Thesis reflections

Jonatan Olofsson

December 1, 2011

Contents

1 Introduction

This papar intends to summarize the proposed master thesis in automated landing of the LinkQuad.

1.1 Timespan and resources

The thesis will be executed during the spring semester 2012 at Linköping University, under the Department of Computer and Information Science (IDA). The work will be implemented on a LinkQuad quadrotor, provided by IDA. Examiner for the thesis will be Patrick Doherty at IDA. Thomas Schön (Dept. of Electrical Engineering, ISY), Piotr Rudol (IDA) and Mariusz Wzorek (IDA) will also be assisting in the project.

2 Previous work

Here, previous known similar work is referenced and discussed. Previous work can be used for both reference implemetations, but also to recognize limitations and restrictions posed on the systems studied.
It turns out that the problem of navigating and landing an UAV using visual feedback has been, albeit sparsely,solved. Even fewer have however succeeded using strictly on-board sensors and real-time algorithms running on the limited processing power that standard UAV's of the size of the LinkQuad generally are equipped with. The LinkQuad may, with its distributed processing power, be well suited for a full implementation and solution to the problem.

2.1 Autonomous landing

The problem of landing a quadrocopter can to a large extent be boiled down to achieving good pose estimates using available information. This problem is studied e.g. in [Mellinger et al., 2010; Brockers et al., 2011], but the most interesting results are obtained in [Blösch et al., 2010; Weiss et al., 2011], where the ideas from [Klein and Murray, 2007] are implemented on a UAV platform and excellent results are obtained.
[Brockers et al., 2011] implements a landing control reference scheme which is summarized in 3.3.

2.2 vSLAM

A first take on a vSLAM algorithm is presented in [Karlsson et al., 2005], resting on the foundation of a Rao-Blackwellized particle filter with Kalman filter banks. It might possibly be of gain to use the theory studied in [van der Merwe et al., 2000], which extends the work in [Montemerlo et al., 2002; Montemerlo, 2003] to use a prior distribution which can be tuned for heavier tailed distributions than the standard gaussian[Merwe and Wan, 2004], since it is explicitly noted in [Karlsson et al., 2005] that the measurement errors seem to be drawn from such a distribution.
The algorithm presented in [Karlsson et al., 2005] also extends to the case of multiple cameras, which could be interesting in a longer perspective. An im-

plementation has been made in a ROS project[Oleynikova, 2011], which may be used for reference.
Another source which could prove interesting in the area of SLAM is [Davison et al., 2007], where the MonoSLAM algorithm is introduced and real-time code samples are referenced.
[Klein and Murray, 2007] uses an alternative approach and splits the tracking problem of SLAM from the mapping problem. This means that the mapping can run more advanced algorithms at a slower pace than required by the tracking.
The algorithm proposed in [Klein and Murray, 2007] uses selected keyframes from which offsets are calculated continously in the tracking thread, while the mapping problem is addressed separately as fast as possible using information only from these keyframes. This opposed to the traditional vSLAM filtering solution where each frame has to be used for continuous filter updates.
By for instance not considering uncertainties in either camera pose or feature location, the complexity of the algorithm is reduced and the number of studied points can be increased to achieve better robustness and performance than when a filtering solution is used [Strasdat et al., 2010].
The mapping implementation in [Strasdat et al., 2010] uses the SBA libary [Lourakis, 2010; Lourakis and Argyros, 2009] to compute the Bundle Adjustment step in the mapping algorithm. Again, this is the method on which [Weiss et al., 2011] bases its implementation.

2.3 Filtering

Using the work presented e.g. in [van der Merwe et al., 2000; Julier, 1995; Julier and Uhlmann, 1997] along with the implementation in [Stevens, 2010], a fairly efficient implemetation of the filter exists today from previous projects by the author of this document. The same code could also, with little effort, be used in the vSLAM algorithm.
In a 3D-environment, it is desirable to select an appropriate representation of the orientation, e.g. the quaternion representation. To cope with the restrictions in such a representation, [Ude, 1999] could be studied, for instance.

