\hat{\mathbf{R}}_G & \mathbf{0}_{3\times 9} & \mathbf{0}_{3\times 3} & \mathbf{I}_3 & \mathbf{0}_{3\times 3} \\ -{}^I\mathbf{R}_G^\top[{}^I\hat{\mathbf{p}}_C]_\times & \mathbf{0}_{3\times 9} & \mathbf{I}_3 & \mathbf{0}_{0\times 0} & \mathbf{I}_3 \end{bmatrix} \tag{3.17}$$

The full augmented state covariance is then given by:

$$\mathbf{P}_{II_{k|k}} = \begin{bmatrix} \mathbf{P}_{II_{k+1|k}} & \mathbf{P}_{IC_{k|k}} \\ \mathbf{P}_{IC_{k|k}}^\top & \mathbf{P}_{CC_{k|k}} \end{bmatrix} \tag{3.18}$$

3.1.4 Discrete-time Process Update

Kalman Filters are formulated as discrete-time recursive filters whereas the state process update was modeled through the continuous-time dynamics given in (3.3). To propagate the state mean through the non-linear process update with discrete IMU measurements, the continuous-time process dynamics are integrated using 4^{th} order Runge-Kutta method. To propagate state uncertainty, the continuous-time error state propagation from (3.10) must also be integrated. The discrete-time state transition and process noise matrices, denoted Φ_k and $\mathbf{Q}_k$ respectively, are given by:

$$\begin{aligned} \boldsymbol{\Phi}_k &= \boldsymbol{\Phi}(t_{k+1}, t_k) = \exp\left(\int_{t_k}^{t_{k+1}} \mathbf{F}(\tau)\, d\tau\right) \\ \mathbf{Q}_k &= \int_{t_k}^{t_{k+1}} \boldsymbol{\Phi}(t_{k+1}, \tau)\mathbf{G}\mathbf{Q}\mathbf{G}\boldsymbol{\Phi}(t_{k+1}, \tau)^\top d\tau \end{aligned} \tag{3.19}$$

The IMU state covariance may then be propagated according to line 3 of Alg. 3:

$$\mathbf{P}_{II_{k+1|k}} = \boldsymbol{\Phi}_{\mathbf{k}}\mathbf{P}_{II_{k|k}}\boldsymbol{\Phi}_{\mathbf{k}}{}^\top + \mathbf{Q}_k \tag{3.20}$$

The discrete-time state transition matrix is also used to propagate the crossvariance terms of the full state covariance:

$$\mathbf{P}_{II_{k|k}} = \begin{bmatrix} \mathbf{P}_{II_{k+1|k}} & \mathbf{\Phi}_k \mathbf{P}_{IC_{k|k}} \\ \mathbf{P}_{IC_{k|k}}^\top \mathbf{\Phi}_k^\top & \mathbf{P}_{CC_{k|k}} \end{bmatrix} \tag{3.21}$$

3.2 Observation Model

The driving premise behind the MSCKF observation model is the notion that viewing a static feature from multiple camera poses results in constraints between all involved camera poses. To reduce computational complexity, the observations are grouped per tracked feature rather than per camera pose, resulting in a state that contains only IMU state and camera poses. Through use of a null space projection, the dependency of the observations on global feature position is eliminated.

In the following subsections, the observation model is formulated and additional techniques to reduce computational cost are outlined. The result is a computationally efficient [68] and probabilistically correct [25] observation model.

3.2.1 Multi-State Constraint Model

The general form for the residual used in the EKF observation update is of the form:

$$\mathbf{r} = \mathbf{H}\tilde{\mathbf{x}} + \mathbf{\Sigma}_z = \mathbf{z} - \hat{\mathbf{z}} \tag{3.22}$$

where $\mathbf{H}$ is the linearized observation model Jacobian, $\tilde{\mathbf{x}}$ is the error state as defined in (3.15), and $\mathbf{\Sigma}_z$ is a diagonal matrix of the observation noise.

To begin, consider the pixel coordinates of single feature f_j that has been observed by a camera poses $\mathbf{x}_{c_i}$ in a set of $\mathbf{M}_i$ camera poses observing the same feature:

$$\mathbf{z}_i^j = \begin{bmatrix} u_i^j \\ v_i^j \end{bmatrix} + \mathbf{n}_i^j = \frac{1}{{}^{c_i}Z_j} \begin{bmatrix} {}^{c_i}X_j \\ {}^{c_i}Y_j \end{bmatrix} + \mathbf{n}_i^j \tag{3.23}$$

where $\mathbf{n}_i^j \sim \mathcal{N}(\mathbf{0}, \sigma_{\mathbf{z}}^{\mathbf{2}})$ is zero-mean additive noise from the camera. The feature position ${}^{c_i}\mathbf{p}_{f_j}$ in the camera frame is given by:

$${}^{c_i}\mathbf{p}_{f_j} = \begin{bmatrix} {}^{c_i}X_j \\ {}^{c_i}Y_j \\ {}^{c_i}Z_j \end{bmatrix} = {}^{C}\mathbf{R}_G({}^{G}\mathbf{p}_{f_j} - {}^{G}\mathbf{p}_{c_i}) \tag{3.24}$$

where ${}^{G}\mathbf{p}_{f_j} \in \mathbb{R}^3$ is the static feature position in the global frame. This value is not directly observable due to the camera pinhole projection model in (3.23) and is initialized through a least-squares minimization. Consequently, the static feature position is inherently coupled with the state, violating independence assumptions. Additionally, the number of features often exceeds the number of keyframes, making incorporation of features into the state unfavorable from an efficiency standpoint. To eliminate this dependency on the global feature, a nulls pace trick is employed.

First, recall the residual form from (3.22). Combining (3.23) and (3.24), it can be seen that the residual contains terms related to camera state error, $\tilde{\mathbf{x}}_{C_i}$, and global feature position error, ${}^{G}\tilde{\mathbf{p}}_{f_j}$. The residual can then be rewritten as two separate terms:

$$\mathbf{r}_i^j \approx \mathbf{H}_{C_i}^j \tilde{\mathbf{x}}_{C_i} + \mathbf{H}_{f_j}^j \,{}^{G}\tilde{\mathbf{p}}_{f_j} + \mathbf{n}_i^j \tag{3.25}$$

and stacking multiple observations yields:

$$\mathbf{r}^j \approx \mathbf{H}_x^j \tilde{\mathbf{x}} + \mathbf{H}_{f_j} {}^G\tilde{\mathbf{p}}_{f_j} + \mathbf{n}^j \tag{3.26}$$

The effect of the global feature position on the residual is eliminated by projecting (3.26) on the left null space, $\mathbf{A}$, of $\mathbf{H}_{f_j}$:

$$\begin{align} \mathbf{A}^\top \mathbf{r}^j = \mathbf{r}_o^j &= \mathbf{A}^\top \mathbf{H}_x^j \tilde{\mathbf{x}} + \mathbf{A}^\top \mathbf{n}^j \tag{3.27} \\ &= \mathbf{H}_{x_o}^j \tilde{\mathbf{x}} + \mathbf{n}_o^j \tag{3.28} \end{align}$$

The resulting residual has the dimensionality $\mathbf{r}_o \in \mathbb{R}^{(2M_j-3)}$. The observation updated in lines 4–6 of Alg. 3 can now be computed using $\mathbf{r}_o$ and $\mathbf{H}_{x_o}^j$ from (3.28). For reader reference, the state and feature observation Jacobians are now provided. Using the chain rule, the observation Jacobians are given by:

$$\begin{align} \mathbf{H}_{C_i}^j &= \frac{\partial \mathbf{z}_i^j}{\partial {}^{C_i}\mathbf{p}_{f_j}} \frac{\partial {}^{C_i}\mathbf{p}_{f_j}}{\partial \mathbf{x}_{C_i}} \nonumber \\ \mathbf{H}_{f_j}^j &= \frac{\partial \mathbf{z}_i^j}{\partial {}^{C_i}\mathbf{p}_{f_j}} \frac{\partial {}^{C_i}\mathbf{p}_{f_j}}{\partial {}^G\mathbf{p}_j} \tag{3.29} \end{align}$$

where:

$$\begin{align} \frac{\partial \mathbf{z}_i^j}{\partial {}^{C_i}\mathbf{p}_{f_j}} &= \begin{bmatrix} 1 & 0 & -\frac{{}^{C_i}\hat{\mathbf{X}}_{f_j}}{{}^{C_i}\hat{\mathbf{Z}}_{f_j}} \\ 0 & 1 & -\frac{{}^{C_i}\hat{\mathbf{Y}}_{f_j}}{{}^{C_i}\hat{\mathbf{Z}}_{f_j}} \end{bmatrix} \tag{3.30} \\ \frac{\partial {}^{C_i}\mathbf{p}_{f_j}}{\partial \mathbf{x}_{C_i}} &= \begin{bmatrix} \left[{}^{C_i}\hat{\mathbf{p}}_j\right]_\times & -{}^{C_i}\hat{\mathbf{R}}_G \end{bmatrix} \tag{3.31} \\ \frac{\partial {}^{C_i}\mathbf{p}_{f_j}}{\partial {}^G\mathbf{p}_{f_j}} &= {}^{C_i}\hat{\mathbf{R}}_G \tag{3.32} \\ \mathbf{H}_{C_i}^j &= \begin{bmatrix} \mathbf{0}_{3\times 21+6i} & \frac{\partial \mathbf{z}_i^j}{\partial {}^{C_i}\mathbf{p}_{f_j}} \frac{\partial {}^{C_i}\mathbf{p}_{f_j}}{\partial \mathbf{x}_{C_i}} & \mathbf{0}_{3\times 6(N-i)} \end{bmatrix} \tag{3.33} \end{align}$$

3.2.2 Sparsity for Computational Efficiency

The observation model detailed in the previous section reduces computational complexity through elimination of static feature positions from the state. One issue with the previously defined model is that the number of observed features can often be quite large. Eliminating them from the state helps, but an observation update processing 10 features in 10 frames will still have a dimensionality of $\mathbb{R}^{170}$ (recall that following the null space projection, the residual is of size $(2M-3)$). As formulated in [46], QR decomposition is used to sparsify the observation Jacobian [4]. The decomposition is defined as:

$$\mathbf{H_X} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{T}_H \\ \mathbf{0} \end{bmatrix} \tag{3.34}$$

where $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are unitary matrices whose columns form the range and null space of $\mathbf{H}_x$ respectively and $\mathbf{T}_H$ is an upper diagonal matrix. Eq. (3.34) then yields:

$$\mathbf{r}_0 = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{T}_H \\ 0 \end{bmatrix} \tilde{\mathbf{X}} + \mathbf{n}_0$$

$$\begin{bmatrix} \mathbf{Q}_1^\top \mathbf{r}_0 \\ \mathbf{Q}\top_2 \mathbf{r}_0 \end{bmatrix} = \begin{bmatrix} \mathbf{T}_H \\ 0 \end{bmatrix} \tilde{\mathbf{X}} + \begin{bmatrix} \mathbf{Q}_1^\top \mathbf{n}_0 \\ \mathbf{Q}\top_2 \mathbf{n}_0 \end{bmatrix} \tag{3.35}$$

Projecting the residual on the unitary matrices allows retention of useful information in $\mathbf{H_X}$ while discarding the noise term $\mathbf{Q}\top_2\mathbf{r}_0$. A new, sparse residual is defined by:

$$\mathbf{r}_n = \mathbf{Q}_1^\top \mathbf{r}_0 = \mathbf{T}_H \tilde{\mathbf{X}} + \mathbf{n}_n \tag{3.36}$$

which can be used to compute the posterior mean observation update of the MSCKF:

$$\mathbf{K} = \mathbf{P}\mathbf{T}_H^\top(\mathbf{T}_H\mathbf{P}\mathbf{T}_H^\top + \mathbf{R}_n)^{-1} \tag{3.37}$$

$$\mathbf{z} - \hat{\mathbf{z}} = \mathbf{K}\mathbf{r}_n. \tag{3.38}$$

The posterior covariance is updated by:

$$\mathbf{P}_{k+1|k+1} = (\mathbf{I}_\delta - \mathbf{K}\mathbf{T}_H)\mathbf{P}_{k+1|k}(\mathbf{I}_\delta - \mathbf{K}\mathbf{T}_H) + \mathbf{K}\mathbf{R}\mathbf{K}^\top \tag{3.39}$$

$$\tag{3.40}$$

Where $\delta = 6N + 21$ is the dimensionality of the full augmented state covariance matrix.

