Towards Autonomous Landing for a Quadrotor, using Monocular SLAM Techniques

Jonatan Olofsson

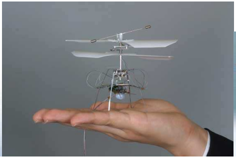
Department of Computer and Information Science
Linköping University, Sweden

Master's Thesis Presentation, 2012-06-08

Presentation Outline

Background

- Increased interest in civilian applications.
- For many applications, a small scale vehicle is desired.
- MAV - Micro Air Vehicle; UAV weighing 5 kg or less[1].


[1] Definitions differ

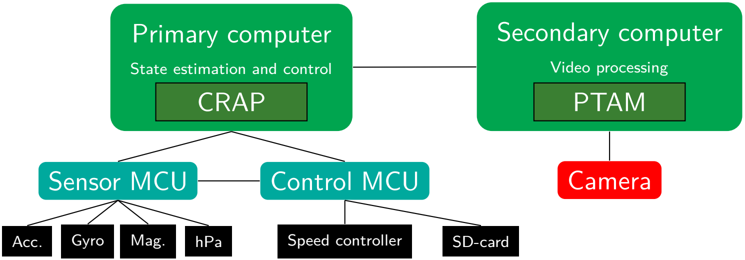
The LinkQuad Platform

Quadrotor research platform developed at AIICS at the Department of Computer and Information Science of Linköping University.

Available sensors: Accelerometers, gyroscopes, pressure sensor, camera, magnetometers, GPS...

The LinkQuad Platform

- Dual gumstix micro-computers (Linux).
- Sensor-board with dual microcontrollers for sensor sampling, data logging and low-level control.



Problem Formulation

Primary goal: Develop a control system for the LinkQuad that use video-based positioning to enable stable landing.

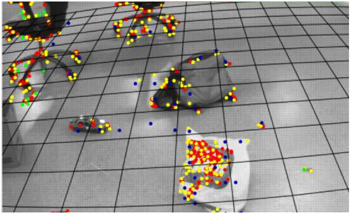
Breakdown:

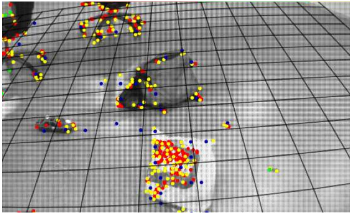
- Localization with Video-based SLAM.
- Sensor Fusion with available sensors.
- Use state estimate for control.
- Generate control reference for landing procedure.
- Landing detection.

Method: Video-based Positioning

Extract 3D information from 2D video-stream.

- vSLAM - Visual SLAM.
- Simultaneous Localisation And Mapping to track features in the video-stream.
- Gives orientation and position relative to the features.
- High computational demands.
- Complex sensor measurements.





Method: Sensor Fusion

Current implementation is based on complementary filtering.

$$\text{angle} = \mathcal{H}_{HP}\left(\int \text{gyro}\,\text{dt}\right) + \mathcal{H}_{LP}\left(accelerometers\right)$$

Performance is adequate, but the c.f is difficult to extend.

Also, camera measurements

- Cannot be used directly in the current filter,
- Fits well into a standard state-space filter framework.

A high-level filtering framework with an advanced motion model was implemented and applied for state estimation.

Method: Control

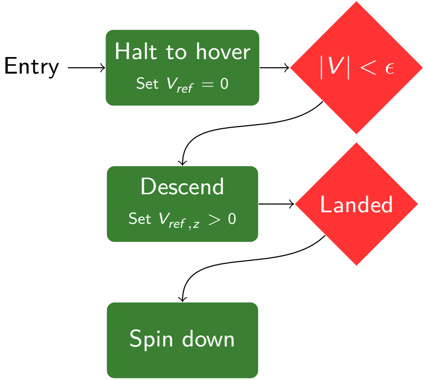
With a linear motion model available, control signals can be computed optimally. Linear Quadratic control offers:

- Simple implementation.
- Simple tuning.
- Optimal control.

Motion model needs to be linear, which is not the case for a quadrotor. This constraint can be circumvented by an extension to LQ control using the **State-Dependent Riccati Equation**.

Method: Reference Generation

With properly implemented control, landing is a matter of descending steadily until landing is detected.



Method: Landing Detection

Landing is generally associated with a lack of descent.