3 Problem formulation

In this section, the thesis subject is introduced and the problem formulation is presented. After the main goal of the thesis is presented, the section continues with a proposed means of implementation.
Selected for having a large potential and expected payoff in terms of both performance and from an educational standpoint, it is proposed that a vSLAM algorithm is implemented for estimation of absolute generalized pose (position and orientation). This approach have a long-term advantage over the optical flow algorithm previously discussed for the LinkQuad system, and may also possibly benefit from previous work done by on the LinkQuad by [Barac, 2011].

3.1 Main goal

The main goal of the thesis work is to develop and implement an autonomous landing mode for the LinkQuad quadrocopter. The following sensors are available onboard as of today:

- Accelerometers (3DOF acceleration),
- Gyroscopes (3DOF angular velocity),
- Pressure sensor (height above ground, HOG),
- Camera.

Sensors of optical flow have been discussed in the forming of the thesis but since optical flow can be computed from the camera feed, this source will be investigated foremost in the event that optical flow i used in the state estimation. Other sensors have been discussed as well and may be added if possible and if needed.
The problem of landing autonomously can be divided into three main subproblems that all need to be solved for full online autonomous landing:

1. State estimation,
2. Landing site designation / waypoint generation,
3. Control.

With access to the VICON-lab at IDA, it is possible to perform work on the latter two subproblems without the need for fully functional state estimation, thus allowing progress on multiple frontiers at once.
The subsystems should also be made pluggable, so that for instance the landing site designation could initially be implemented in a simple manner, later to be replaced by a more advanced algorithm for finding a suitable landing site.

3.2 State estimation

The problem of estimating the full state of the controlled LinkQuad will be tackled by implementing an unscented Kalman filter ([van der Merwe et al., 2000; Merwe and Wan, 2004; Julier, 1995; Julier and Uhlmann, 1997]) and fusing all available sensor data. The first three of the above mentioned onboard sensors are straightforward to implement, while the camera will require more work and will play a significant role in the thesis.
As the algorithm for the vSLAM, it is proposed that a keyframe solution similar to that of [Weiss et al., 2011] is pursued, as it seems to yield good results to a tractable computational cost.

3.3 Landing site designation / waypoint generation

The problem of finding a suitable landing site for the LinkQuad can, with a control logic suitably interfaced, easily be scaled. Examples of designation schemes are

- Descend-where-you-are,
- Visual marker,
- Virtual marker,

- Surface finding within specified area.

For the landing, following the example of for instance [Mellinger et al., 2010; Brockers et al., 2011], it may be suitable to implement a strategy as follows:

1. Landing site detection
2. Estimate refinement
3. Approach
4. Descend
5. (Recovery)

where the last step brings the helicopter back to a stable hovering state in case the landing failed (e.g. if it falls of the landing platform).
Each of these steps can be implemented as modes in a general mode-selected controller framework where mode transition conditions can be generally implemented.

3.4 Control

The outer loop controller will be implemented as a, possibly scheduled, LQG controller using a linearized dynamic model of the quadrocopter. The controller will be used to follow a trajectory reference as computed given a set of waypoints.

4 Resource plan

This section describes the initial plan on how to use the processing power available on the LinkQuad.

4.1 Available resources

Each LinkQuad is equipped with a LinkBoard. A LinkBoard is in turn equipped with

- Sensor MCU,
- Control MCU,
- two Gumstix® modules running Linux.

4.2 Resource usage

Sensor MCU:

The sensor MCU will be used to sample and compile sensor data from accelerometers, gyroscopes and pressure sensor. If possible, the MCU will oversample the measurements, and at a fixed rate of e.g. 100 Hz send the data to the observer, implemented on one of the Gumstix® modules. The module should also be able to send filtered measurements to the backup control MCU.

Control MCU:

The control MCU is important in the aspect that it acts as a watchdog, should a failure occur with the primary controller on one of the Gumstix® modules. On failure, it reverts control to the lower level stabilizing PID loop, which can then be used to safely manually land the LinkQuad. The control MCU is also used to actuate the higher-level controls given by the controller on the Gumstix® module.