3.3 Feature Detection and Tracking

The observation model formulated in the previous section provides a methodology for which feature observations between multiple camera keyframes can be used to infer motion. The assumption is that sufficient identifiable features are present and traced accurately between keyframes. The model, however, makes no mention of *how* these features are extracted and tracked, nor how keyframes should be initialized and marginalized. The following section develops the image processing backend required to provide meaningful observations to the MSCKF. Feature extraction and tracking is covered first, followed by details of keyframe insertion and removal.The image processing pipeline can be seen in Fig. 3.2.

3.3.1 Feature Extraction and Tracking

Image features serve as representations in the image space of identifiable points within the world, typically corners, that remain consistent through small changes in perspective. Rather than comparing raw intensity values from images, as is performed in photometric methods, feature-based approaches map a geometric representation onto the identifiable points. This geometric representation is at the core of the observation model of the MSCKF.

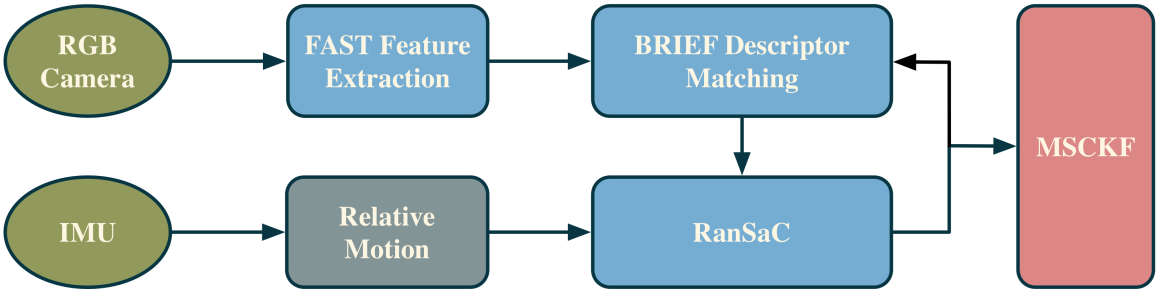

Figure 3.2: Image processing pipeline for MSCKF. FAST features are tracked between frames using Brief Descriptor Matching and outliers are identified through RanSaC. Unmatched features are added added as additional keypoints within the new frame.

There are a variety of methods available for feature detection within images. A summary and comparison of modern visual features is presented by Hartmann, et al. [23] and the paper also notes that a combination of FAST features and BRIEF descriptor matching provides the best balance of tracking performance and computational efficiency. The FAST feature [54] is designed specifically for low computational cost, making it an attractive choice for real-time visual odometry use. To summarize the FAST feature: a candidate point p is classified as a corner if there exists a set of n contiguous points in a circle of 16 pixels around the candidate that are either darker or lighter than the candidate pixel, plus or minus some threshold t. A value of $n = 10$ is the default for the OpenCV implementation and was the value used in experimentation.

The constraint-based formulation of the filter requires accurate and consistent tracking of features between image frames. BRIEF descriptor matching is used to track features between frames. The authors of [68] note that Kanade-Lucas Tracking achieves achieves acceptable performance at a lower computational cost relative to descriptor-based matching. While this may hold true with stereo cameras that provide dual feature observations, monocular odometry requires higher fidelity feature matching. Both methods were evaluated and descriptor-based methods were found to offer far superior performance with an acceptable increase in computational cost. Once tracked between frames, further outliers are removed through Random Sample Consensus. The remaining unmatched features are added to the keyframe for future tracking.

3.3.2 Keyframe Initialization and Marginalization

Image keyframes and the features therein comprise the primary observations for the MSCKF. Keyframe intialization occurs when a new set of features from an image is received. The state is augmented with the new frame and the feature locations stored in a feature server. Observations are delayed until one of two critera is met. When a feature goes out of frame, an observation update is performed on that feature using all involved keyframes. The second criterion is through keyframe marginalization. When a camera frame is removed, an observation update is processed on all features within that frame.

Limited computational resources requires that a finite number of keyframes be maintained at any given time, denoted by N. The authors of [68] propose a method whereby two camera states are removed at every other time step once the buffer limit has been reached, similar to a selection strategy used by Shen, et al. [64]. If the relative motion between the second latest camera state and the state prior is above a user-defined threshold, the oldest camera state is removed (otherwise the second latest state is removed). An alternative method was outlined in the original MSCKF paper [46] that recommends removing $\frac{N}{3}$ evenly-spaced frames whenever the limit is reached. Both methods were evaluated during development of this thesis and the latter method was found to offer superior performance. The sudden jumps in computational load noted by [68] did not have noticeable impact on the state estimator performance during real-time use.

3.4 Results

Performance of the MSCKF state estimator was evaluated through a series of experiments. The quantitative evaluation methodologies were largely the same as the motion capture testing detailed in Chapter 2. The experimental platform was the same, however only observations from the IMU and RGB camera were processed by the filter. In addition to motion capture testing, qualitative evaluation was provided with a loop through the Gates Highbay at Carnegie Mellon University to demonstrate estimator performance in cluttered environments representative of the target domains. Finally, the developed MSCKF formulation was compared with other state of the art visual odometry methodologies on the TUM Motion Capture Dataset [59].

3.4.1 Motion Capture Results

The first flight in the motion capture arena was a hovering flight with only minor translational motion along each axis and little rotational movement. Sufficient vehicle motion was required to produce accurate estimates of 3D feature positions. This flight demonstrated state estimator performance during near-hover conditions. Position estimates and corresponding ground truth are shown in Fig. 3.3. Position RMSE with respect to ground truth for near-hover flights was $\epsilon_{RMSE} = 0.52$.

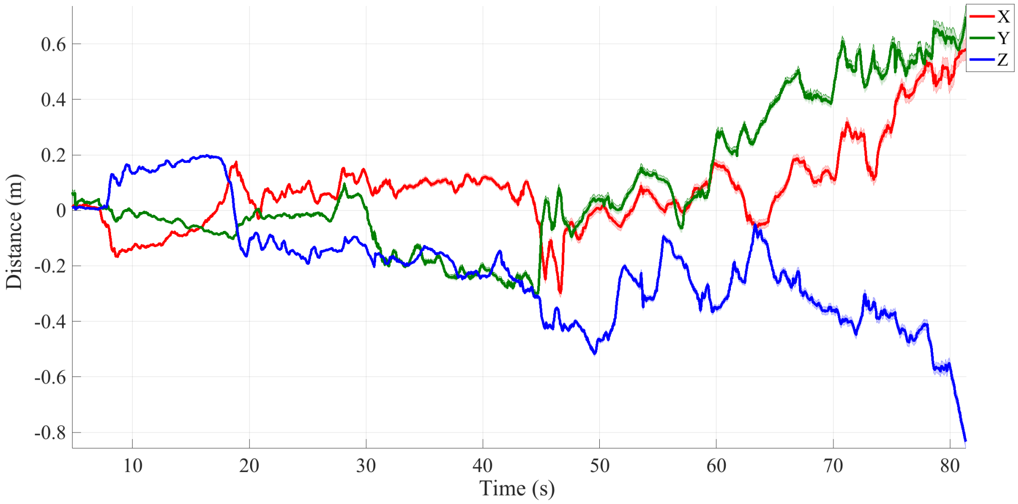

Figure 3.3: Position error with respect to ground truth and corresponding estimator variance during flight in near-hover flight

The second flight is a highly-structured environment with little texture or contrast for feature generation. Position error and covariance are shown in Fig. 3.4. This environment proved challenging for the MSCKF. Lack of unique texture dramatically increased estimator covariance throughout the flight. The estimator performed particular poorly when flying close to the edges of the arena. The lack of distinguishing features in the curtain were exacerbated by feature aliasing when viewing the net, as illustrated in Fig. 3.5. Position RMSE for structured environments was $\epsilon_{RMSE} = 0.45$

The final motion capture flight is through a cluttered, highly-textured environment. Here, ample texture and contrast provided improved performance for the vision-based state estimator. The resultant position error with covariance is shown

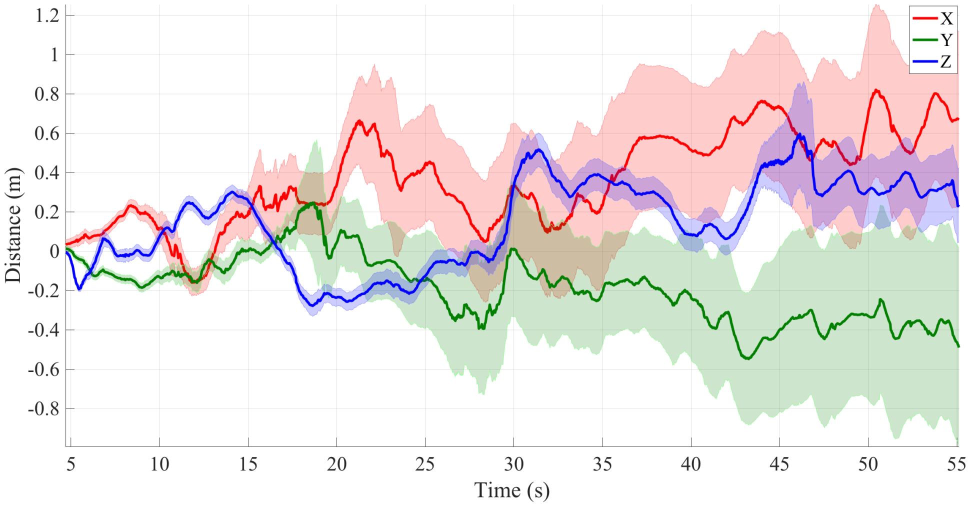

Figure 3.4: Position error with respect to ground truth and corresponding estimator variance during flight in highly structured environment

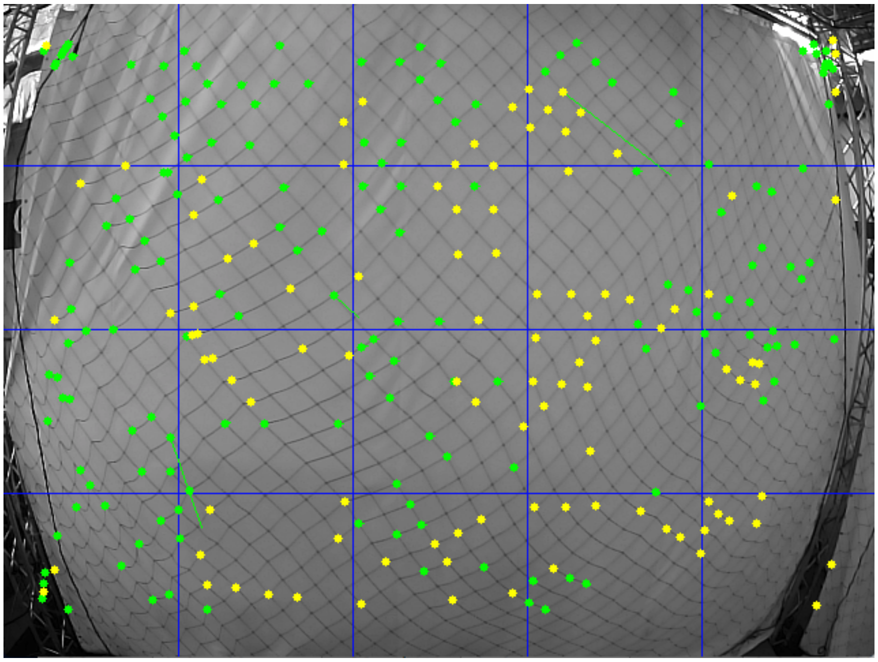

Figure 3.5: Flight close to the net in the motion capture arena produces significant aliasing of feature tracking, degrading visual state estimation performance

in Fig. 2.9. Position RMSE for cluttered environments was $\epsilon_{RMSE} = 0.42$.