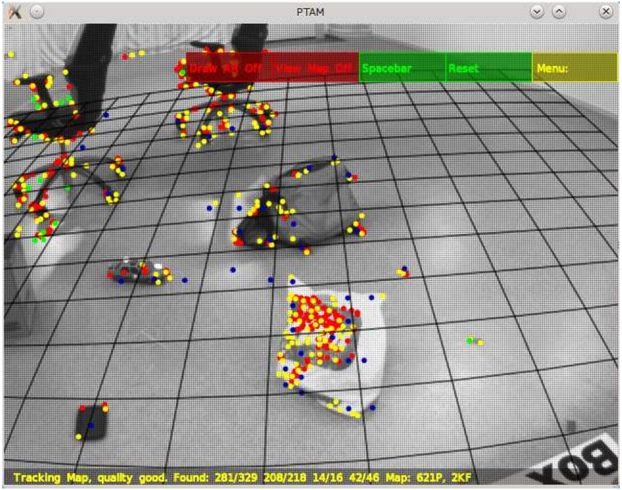
Two interesting states to monitor:

- Vertical velocity.
- Vertical wind velocity.

Without modeled ground force, the observer will explain the added force with upward winds.

Monocular SLAM

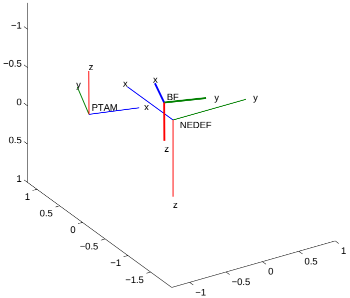
PTAM - Parallel Tracking And Mapping



Measurements are not metric.

Camera Measurements

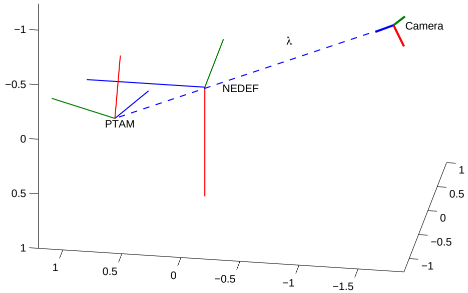
Measurement: Pose relative to PTAM coordinate system.



- PTAM's coordinate system is quite arbitrarily initialised.
- Rotation and translation are measured in the local PTAM coordinate system.
- Its relation to the world must be established.

Initialization

PTAM tries to initialize its coordinate system on ground plane.



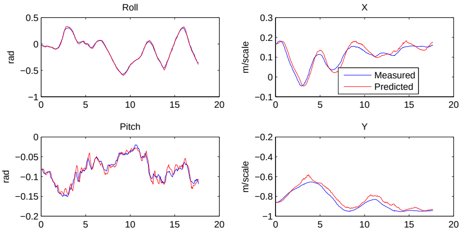
$$\begin{cases} \varnothing_{\text{PTAM}} & = \xi + R(q^{wb}) r_{\text{camera}/\mathcal{G}} + \lambda R(q^{wP}) \frac{X^{PTAM}}{|X^{PTAM}|} \\ \varnothing_{\text{PTAM}} \cdot \hat{z} & = 0 \end{cases}$$

Camera Measurements

To extract positioning measurements usable in the real world, the coordinate systems have to be related.

$$x^{\text{PTAM}} = S(s)R(q^{Pw})T(-\varnothing_{\text{PTAM}})x^{\text{NEDEF}}$$
$$q^{PTAM,c} = q^{Pw}q^{wb}q^{bc}$$

- Describes the measurements in terms of the estimated state and known transformations.



Sensor Fusion

Uses information from all sensors with models of expected behavior to estimate the movements of the vehicle.

The Kalman filter is the standard choice for high-level state estimation.

Non-linear extensions:

- EKF - Extended Kalman Filter
- UKF - Unscented Kalman Filter

The UKF has issues with the applied motion model in high uncertainty simulations. The EKF was selected as the more reliable filter.

Filtering Framework

The standard formulation of a high-level (Bayesian) state estimation framework relies on two separate steps.

Measurement update: Sample the sensors and weigh their likelihood against the current estimate.

Time update: Use the motion model to predict the vehicle's motions.

Through discrete instances of time, the steps are independent.

Motion Model

A nonlinear physical model of the quadrotor is studied in the thesis.

- Detailed physical model.
- Same model is used for state estimation, control and simulation.
- May easily be replaced by simpler model, e.g. for state estimation.

The model also includes sensor-models for all used sensors.