Controller and observer Gumstix® module:

One of the Gumstix® modules will be primarily used for logic, state observation - i.e. run the UKF algorithm - and controller. All modules measuring some kind of sensory percept should report this to the observer which will fuse the information to a single best estimate of the full state. The state can then be immediately used by the controller to stabilize the LinkQuad and follow the reference.

Video processing Gumstix® module:

The second Gumstix® module will be dedicated to the evaluation of the video feed in terms of running the vSLAM algorithm and keeping track of the map. This will require intimate two-way communication with the observer.

References

D. Mellinger, M. Shomin, and V. Kumar. Control of Quadrotors for Robust Perching and Landing. In *Proceedings of the International Powered Lift Conference*, Oct 2010.

Roland Brockers, Patrick Bouffard, Jeremy Ma, Larry Matthies, and Claire Tomlin. Autonomous landing and ingress of micro-air-vehicles in urban environments based on monocular vision. In Thomas George, M. Saif Islam, and Achyut K. Dutta, editors, *Micro- and Nanotechnology Sensors, Systems, and Applications III*, volume 8031, page 803111. SPIE, 2011. doi: 10.1117/12.884449. URL http://link.aip.org/link/?PSI/8031/803111/1.

Michael Blösch, Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. Vision based MAV navigation in unknown and unstructured environments. In *ICRA*, pages 21–28, 2010. URL http://dx.doi.org/10.1109/ROBOT.2010.5509920.

S Weiss, D Scaramuzza, and R Siegwart. Monocular-SLAM based navigation for autonomous micro helicopters in GPS-denied environments. *Journal of Field Robotics*, 28(6):854–874, 2011.

Georg Klein and David Murray. Parallel Tracking and Mapping for Small AR Workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007.

Niklas Karlsson, Enrico Di Bernardo, Jim Ostrowski, Luis Goncalves, Paolo Pirjanian, and Mario E. Munich. The vSLAM algorithm for robust localization and mapping. In *In Proc. of Int. Conf. on Robotics and Automation (ICRA*, pages 24–29, 2005.

Rudolph van der Merwe, Arnaud Doucet, Nando de Freitas, and Eric A. Wan. The Unscented Particle Filter. In *NIPS'00*, pages 584–590, 2000.

M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI)*, pages 593–598, July 2002.

Michael Montemerlo. *FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem with Unknown Data Association*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, July 2003.

Rudolph Van Der Merwe and Eric Wan. Sigma-Point Kalman Filters For Integrated Navigation, 2004.

Helen Oleynikova. ROS vSLAM, September 2011. URL http://www.ros.org/wiki/vslam. [online] http://www.ros.org/wiki/vslam.

Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. MonoSLAM: Real-Time Single Camera SLAM. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29:1052–1067, June 2007. ISSN 0162-8828. doi: 10.1109/TPAMI.2007.1049. URL http://dx.doi.org/10.1109/TPAMI.2007.1049.

Hauke Strasdat, J. M. M. Montiel, and Andrew J. Davison. Real-time monocular SLAM: Why filter? In *ICRA*, pages 2657–2664, 2010.

Manolis Lourakis. SBA, August 2010. URL http://www.ics.forth.gr/~lourakis/sba. [online] http://www.ics.forth.gr/~lourakis/sba/.

M.I. A. Lourakis and A.A. Argyros. SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. Math. Software*, 36(1):1–30, 2009. doi: 10.1145/1486525.1486527.

Simon J | Uhlmann Jeffrey K | Durrant-Whyte Hugh F Julier. A new approach for filtering nonlinear systems. In *1995 American Control Conference, 14th, Seattle, WA; UNITED STATES; 21-23 June 1995*, pages 1628–1632, 1995.

Simon J. Julier and Jeffrey K. Uhlmann. A New Extension of the Kalman Filter to Nonlinear Systems. In , pages 182–193, 1997.

Michael Stevens. Bayesclasses, August 2010. URL http://bayesclasses.sourceforge.net. [online] http://bayesclasses.sourceforge.net.

Ales Ude. Filtering in a unit quaternion space for model-based object tracking. *Robotics and Autonomous Systems*, 28(2-3):163–172, 1999.

Daniel Barac. Localization algorithms for indoor UAVs, 2011.