The previous results illustrate the effect of the environment on visual state es-

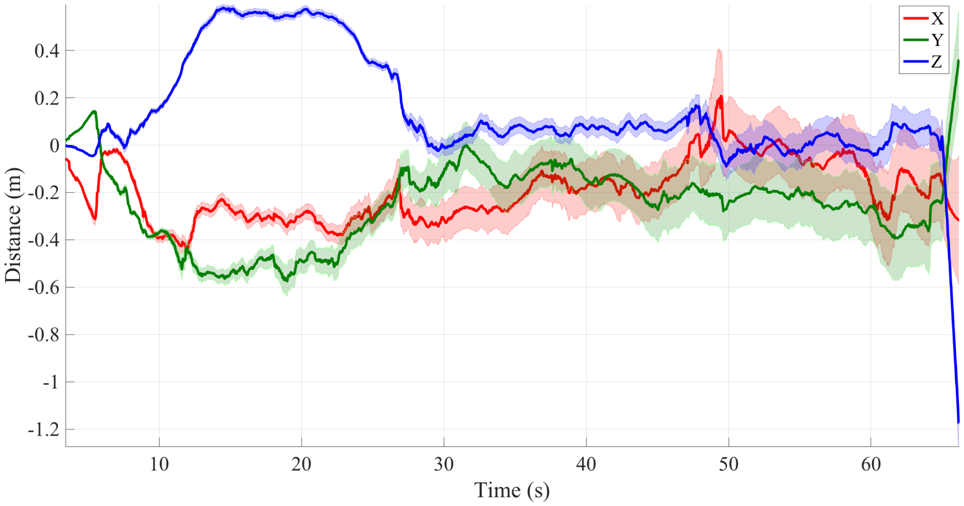

Figure 3.6: Position error with respect to ground truth and corresponding estimator variance during flight in unstructured environment

timator performance. In texture-rich environments, such as the near-hover and unstructured flights, the estimator was able to achieve consistent performance. Flight in a visually-sparse environment produced degraded estimator performance, made worse due to visual aliasing from a net in the motion capture arena. The improved performance in cluttered environments reinforced the motivation for incorporating vision-based modalities. The following section provides qualitative results demonstrating the estimator capability.

3.4.2 Highbay Results

A circular flight was conducted through the Gates Highbay at Carnegie Mellon University to provide qualitative analysis of the MSCKF operating in environments representative of target domains. The flight completed a circular path around the cluttered highbay before returning close to the original starting position. The resultant trajectory is shown in Fig. 3.7. The green circle and red x represent the start and endpoints, respectively. The estimator was able to safely navigate the environment and produce a consistent estimate of vehicle motion, as is visible through the minimal deviation between start and end locations.

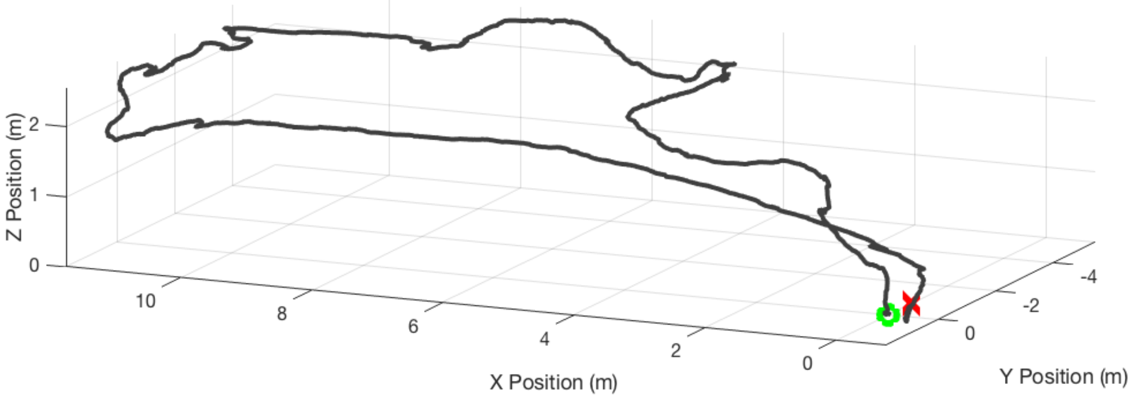

Figure 3.7: Estimated trajectory from MSCKF during a looped traversal of the CMU Gates Highbay.

3.4.3 TUM Visual-Inertial Odometry Dataset Results

The developed monocular MSCKF was evaluated on the TUM Visual Inertial Datasets [59] to compare performance with other state estimation methodologies. In particular, the six *Room* datasets were used to provide ground truth trajectories for comparison. The results for ROVIO [6] and OK-VIS [38] are provided by Schubert, et. al. [59]. In addition to evaluating the monocular MSCKF (Mono-MSCKF) approach developed in this thesis, results are provided for the Stereo MSCKF (S-MSCKF) implementation developed by Sun, et. al. [68]. Table 3.1 provides the cumulative results.

As expected, the stereo, optimization-based approach of OK-VIS achieved the greatest accuracy across all trials. The developed Mono-MSCKF approach was able to outperform the current state of the art monucular odometry method ROVIO in Rooms 1 and 2, and even besting the stereo MSCKF variant in Room 2. The im-

Dataset		**Odometry Method**		
	OK-VIS	S-MSCKF	Mono-MSCKF	ROVIO
Room 1	0.06	0.13	0.13	0.16
Room 2	0.11	0.24	0.20	0.33
Room 3	0.07	0.15	0.19	0.15
Room 4	0.03	0.06	0.12	0.09
Room 5	0.07	0.09	0.13	0.12
Room 6	0.04	0.06	0.20	0.05

Table 3.1: Position RMSE (m) with respect to ground truth for motion capture trials of TUM Visual-Inertial Dataset. The developed method is highlighted in red. Note, the developed method surpasses state of the art monocular performance in Rooms 1 and 2.

proved performance over the stereo approach can be attributed to the superior feature matching and keyframe marginalization methods.

3.5 Chapter Summary

This chapter detailed the implementation of a computationally efficient visual-inertial odometry framework known as the Multi-State Constraint Kalman Filter (MSCKF). The MSCKF is a tightly-coupled Kalman Filter that formulates the observation update as a constraint between multiple camera poses viewing static global features. The MSCKF reduces computational complexity by eliminating global features from the state with a null space projection and further increases efficiency with a QR decomposition for the residual.

The laser-based approach developed in Chapter 2 offered acceptable performance but made strong assumptions of environmental structure. The use of visual feature tracking in the MSCKF eliminates the assumptions of environmental structure and allows for estimation of an expanded vehicle state that incorporates full rotation and IMU-to-camera extrensics. Yet, Vision-based systems do make assumptions of sufficient and consistent illumination, which are explored further in the following chapter.

A feature extraction and tracking backend was developed in Section 3.3 to provide observations for the MSCKF. Keyframe initalization and margininalization techniques were covered in this section as well.

The developed state estimator was compared with modern stereo and monocular methods on publicly available datasets. The developed monocular MSCKF outperformed the current state of the art monocular visual odometry method on multiple trials. The state estimator was then evaluated through in-flight testing in both a motion capture arena as well a circular loop through the texture-rich environment of the Gates Highbay. The MSCKF estimator achieved slightly lower accuracy overall compared to the laser-based system. However, the performance with respect to environmental composition was inverse to the laser-based method. Chapter 4 develops a state estimator that leverages these complimentary failure modes to produce a robust and consistent state estimate.

Chapter 4

Multi-modal State Estimation

The previous chapters have discussed state of the art methods for estimating state using two primary classes of sensors: laser scanners and cameras. As discussed in Chapter 1 and as visible from the results of the two aforementioned methods, the observation models face challenges in accuracy when their underlying assumptions are violated.

The key insight of the complimentary failure modes of laser- and vision-based systems leads to the conclusion that a combined, multi-modal system offers greater accuracy and improved robustness to environmental changes. To that end, the filter from Chapter 3 is extended to incorporate both laser and altimeter observations in Section 4.1. An additional altitude observation model is used to account for changes in surface level, such as flying over obstacles. Additionally, an *absolute* laser observation is formulated to reduce drift over long-duration flights. Incorporation of additional sensor modalities requires additional filter infrastructure to account for out-of-order observations and sensor switching. Section 4.2 addresses both issues. Finally, the robustness of the multi-modal state estimator is evaluated through a series of experiments that highlight particular failure modes for each sensor.

4.1 Laser-Visual Multi-State Constraint Kalman Filter

The backbone of the Laser-Visual MSCKF (LV-MSCKF) is the MSCKF described in Chapter 3. While the UKF provides favorable performance when propagating uncertainty through non-linear models, the null space projection of the MSCKF necessitates the use of Jacobians.

The visual odometry observation model remains, allowing for the estimation of the full state, including rotation. The process model and state augmentation methodologies from Section 3.1 are left unchanged as well. The results from both SC-UKF and MSCKF highlight drift that accumulates from integration of *relative* odometric observations. A new observation method is developed to reduce drift and improve performance on longer duration flights. A diagram of the developed system is shown in Fig. 4.1.