Linear Quadratic control

- Optimal control of a linear motion model.
- Control signal minimizes quadratic criterion, weighing control error against agressive control.

$$\dot{x} = Ax + Bu$$

$$\min_{u=-Lx} \mathcal{J} = \int_0^\infty x^T Q x + u^T R u \, dt$$

Feedback solution, given the solution of the **Riccati Equation**.

$$A^T S + SA + M^T QM - SBR^{-1} B^T S = 0$$

Linear Quadratic control

- Appealingly simple, when you know it.
- Requires a linear motion model, A.

The motions of a quadrotor are nonlinear, but can locally be described by a linear approximation.

- The motion model needs linearization.
- The linearization needs a linearization point.

State-Dependent Riccati Equation

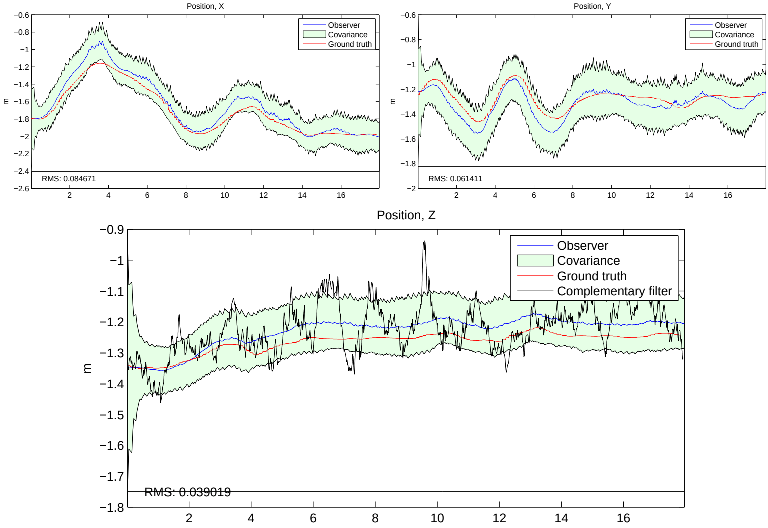
Basic idea: Linearize the physical model and use LQ theory.

$$\dot{x} = f(x,u) \approx f(x_0,u_0) + \underbrace{\left.\frac{\partial f}{\partial x}\right|_{\substack{x=x_0\\u=u_0}}}_{A} \underbrace{(x-x_0)}_{\Delta x} + \underbrace{\left.\frac{\partial f}{\partial u}\right|_{\substack{x=x_0\\u=u_0}}}_{B} \underbrace{(u-u_0)}_{\Delta u}$$

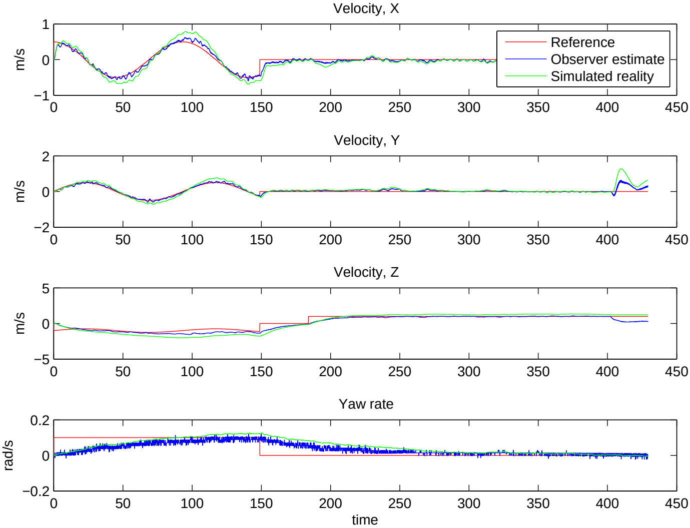
By adding a homogeneous state, the linear property of the equation is regained. The result is locally valid in every differentiable point in the state space.

$$\dot{X} = \begin{bmatrix} \dot{x} \\ 0 \end{bmatrix} = \underbrace{\begin{bmatrix} A & f(x_0,u_0) - Ax_0 \\ 0 & 0 \end{bmatrix}}_{\bar{A}} \underbrace{\begin{bmatrix} x \\ 1 \end{bmatrix}}_{\bar{X}} + \underbrace{\begin{bmatrix} B \\ 0 \end{bmatrix}}_{\bar{B}} \Delta u.$$

Results: Positioning



Results: Control



Conclusions

- General algorithms and control structure were fully implemented.
- PTAM modifications enables full autonomousity.
- Simulated advanced control and landing performed in simulation.
- Implementation covers advanced control.
- Tuning of filtering and control remain.
- Results suggest the system is viable to perform landing.