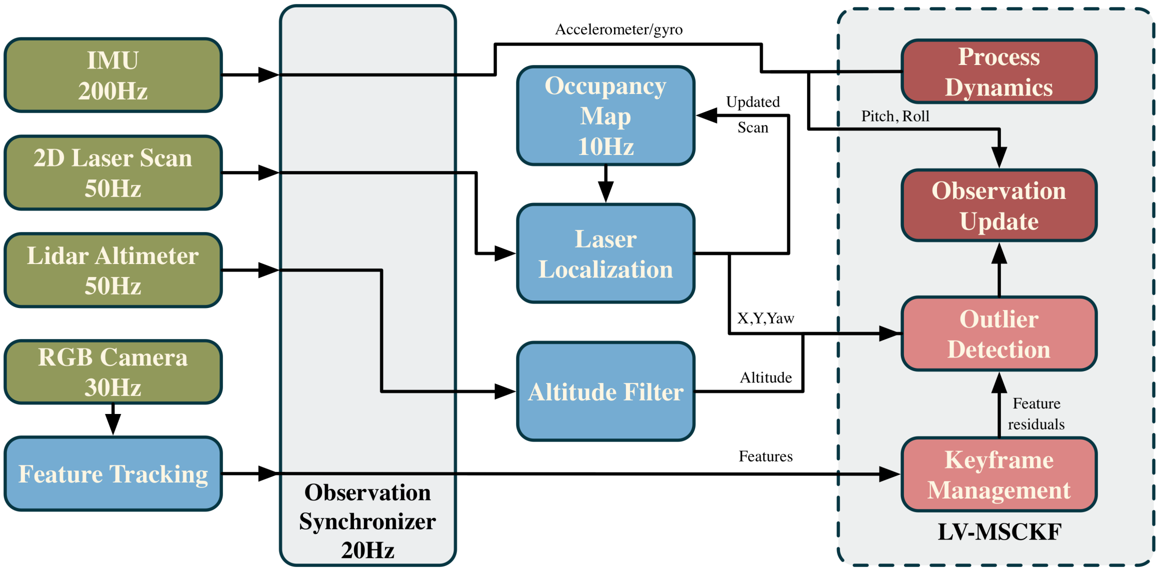

Figure 4.1: System Diagram of LV-MSCKF State Estimator

A full SLAM solution [14, 33, 44, 48] can dramatically reduce drift at the cost of computationally efficiency through a combination of a persistent map and refinement techniques like loop closure and bundle adjustment. Utilizing multi-modal observations requires that both sensing modalities minimize computational complexity in order to facilitate real-time operation. To that end, the developed approach presents a middle ground: reducing drift while maintaining computational tractability. A persistent map is generated through accumulated sensor scans and global position is estimated by performing a grid search of poses to maximize the likelihood of a projected scan into the map. No long-term landmark descriptors or loop closures are employed in order to maintain computational tractability.

4.1.1 Grid-Search Laser Localization

Despite the increased dimensionality of the main state in Chapter 3, the developed laser localization approach still maintains the 2.5D assumption from Chapter 2. A local Occupancy Grid (OG) map is generated and a naive maximum likelihood grid search is used to determine x, y and yaw.

Fig. 4.2 provides an illustration of the OG map used in the laser localization. Cell occupancy is determined through log-odds probability based on 2D laser strikes and hit chance. The map is initialized based on the initial pose of the vehicle and the first scan. Following initialization, laser localization begins.

At each time interval, the extrema of the input scan points are used to form a bounding box, based on which a submap is extracted from the global map. The submap is converted to a distance grid using an OpenCV distance transform image [17], where each pixel value corresponds to the inverse distance to the closest occupied pixel. An occupancy grid map and its corresponding distance transform is

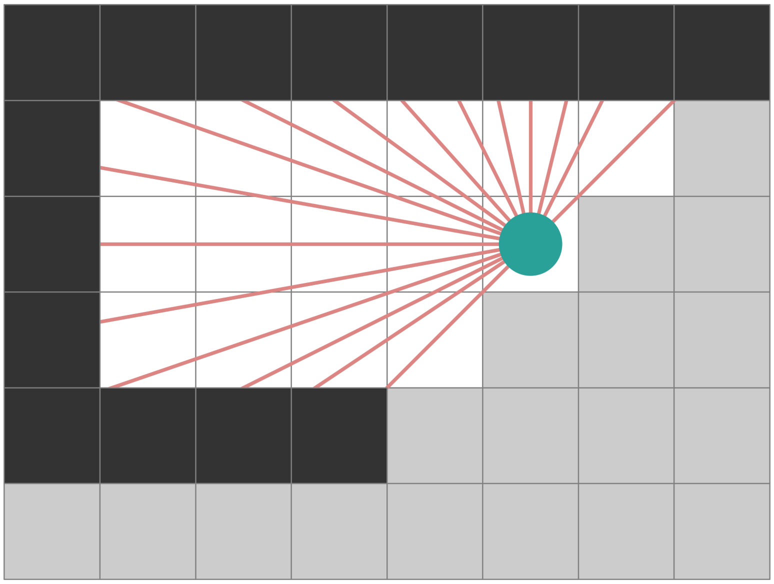

Figure 4.2: Laser scan projected into a generated local occupancy grid map. The robot, projecting the red laser scans, is shown in green. The dark grey cells represent known occupied space, the white cells represent known unoccupied space, and light gray corresponds to undetermined cells.

shown in 4.3. For all 3^n poses, the scan is transformed to a new world frame scan $\mathbf{Q_X}$ with the estimated pose $\mathbf{X}$. The pose likelihood is determined by:

$$P(\mathbf{X}|\mathbf{Q_x}) = \sum_{i=1}^{N} \frac{z_{hit}}{d_i - z_{random}} \tag{4.1}$$

where z_{hit} and z_{random} are fixed user-defined parameters corresponding to scan hit probability and randomness, respectively. The pixel value corresponding to the location of a transformed scan point in the distance transform image is given by d_i. The pose with the largest likelihood is selected as the estimated pose. Following laser localization, the laser scan is added to the map based on the maximum likelihood estimated pose.

Larger grid resolution can reduce computational complexity but may introduce snapping of the pose due to the larger cell size. A low-pass filter is applied in such cases

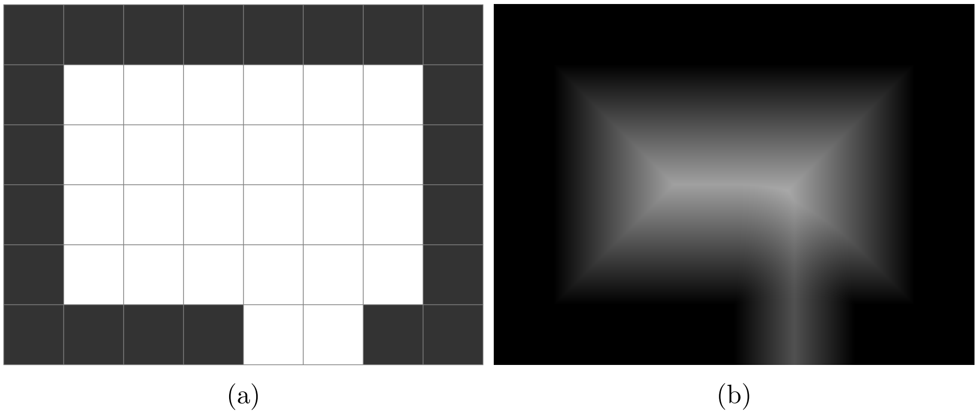

Figure 4.3: (a) Occupancy grid representation and (b) The corresponding distance transform image used in the Laser Localization. Pixel values corresponds to the distance to the nearest 0-valued pixel (occupied space).

to create a smoother estimate of the state transition while maintaining the reduced complexity of the map. During testing, this low-pass filtering had little impact on the performance of the state estimator.

4.1.2 Laser Localization Noise Model

Observational uncertainty is a key component in posterior estimate in the correction update of a Kalman Filter. The accuracy of the laser localization model is affected by a myriad of factors, from environmental structure to surface specularity. Fixed noise parameters are insufficient to express the dynamic uncertainty of the localization model. The work of Censi, et al. [10] provides a more probablistically sound estimate of the *achievable* observational uncertainty. A summary of the model is given in the following paragraphs. Fig. 4.4 is from the aforementioned paper and provides a reference for the notation used in the formulation.

The work of Censi provides a lower bound for the uncertainty that is a function of the expected readings and orientation of the incident surface. The robot pose is

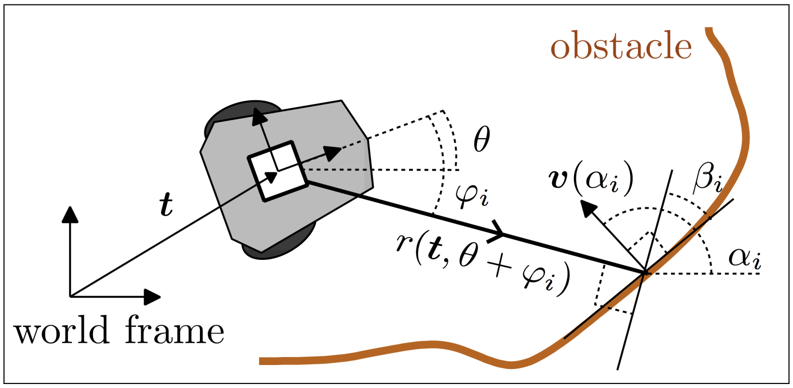

Figure 4.4: Coordinate frame for laser localization noise [10]. The robot coordinates are $\mathbf{x} = (\mathbf{t}, \theta)$, with the i^{th} sensor ray cast in direction φ_i. The true distance of the sense obstacle is $r(\mathbf{t}, \theta + \varphi_i)$. The angle α_i is the direction of corresponding normal vector $\mathbf{v}(\alpha_i)$.

modeled with 2D position and rotation $\mathbf{x} \equiv (\mathbf{t}, \theta)$. The sensor model is given by:

$$\tilde{p}_i = r(\mathbf{t}, \theta + \varphi_i) + \epsilon_i \qquad i = 1 \dots n \tag{4.2}$$

$$r : \mathbb{R}^2 \times [0, 2\pi) \to \mathbb{R}^+ \tag{4.3}$$

where $r(\mathbf{t}, \theta)$ is a ray-tracing function that represents the observed range to the nearest obstacle at heading θ and ϵ_i is zero-mean Gaussian. The lower bounded covariance matrix is defined by the inverse of the Fisher Information Matrix (FIM), which is in turn defined by the first derivatives of the ray-tracing function r with respect to the robot pose.

$$\mathcal{I}(\mathbf{x}) = \frac{1}{\sigma^2} \sum_i^n \left[\frac{\partial r_i}{\partial \mathbf{x}}^\top \frac{\partial r_i}{\partial \mathbf{x}} \right] \tag{4.4}$$

The necessary derivatives depend on the orientation, given by α, of the surface at the given point of intersection. Taking the partial derivatives, the FIM is found to be:

$$\mathcal{I}(\mathbf{x}) = \frac{1}{\sigma^2} \sum_{i}^{n} \begin{bmatrix} \frac{\mathbf{v}(\alpha_i)\mathbf{v}(\alpha_i)^\top}{\cos^2(\beta_i)} & r_i \frac{\tan(\beta_i)}{\cos(\beta_i)} \mathbf{v}(\alpha_i) \\ \mathbf{0} & r_i^2 \tan^2(\beta_i) \end{bmatrix} \tag{4.5}$$

$$\tag{4.6}$$

and the resultant lower-bounded covariance matrix is given by:

$$\Sigma_z = (\mathcal{I}(\mathbf{x}))^{-1} \tag{4.7}$$

The covariance matrix serves as a baseline, and in practice, is scaled to suit the performance requirements of the filter.

4.1.3 Altitude With Surface Height Detection

The MSCKF developed in Chapter 3 observes altitude through integration of visual odometry, however, absolute observations from a downward-facing lidar, as used is Chapter 2, can provide a more accurate estimate. Operation in cluttered indoor environments often requires flight over elevated terrain.

Observing altitude over cluttered, non-planar terrain is not feasible with a single rangefinder, however most human constructed environments contain primarily planar surfaces with varying heights (ie, chairs, tables). This assumption is leveraged to implement an altimeter observation that accounts for changes in floor level. Fig. 4.5 provides an illustration for altimeter outlier rejection and reinitialization.

Altitude observations are generated in the same manner as (2.10), in that the altimeter observation is rotated into the world frame and the observed position in the

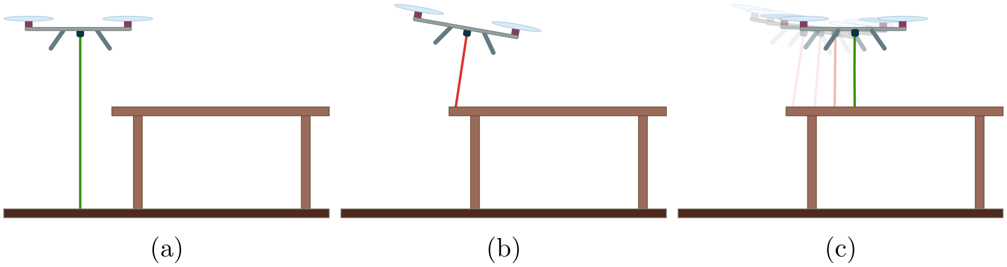

Figure 4.5: (a) MAV with valid altitude observation, (b) Flight over obstacle creates outlier altitude observation, (c) Multiple consecutive outliers are averaged to reinitialize the altitude filter

z axis is compared with the filter. However, a floor height term, h is incorporated, such that:

$$z = \mathbf{c}^I \hat{\mathbf{R}}_G^\top \mathbf{z}_{obs} - \hat{p}_z + h \tag{4.8}$$

When passing over obstacles, the outlier altitude observations are rejected, but stored in a temporary queue. If persistent outliers continue, altimeter is reinitialized with an adjusted floor height:

$$h = \hat{z} - \sum_{i=1}^{N} z_i \tag{4.9}$$

where $\hat{z}$ is the most recent altitude estimate from the filter and $z_i \in Z_o$ are all altitude observations in the outlier set.

4.2 Implementation Details

4.2.1 Observation Synchronization

The addition of sensing modalities to the traditional MSCKF presents challenges related to observation synchronization. On real-time systems, out-of-order or delayed observations are not uncommon. Processing non-consecutive observations fundamentally violates the Markovian assumptions underpinning Bayesian filtering. Additionally, temporally heterogeneous observations should be processed in parallel if arriving with a small time parameter, or otherwise sequentially. Figure 4.6 illustrates varying sensor timings and periods of parallel observations.

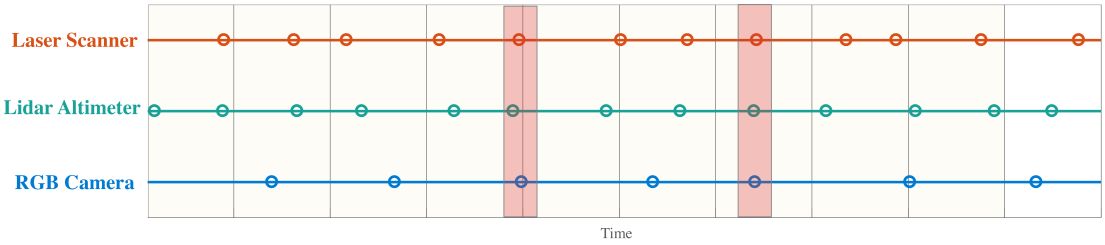

Figure 4.6: Different sensor types often produce observations at dissimilar rates but should be combined when appropriate. The circles represent received observations for the given sensor. The regions highlighted in red represent time periods where observations should be combined.

To account for the aforementioned problems, an observation synchronizer was developed. The filter is run at a fixed rate that is lower than the slowest observation. Given the operational rates of the sensors, which can be seen in 4.1, the filter update rate is set to 20Hz. All incoming observations are placed in a sorted queue. At the defined filter rate, the messages in the queue are processed from oldest to newest. If a set of messages are within a small time delta ($\Delta_t = 0.02$s for this filter, corresponding to the lidar and laser scanner rates), the observations are grouped and processed in parallel. The exception are IMU which are processed separately.

Sensor Type	**Observation Rate (Hz)**
IMU	200
Lidar Altimeter	50
Laser Scanner	50
RGB Camera	30

Table 4.1: Sensor update rates for experimental platform

Another consideration is observation updates that exceed the specified filter rate, leaving some messages unprocessed. There is a potential for messages to accumulate at a rate higher than the filter can process them, creating a snowball effect. Both sensor modalities are developed to be computationally efficient, and as such, the issue never affected the performance of the state estimator during experimental trials.

4.2.2 Handling Observation Degradation

A principle motivator for the filter formulated in this chapter is the insight of the complimentary failure modes between vision- and laser-based system. A robust system must identify and account for failed or degraded sensors. The former, complete sensor failure, is handled by simply continuing other observation updates utilizing the other functional sensors [63, 66].

Observing and quantifying sensor degradation is more challenging than total sensor failure. A myriad of factors can affect the performance of a particular sensor modality throughout operation. Unstructured environments for can compromise laser scan matching or changes in illumination can challenge visual odometry systems. Recent work has sought to address the challenge of introspecting about sensor performance in real-time in the absence of ground truth. The work of Vega, et al. [75] presented an approach based on Expectation Maximization that utilized the sensor observations themselves. Another approach by Hu, et al. [29, 30] used Adaptive Kalman Filtering and a metric defined by the trace of the state covariance to introspect about filter performance as a whole and its responsiveness to parameter changes. One work by Pirmoradi, et al. [51] used redundancy and parallel sensors to detect sensor degradation.

The aforementioned methods have varying effectiveness and the topic is still very much an open research problem. The chosen method for this work is through the use of Mahalanobis distance, a metric for the distance of a point from a distribution, coupled with a χ^2. Use of the Mahalanobis distance is a common method in statistics for detecting outliers within a distribution and has also been applied to sensor degradation detection on multi-sensor aerial platforms [24, 43]. The Mahalanobis distance as applied to state estimation is given by:

$$D_m(\mathbf{x}) = \sqrt{(\mathbf{x} - \mu)^\top \mathbf{S}^{-1}(\mathbf{x} - \mu)} \tag{4.10}$$

$$D_m^2 = \mathbf{r}_z^\top \mathbf{S}_z^{-1} \mathbf{r}_z^\top \tag{4.11}$$

Where

$$\mathbf{r}_z = \mathbf{z} - \hat{\mathbf{z}}$$

is the residual from the observation update step, and

$$\mathbf{S} = (\mathbf{H_t} \bar{\mathbf{\Sigma}}_t \mathbf{H}_t^\top + \mathbf{Q}_t),$$

corresponds to the denomenator of the Kalman Gain matrix from line 4 of Alg. 3.

At every observation update, the squared Mahalanobis distance D_m^2 for the corresponding sensor is calculated. An outlier is defined by any observation such that

$$D_m^2 > \Gamma,$$

where Γ is corresponds to the 95%-Quartile of a χ^2 distribution of n degrees of freedom, and $n : \mathbf{r}_z \in \mathbb{R}^n$ corresponds to the dimensionality of the residual. Outlier observations are rejected and not processed during the observation update.

The Mahalanobis distance check detailed above works well to detect observational outliers, thus detecting observational degradation. Given the dependency on prior observations for the main sensor modalities, e.g., multi-state constraints for vision and map localization for laser, the continuance of degraded observations will ultimately lead to divergence of the respective modality. This condition, in addition to non-trivial periods of missing observations from a sensor, can be classified as sensor failure.

Two separate metrics are used to determine if a sensor is in a failure state. Continued use of degraded observations is detected by maintaining a cumulative sum of Mahalanobis distances:

$$\varrho_t = \begin{cases} 0, & \text{if } D_m^2(\mathbf{r}_{z_t}) < \Gamma \\ \varrho_{t-1} + D_m^2(\mathbf{r}_{z_t}), & \text{otherwise} \end{cases} \tag{4.12}$$

If the cumulative sum for a sensor type exceeds a threshold parameter, $\varrho_t > \delta_D$, or if the time since the last observation exceeds a threshold $\Delta_{t_z} > \delta_t$, the corresponding sensor is put into a failure state. Once a sensor failure is detected, the sensor is reset. For vision, all active features and augmented states are cleared without processing and the system is reinitialized. Processing all prior features and states prior to clearing could result in a large computation times, potentially compromising the other valid observation modes. For laser localization, the map is cleared and reinitialized with the next scan using the current estimated pose.

4.3 Results

The LV-MSCKF is evaluated through similar trials as the previous two chapters. This provides consistency of analysis across a variety of environments and filter types. Additional tests were performed to validate the performance of the LV-MSCKF operating with only vision or laser (in addition to IMU and Lidar altimeter). Figure 4.7 provides a summary of the RMSE values across all trials for all filter types evaluated. In addition to motion capture flights, two separate traversals of challenging environments are conducted to evaluate state estimator performance and to demonstrate robustness to sensor degradation.

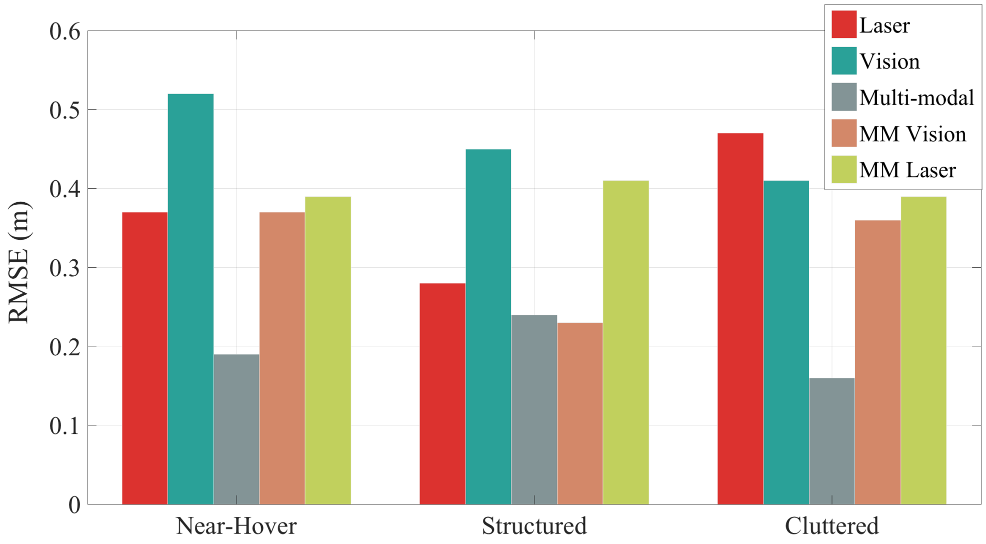

Figure 4.7: Position RMSE for all trials and observation models. RMSE values are averaged across 3 trials for each sensing mode and environment.

4.3.1 Motion Capture Testing

As in the previous two chapters, three separate flights are conducted to provide quantitative analysis of performance. The trials consist of a near-hover flight with minimal translation, a highly-structured environment, and a highly-cluttered environment. Fig. 4.8 shows state estimator performance during near-hover flight. The system is then evaluated in the structured environment in Fig. 4.9, and finally, Fig. 4.10 shows performance in the cluttered, texture-rich environment.

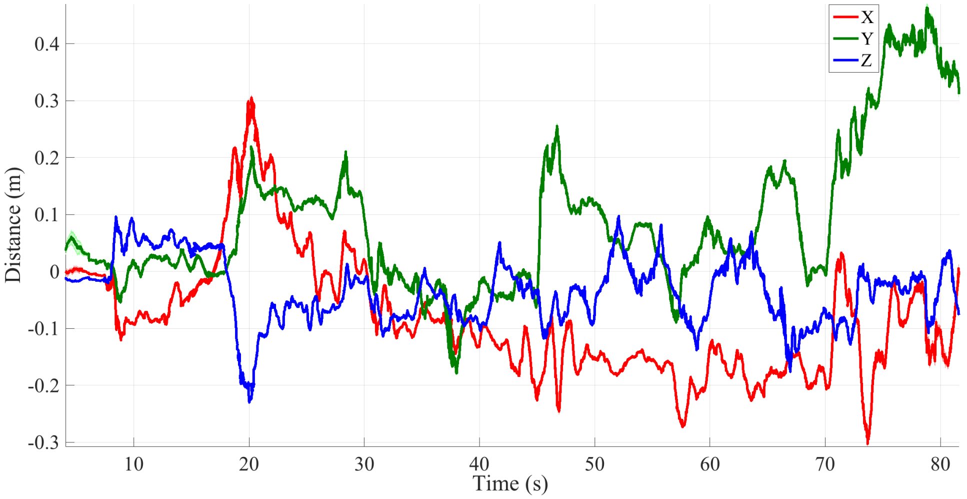

Figure 4.8: Position error LV-MSCKF with respect to ground truth in a near-hover flight

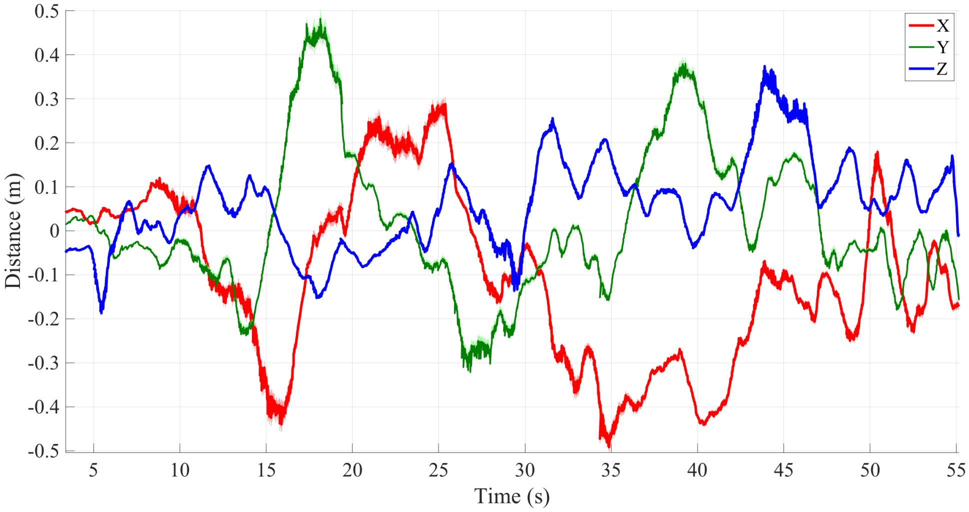

Figure 4.9: Position error of LV-MSCKF with respect to ground truth in structured environment

These experiments highlight the performance of the multi-modal formulation of LV-MSCKF. Overall estimator accuracy was improved dramatically compared to the other uni-modal formulations. Additionally, sensitivity to environmental composition was noticeably reduced. Performance was accurate and consistent across all trials.

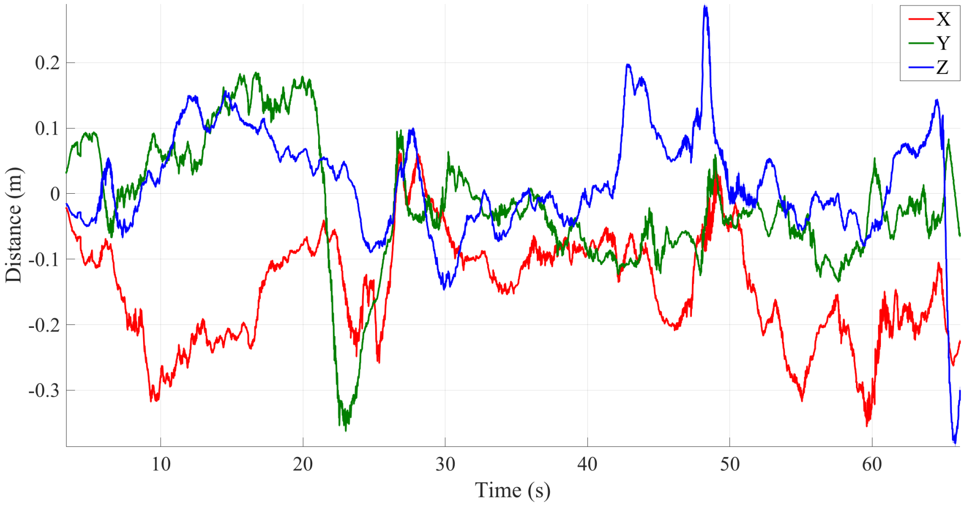

Figure 4.10: Position error of LV-MSCKF with respect to ground truth in cluttered environment

This robustness was demonstrated more thoroughly in the following section, where the robot traversed difficult environments that challenged both sensing modalities.

4.3.2 Traversals of Challenging Environments

The LV-MSCKF demonstrated strong performance in controlled, motion capture settings, however it is valuable to assess real-world performance in environments that represent those found in the target domains. The first evaluation is a Highbay loop like the one performed with the vision-only MSCKF in Chapter 3. As expected, the environment proved little challenge for the improved multi-modal state estimator. Fig. 4.11 shows the resultant trajectory trajectory produced by the estimator. The laser observation allowed for the addition of the observed point cloud for visual representation.

The Highbay loop provided a baseline for LV-MSCKF performance in cluttered, indoor environments, however further testing is needed to evaluate the robustness of

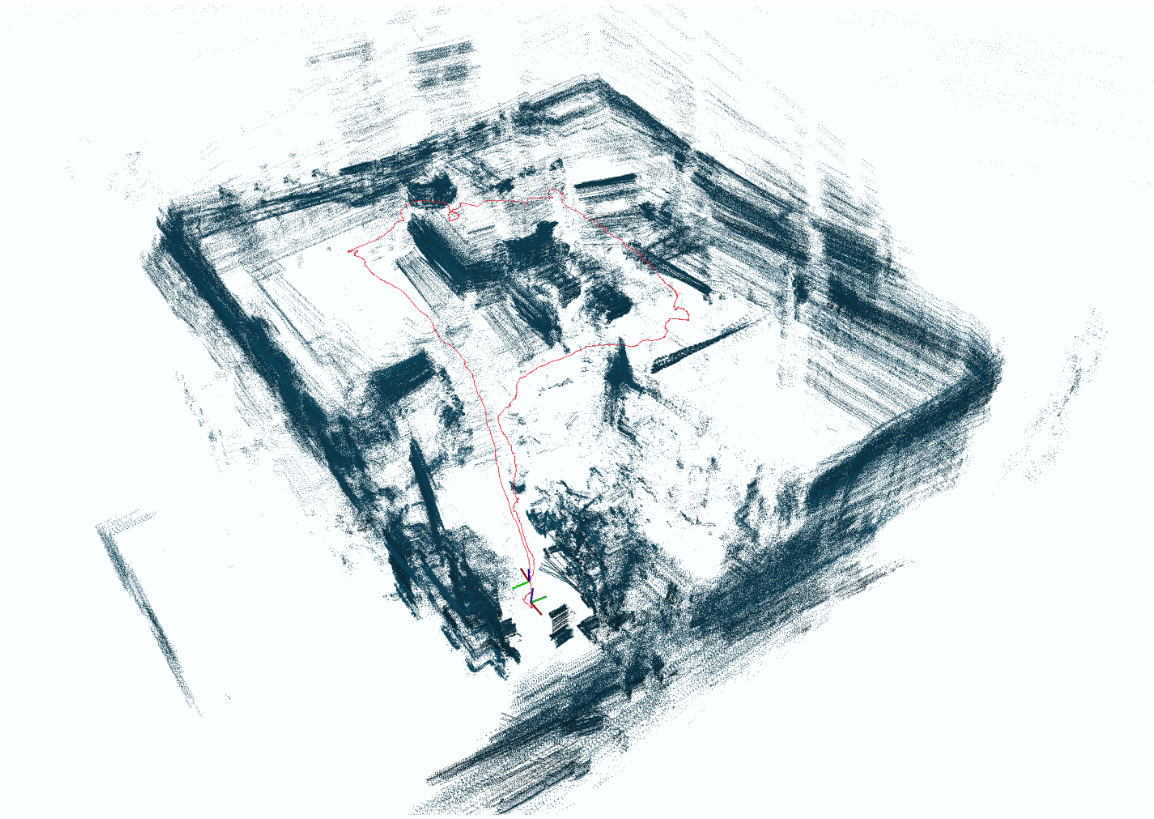

Figure 4.11: Trajectory and point cloud generated by LV-MSCKF during a loop in the CMU Gates Highbay

the multi-modal formulation. A longer duration flight was conducted that traversed a series of hallways, circled the highbay, and returned back to the starting location. The estimated trajectory is shown in Fig. 4.12.

This flight challenged both major sensing modalities at different locations. The long hallways provided difficulty for the laser localization. At numerous points, changes in vehicle attitude caused the planar laser scanner observations to intersect the ceiling of the hallway. These observations produced spurious walls in the localization map that resulted in the vehicle's estimated position traveling *backwards* down the hallways. The gating mechanism introduced in Section 4.1 was able to detect laser localization degradation resultant from map corruption. At several points

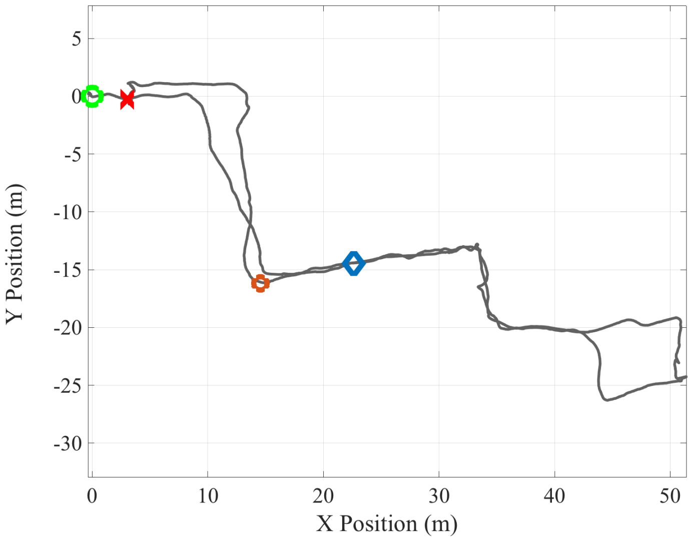

Figure 4.12: Long duration flight traversing a hallway and circling a highbay. Blue diamonds represent locations where the laser observed the ceiling, corrupting the map and degrading the laser localization. The orange circle denotes where the vision system lost tracking while rounding a corner. State estimator performance was unaffected in both cases due to the robust multi-modal formulation.

during the flight, as shown by the blue diamonds in Fig. 4.12, the degraded laser localization was reset and reinitialized in flight without significant impact on state estimator performance. Performing three separate evaluations,

	Start-End RMSE (m)	Estimated Distance(m)
With Gating	2.23	191.91
Without Gating	10.15	188.18

Table 4.2: Estimated RMSE of endpoint with respect to start point and distance traversed during hallway trial, averaged across 3 trials for each method. True endpoint distance was approximately $1m$. Significant error accumulates when outlier rejection and sensor degradation detection is not utilized.

The vision system was also challenged during the flight. Texture was particularly sparse while rounding one corner. Coupled with motion blur, the vision system was unable to extract meaningful features. Fig. 4.13 shows the image captured by the visual odometer at the point of failure. Upon tracking loss, both the image processing backend and the visual odometry portion of the state were reset. The failure point is marked with the orange circle in Fig. 4.12. The vision system was reinitialized without complication and the LV-MSCKF was able to maintain a consistent estimate of the state.

Figure 4.13: Image capturing the point of failure of visual-odometry system. Sparse texture coupled with motion blur produce insufficient contrast for feature extraction.

4.4 Chapter Summary

In this chapter, a multi-modal state estimator for micro aerial vehicles was developed. The state estimator was build on the foundation of the Multi-State Constraint Kalman Filter formulated in Chapter 3 and was extended with expanded forms of the laser-based localization and lidar altimeter observations from Chapter 2.

The odometric observation modalities used in the previous two chapters accumulate unbounded drift due to integration of relative observations. A laser-localization observation model was formulated to reduce the drift on long duration flights. The position and yaw of the vehicle is estimated through a grid-search maximum likelihood estimation by projecting laser scans into a local map. New scans are added to the map based on the optimal estimate from the localization algorithm. The altimeter model was modified to allow for consistent altitude estimation when flying over raised surfaces by adding a floor level variable was incorporated into the observation model.

Two major challenges when performing multi-sensor fusion are observation synchronization and identification and adaptation to degraded sensors. The first challenge was addressed with an observation synchronizer that sorts incoming observations and groups observations that occur within a specified time frame. The filter is run at a fixed rate that is lower than the slowest sensor to allow adequate time for message sorting.

Sensor degradation and failure is detected using accumulated Mahalanobis distance outlier rejection. Individual outlier observations are rejected, but multiple consecutive outliers that exceed a threshold indicate a divergent observation modality. Failed observation modes are reset and reinitialized.

The state estimator was evaluated in flights similar to the experiments and Chapters 2 and 3. Motion capture flights with varying environments were used to provide quantitative evaluation. The multi-modal formulation of the LV-MSCKF demonstrated improved accuracy and robustness to environmental diversity across all trials. A Highbay loop demonstrated baseline performance in cluttered environments. Finally, a long-duration flight traversing structured hallways and a 3D rich Highbay induced observational degradation and failure in both sensing modalities. The multi-modal formulation was able to continue unaffected while the corresponding sensor was reinitialized. These flights demonstrate the ability for LV-MSCKF to provide a state estimate that is consistent, accurate, and robust to both sensor degradation and environmental diversity.

Chapter 5

Conclusion

5.1 Summary

Micro aerial vehicles (MAVs) are an agile platform capable of information gathering in a wide variety of domains, from environmental management to infrastructure inspection and search and rescue. Diverse deployments may include domains where human control is limited or unavailable, creating a necessity for varying degrees of autonomy. State estimation is a core component of autonomy that affects the performance of nearly all other robotic systems. Operation in cluttered industrial environments often prevents the use of GPS and can challenge modern laser- and vision-based observation methods. Additionally, MAVs are subject to size, weight, and power constraints that preclude more information-dense and computationally expensive localization methods. This thesis addressed both issues by implementing and evaluating two modern state estimation methodologies using a 2D laser scanner and a rolling shutter camera. The choice of methodologies was driven by computational efficiency and extensibility. The two observation methods were combined to create a robust, multi-modal state estimator capable of operating in a variety of environments.

Chapter 2 detailed the formulation of an Unscented Kalman Filter that uses a 2D laser scanner as the primary sensing modality. The lack of dense 3D point clouds necessitated a 2.5D assumption of the environment; distance to walls did not vary with respect to vehicle height. The simplified environmental model allowed for motion estimation through an Iterative Closest Point (ICP) odometry algorithm using only 2D planar point clouds. The relative observations produced by the ICP algorithm, which maintains an absolute estimate of vehicle position. A method for augmenting the state with prior clones of poses was detailed to allow for incorporation of relative observations. The state estimator was evaluated through several flights and exhibited strong performance, however violations of environmental structure produced several spurious correction updates.

Chapter 3 detailed an alternative Extended Kalman Filter-based state estimator that uses a camera as the primariy sensing modality. The Multi-State Constraint Kalman Filter (MSCKF) is a tightly-coupled visual odometry algorithm designed specifically for computational efficiency. Per its namesake, the observation model for the MSCKF is formulated as a constraint between multiple camera poses viewing a single static feature. Global feature positions are marginalized out of the observation residual using a null space projection, resulting in an augmented state comprised of only camera poses. An image processing was implemented to extract and track features between camera frames for use in the observation update. Evaluation of the MSCKF was performed through a series of flights similar to the experiments in Chapter 2. While vision-based state estimation offered inferior performance compared to the laser-based method, the MSCKF was less sensitive to changes in environmental structure. Additionally, the monocular MSCKF was shown to compete with and even outperform a state of the art monocular visual odometry method on publicly available datasets.

Chapter 4 extends the MSCKF framework to incorporate additional observations from the 2D laser scanner. The multi-modal formulation, denoted as Laser-Visual MSCKF (LV-MSCKF), was motivated by the insight that laser- and vision-based systems have complimentary failure modes: laser scanners require environmental structure but can operate in low light, whereas vision-based methods are agnostic to environmental structure but highly sensitive to changes in illumination. An alternative laser observation was outlined to reduce accumulated drift by maintaining and localizing against a local map. Properly exploiting the complimentary observation modes necessitates a methodology for identifying sensor performance degradation. The Mahalanobis distance metric was used to detect outliers, and consecutive outliers are used to identify divergence of an observation mode. The LV-MSCKF achieved significantly improved accuracy and consistency across all motion capture trials. Real-world performance was evaluated with a flight through a Highbay. Finally, LV-MSCKF demonstrated robustness to sensor failure and environmental diversity during a long-duration flight that traversed hallways and a cluttered Highbay. Despite moments of observational degradation and failure, the system was able to produce a consistent and accurate state estimate.

5.2 Future Work

Several interesting avenues of future research exist within the field of multi-modal state estimation for MAVs. The cost and performance of RGB-D sensor have been improved due to recent adavancements [34]. A number of works have examined their use in MAV state estimation [31, 48, 82] and others have noted their robustness to lighting changes [42, 76]. The tightly-coupled MSCKF formulation developed by Mourikis et. al. [46] and implemented in Chapter 3 could be generalized to include alternative forms of geometric features, i.e. depth-based features from an RGB-D camera. Observations of depth could allow for expanded observation models that leverage information about underlying environmental structure, such as surface normals, while still maintaining the efficient state-constraint observation model of MSCKF. Generalized ICP [60] and Normal ICP [61] have used similar techniques to greatly improve scan matching performance and robustness.

Bibliography

[1] Markus Achtelik, Abraham Bachrach, Ruijie He, Samuel Prentice, and Nicholas Roy. Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments. page 733219, May 2009.

[2] A. Alaeddini and K. A. Morgansen. Trajectory design for a nonlinear system to insure observability. In *2014 European Control Conference (ECC)*, pages 2520–2525, June 2014.

[3] Hatem Alismail, Brett Browning, and Simon Lucey. Direct Visual Odometry using Bit-Planes. *arXiv:1604.00990 [cs]*, April 2016.

[4] D. S. Bayard and P. B. Brugarolas. An estimation algorithm for vision-based exploration of small bodies in space. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 4589–4595 vol. 7, June 2005.

[5] José-Luis Blanco. A tutorial on se (3) transformation parameterizations and on-manifold optimization. *University of Malaga, Tech. Rep*, 3, 2010.

[6] Michael Bloesch, Sammy Omari, Marco Hutter, and Roland Siegwart. Robust visual inertial odometry using a direct EKF-based approach. pages 298–304. IEEE, September 2015.

[7] Martin Brossard, Silvere Bonnabel, and Axel Barrau. Unscented Kalman Filter on Lie Groups for Visual Inertial Odometry. page 9.

[8] A. Bry, A. Bachrach, and N. Roy. State estimation for aggressive flight in GPS-denied environments using onboard sensing. In *2012 IEEE International Conference on Robotics and Automation*, pages 1–8, May 2012.

[9] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, Jose Neira, Ian Reid, and John J. Leonard. Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age. *IEEE Transactions on Robotics*, 32(6):1309–1332, December 2016.

[10] Andrea Censi. On achievable accuracy for range-finder localization. In *Robotics and Automation, 2007 IEEE International Conference On*, pages 4170–4175. IEEE, 2007.

[11] Yang Chen and Gérard Medioni. Object Modeling by Registration of Multiple Range Images. *Image Vision Comput.*, 10:145–155, January 1992.

[12] Yang Cheng, Mark Maimone, and Larry Matthies. Visual odometry on the Mars exploration rovers. In *Systems, Man and Cybernetics, 2005 IEEE International Conference On*, volume 1, pages 903–910. IEEE, 2005.

[13] Carlos Costa, Heber Sobreira, Armando Sousa, and Germano Veiga. *Robust and Accurate Localization System for Mobile Manipulators in Cluttered Environments*, volume 2015. March 2015.

[14] Frank Dellaert. Factor Graphs and GTSAM: A Hands-on Introduction. page 27.

[15] Kevin Eckenhoff, Patrick Geneva, and Guoquan Huang. Continuous Preintegration Theory for Graph-based Visual-Inertial Navigation. *arXiv:1805.02774 [cs]*, May 2018.

[16] J. Engel, V. Koltun, and D. Cremers. Direct Sparse Odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2017.

[17] Pedro F Felzenszwalb and Daniel P Huttenlocher. Distance Transforms of Sampled Functions. page 15.

[18] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. Georgia Institute of Technology, 2015.

[19] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: Fast semi-direct monocular visual odometry. In *Robotics and Automation (ICRA), 2014 IEEE International Conference On*, pages 15–22. IEEE, 2014.

[20] Paul Furgale, Joern Rehder, and Roland Siegwart. Unified temporal and spatial calibration for multi-sensor systems. pages 1280–1286. IEEE, November 2013.

[21] G. Grisettiyz, C. Stachniss, and W. Burgard. Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling. pages 2432–2437. IEEE, 2005.

[22] Beom W. Gu, Su Y. Choi, Young Soo Choi, Guowei Cai, Lakmal Seneviratne, and Chun T. Rim. Novel Roaming and Stationary Tethered Aerial Robots for Continuous Mobile Missions in Nuclear Power Plants. *Nuclear Engineering and Technology*, 48(4):982–996, August 2016.

[23] J. Hartmann, J. H. Klüssendorff, and E. Maehle. A comparison of feature descriptors for visual SLAM. In *2013 European Conference on Mobile Robots*, pages 56–61, September 2013.

[24] Karol Hausman, Stephan Weiss, Roland Brockers, Larry Matthies, and Gaurav S. Sukhatme. Self-calibrating multi-sensor fusion with probabilistic measurement validation for seamless sensor switching on a UAV. pages 4289–4296. IEEE, May 2016.

[25] Joel A Hesch, Dimitrios G Kottas, Sean L Bowman, and Stergios I Roumeliotis. Observability-constrained Vision-aided Inertial Navigation. page 24.

[26] Joel A Hesch, Dimitrios G Kottas, Sean L Bowman, and Stergios I Roumeliotis. Camera-IMU-based localization: Observability analysis and consistency improvement. *The International Journal of Robotics Research*, 33(1):182–201, January 2014.

[27] Wolfgang Hess, Damon Kohler, Holger Rapp, and Daniel Andor. Real-time loop closure in 2D LIDAR SLAM. In *Robotics and Automation (ICRA), 2016 IEEE International Conference On*, pages 1271–1278. IEEE, 2016.

[28] Timo Hinzmann, Thomas Schneider, Marcin Dymczyk, Andreas Schaffner, Simon Lynen, Roland Siegwart, and Igor Gilitschenski. Monocular Visual-Inertial SLAM for Fixed-Wing UAVs Using Sliding Window Based Nonlinear Optimization. In *Advances in Visual Computing*, pages 569–581. Springer, Cham, December 2016.

[29] Congwei Hu, Wu Chen, Yongqi Chen, Dajie Liu, and others. Adaptive Kalman filtering for vehicle navigation. *Journal of Global Positioning Systems*, 2(1):42–47, 2003.

[30] Humphrey Hu and George Kantor. Introspective Evaluation of Perception Performance for Parameter Tuning without Ground Truth. July 2017.

[31] Albert S. Huang, Abraham Bachrach, Peter Henry, Michael Krainin, Daniel Maturana, Dieter Fox, and Nicholas Roy. Visual Odometry and Mapping for Autonomous Flight Using an RGB-D Camera. In Henrik I. Christensen and Oussama Khatib, editors, *Robotics Research*, number 100 in Springer Tracts in Advanced Robotics, pages 235–252. Springer International Publishing, 2017.

[32] Simon J Julier and Jeffrey K Uhlmann. A New Extension of the Kalman Filter to Nonlinear Systems. page 12.

[33] M. Kaess, A. Ranganathan, and F. Dellaert. iSAM: Incremental Smoothing and Mapping. *IEEE Transactions on Robotics*, 24(6):1365–1378, December 2008.

[34] Leonid Keselman, John Iselin Woodfill, Anders Grunnet-Jepsen, and Achintya Bhowmik. Intel RealSense Stereoscopic Depth Cameras. *arXiv preprint arXiv:1705.05548*, 2017.

[35] A. Koppel, J. Fink, G. Warnell, E. Stump, and A. Ribeiro. Online learning for characterizing unknown environments in ground robotic vehicle models. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 626–633, October 2016.

[36] Bor-Woei Kuo, Hsun-Hao Chang, Yung-Chang Chen, and Shi-Yu Huang. A Light-and-Fast SLAM Algorithm for Robots in Indoor Environments Using Line Segment Map. *Journal of Robotics*, 2011:1–12, 2011.

[37] T. Lefebvre, H. Bruyninckx, and J. De Schuller. Comment on ”A new method for the nonlinear transformation of means and covariances in filters and estimators”

[with authors' reply]. *IEEE Transactions on Automatic Control*, 47(8):1406–1409, August 2002.

[38] Stefan Leutenegger, Paul Furgale, Vincent Rabaud, Margarita Chli, Kurt Konolige, and Roland Siegwart. Keyframe-Based Visual-Inertial SLAM using Nonlinear Optimization. Robotics: Science and Systems Foundation, June 2013.

[39] Mingyang Li and Anastasios I. Mourikis. High-precision, consistent EKF-based visual-inertial odometry. *The International Journal of Robotics Research*, 32(6):690–711, 2013.

[40] Kai Lingemann, Andreas Nüchter, Joachim Hertzberg, and Hartmut Surmann. High-speed laser localization for mobile robots. *Robotics and Autonomous Systems*, 51(4):275–296, June 2005.

[41] Giuseppe Loianno, Michael Watterson, and Vijay Kumar. Visual inertial odometry for quadrotors on SE (3). In *Robotics and Automation (ICRA), 2016 IEEE International Conference On*, pages 1544–1551. IEEE, 2016.

[42] Y. Lu and D. Song. Robustness to lighting variations: An RGB-D indoor visual odometry using line segments. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 688–694, September 2015.

[43] Simon Lynen, Markus W. Achtelik, Stephan Weiss, Margarita Chli, and Roland Siegwart. A robust and modular multi-sensor fusion approach applied to MAV navigation. pages 3923–3929. IEEE, November 2013.

[44] Ellon Mendes, Pierrick Koch, and Simon Lacroix. ICP-based pose-graph SLAM. In *Safety, Security, and Rescue Robotics (SSRR), 2016 IEEE International Symposium On*, pages 195–200. IEEE, 2016.

[45] Anastasios I. Mourikis and Stergios I. Roumeliotis. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Robotics and Automation, 2007 IEEE International Conference On*, pages 3565–3572. IEEE, 2007.

[46] Anastasios I. Mourikis and Stergios I. Roumeliotis. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Robotics and Automation, 2007 IEEE International Conference On*, pages 3565–3572. IEEE, 2007.

[47] Raul Mur-Artal, J. M. M. Montiel, and Juan D. Tardos. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, 31(5):1147–1163, October 2015.

[48] Raul Mur-Artal and Juan D. Tardos. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, October 2017.

[49] J. Nikolic, M. Burri, J. Rehder, S. Leutenegger, C. Huerzeler, and R. Siegwart. A UAV system for inspection of industrial facilities. pages 1–8. IEEE, March 2013.

[50] Edwin B. Olson. *Robust and Efficient Robotic Mapping.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2008.

[51] F. N. Pirmoradi, F. Sassani, and C. W. de Silva. Fault detection and diagnosis in a spacecraft attitude determination system. *Acta Astronautica*, 65(5):710–729, September 2009.

[52] François Pomerleau, Francis Colas, Roland Siegwart, and Stéphane Magnenat. Comparing ICP variants on real-world data sets: Open-source library and experimental protocol. *Autonomous Robots*, 34(3):133–148, 2013.

[53] Tong Qin, Peiliang Li, and Shaojie Shen. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *arXiv:1708.03852 [cs]*, August 2017.

[54] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):105–119, January 2010.

[55] S.I. Roumeliotis and J.W. Burdick. Stochastic cloning: A generalized framework for processing relative state measurements. volume 2, pages 1788–1795. IEEE, 2002.

[56] Stergios I. Roumeliotis, Gaurav S. Sukhatme, and George A. Bekey. Circumventing dynamic modeling: Evaluation of the error-state kalman filter applied to mobile robot localization. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference On*, volume 2, pages 1656–1663. IEEE, 1999.

[57] Jörg Röwekämper, Christoph Sprunk, Gian Diego Tipaldi, Cyrill Stachniss, Patrick Pfaff, and Wolfram Burgard. On the position accuracy of mobile robot localization based on particle filters combined with scan matching. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference On*, pages 3158–3164. IEEE, 2012.

[58] Simo Särkkä, Aki Vehtari, and Jouko Lampinen. Rao-Blackwellized particle filter for multiple target tracking. *Information Fusion*, 8(1):2–15, January 2007.

[59] David Schubert, Thore Goll, Nikolaus Demmel, Vladyslav Usenko, Jörg Stückler, and Daniel Cremers. The TUM VI Benchmark for Evaluating Visual-Inertial Odometry. *arXiv:1804.06120 [cs]*, April 2018.

[60] Aleksandr Segal, Dirk Haehnel, and Sebastian Thrun. Generalized-icp. In *Robotics: Science and Systems*, volume 2, page 435, 2009.

[61] J. Serafin and G. Grisetti. NICP: Dense normal based point cloud registration. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 742–749, September 2015.

[62] Shaojie Shen and Nathan Michael. State Estimation for Indoor and Outdoor Operation with a Micro-Aerial Vehicle. In *Experimental Robotics*, volume 88, pages 273–288. Springer International Publishing, Heidelberg, 2013.

[63] Shaojie Shen, Yash Mulgaonkar, Nathan Michael, and Vijay Kumar. Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft MAV. In *Robotics and Automation (ICRA), 2014 IEEE International Conference On*, pages 4974–4981. IEEE, 2014.

[64] Shaojie Shen, Yash Mulgaonkar, Nathan Michael, and Vijay Kumar. Initialization-Free Monocular Visual-Inertial State Estimation with Application to Autonomous MAVs. volume 109, pages 211–227. January 2016.

[65] Hocheol Shin, Changhoi Kim, Yongchil Seo, Kyungmin Jeong, Youngsoo Choi, Byungseon Choi, and Jeikwon Moon. Aerial work robot for a nuclear power plant with a pressurized heavy water reactor. *Annals of Nuclear Energy*, 92:284–288, June 2016.

[66] Yu Song, Stephen Nuske, and Sebastian Scherer. A Multi-Sensor Fusion MAV State Estimation from Long-Range Stereo, IMU, GPS and Barometric Sensors. *Sensors*, 17(1):11, December 2016.

[67] Hauke Strasdat. *Local Accuracy and Global Consistency for Efficient SLAM.* PhD thesis, Imperial College London, 2012.

[68] Ke Sun, Kartik Mohta, Bernd Pfrommer, Michael Watterson, Sikang Liu, Yash Mulgaonkar, Camillo J. Taylor, and Vijay Kumar. Robust Stereo Visual Inertial Odometry for Fast Autonomous Flight. *arXiv:1712.00036 [cs]*, November 2017.

[69] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. volume 1, pages 321–328. IEEE, 2000.

[70] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics.* MIT press, 2005.

[71] Sebastian Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 128(1–2):99–141, May 2001.

[72] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grixa, F. Ruess, M. Suppa, and D. Burschka. Toward a Fully Autonomous UAV: Research Platform for Indoor and Outdoor Urban Search and Rescue. *IEEE Robotics Automation Magazine*, 19(3):46–56, September 2012.

[73] Nikolas Trawny and Stergios I. Roumeliotis. Indirect Kalman filter for 3D attitude estimation. *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep*, 2:2005, 2005.

[74] V. Usenko, J. Engel, J. Stückler, and D. Cremers. Direct visual-inertial odometry with stereo cameras. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1885–1892, May 2016.

[75] William Vega-Brown and Nicholas Roy. CELLO-EM: Adaptive sensor models without ground truth. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference On*, pages 1907–1914. IEEE, 2013.

[76] A. R. Vetrella, A. Savvaris, G. Fasano, and D. Accardo. RGB-D camera-based quadrotor navigation in GPS-denied and low light environments using known 3D markers. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 185–192, June 2015.

[77] E.A. Wan and R. Van Der Merwe. The unscented Kalman filter for nonlinear estimation. pages 153–158. IEEE, 2000.

[78] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli, and R. Siegwart. Real-time onboard visual-inertial state estimation and self-calibration of MAVs in unknown environments. In *2012 IEEE International Conference on Robotics and Automation*, pages 957–964, May 2012.

[79] Z. Yang and S. Shen. Monocular Visual #x2013;Inertial State Estimation With Online Initialization and Camera #x2013;IMU Extrinsic Calibration. *IEEE Transactions on Automation Science and Engineering*, 14(1):39–51, January 2017.

[80] Ji Zhang and Sanjiv Singh. LOAM: Lidar Odometry and Mapping in Real-time. In *Robotics: Science and Systems*, volume 2, 2014.

[81] Teng Zhang, Kanzhi Wu, Daobilige Su, Shoudong Huang, and Gamini Dissanayake. An Invariant-EKF VINS Algorithm for Improving Consistency. *arXiv:1702.07920 [cs]*, February 2017.

[82] Yigong Zhang, Zhixing Hou, Jian Yang, and Hui Kong. Maximum clique based RGB-D visual odometry. In *Pattern Recognition (ICPR), 2016 23rd International Conference On*, pages 2764–2769. IEEE, 2016.

[83] X. Zheng, Z. Moratto, M. Li, and A. I. Mourikis. Photometric patch-based visual-inertial odometry. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3264–3271, May 2017.