Hexacopter
Progress Report

Brett Downing

Progress Report
Accurate judgement of progress to date and plan of further progress.

1 Preamble

Multicopters are increasingly popular... Simplicity... Market Saturation... rapidly changing legal situation... Recent Developments in power and control electronics.

Much of the technology related to multicopters is applicable to most other forms of UAV. Agriculture, targeted crop dusting Mapping Cinematography, extreme-sport photography,

2 Where we started

The Hexacopter is a DJI F550 frame with fancy-pants motors and ESCs The 2013 team decided to use a NAZA-V1-lite flight computer. This works well for free-flight, but does not make provisions for way-point navigation or telemetry. The on-board server is currently doing all of the autonomous navigation processing which is appropriate for computer-vision directed flight-modes, but adds additional hurdles to way-point navigation; a problem that is already very well-solved in industrial, hobbyist, and consumer grade drones.

3 What we've achieved

We've implemented and tested the code that we were able to salvage from the previous year-groups, Mathematics that estimates the location of a point of interest based on certain assumptions

We've made a proposal and ordered parts to move to the Ardupilot flight control software running on the Pixhawk flight computer. This move allows us to utilise the vast array of supporting software that the Ardpilot community has written including ground-stations, telemetry loggers, Smartphone apps, Kalman navigation filters and automatic flight control tuning.

Oddly, our tests of the previous year-groups' code have produced better results than are published in the respective papers, however the claims of the previous groups still appear grossly overstated. Various improvements to the hardware including landing gear, Wiring harness, enclosures

Reverse-engineering of circuits used by the previous groups. Having talked to the previous teams, we appear to have generated better documentation about the hardware than the teams had worked with initially.

4 What we intend to do

Object Tracking, chase-cam

4.1 Everything in Its Place

The object tracking code we ported over from the 2014 team took a relatively simplistic approach, feeding the pixel position from the image directly into two PID controllers. This did work, but altitude, camera angle, parallax, and other variables tended to make the controller go from a-bit-weak, to utterly unstable, within seconds.

We've built up a code-base to estimate the physical position of the target in coordinates relative to the copter, trying to sanitize the inputs to the control loop. The control loop now takes inputs of the target coordinates in metres. Our controller is still a collection of basic PID loops, but already, the behaviour is far more consistent in flight.

We expect to improve on the physicality of this, changing the controller outputs to units of metres per second, and estimating latitude and longitude of the target to assist in lost-target recovery (Section 4.2). Logging the estimates of the target's global coordinates is likely to lead to insights into the stability and validity of our work.

4.2 Lost and Found

The copter is already beginning to exhibit strong tracking behaviour but having encountered a camera stability issue (Discussed in 5.1), we've not yet been able to apply aggressive parameters to the chase algorithm. The tracked object frequently leaves the field of view of the camera. We intend to write a search algorithm to attempt to locate an object shortly after it has left the frame.

5 Major Challenges

5.1 Flying on a Steady Cam

As the platform stands, The gimbal does not sufficiently isolate platform motion to stabilise a simple control loop. For example, a forward motion instruction from the chase algorithm causes the copter's rear rotors to throttle up, then the platform pitches forward, then the copter accelerates forward. The pitch data is sent to the camera stabilisation servo which has a slow slew rate and a small delay time. This delay couples pitch motion into the image processing loop and adds a brief but intense upward swing to the location of the target in the image as the copter accelerates. This behaviour is controlled by the physical properties of the copter, the flight-computer's control loops, the servo's controller and the shutter lag on the camera. While all of this is technically possible to compensate for, calibrating against mechanical and electrical properties of a commodity servo and reverse-engineering a third-order control loop introduces a large number of variables and doesn't lend itself easily to empirical validation. Many of these parts were not selected with any great care either and we fully expect to swap out or modify parts of the copter during the course of the project. It makes more sense to solve single whole problems; either cleanly stabilising the camera with a higher performance gimbal, or applying software image stabilisation using motion data from the flight controller to calculate through bulk motions followed by optical flow methods to remove the remaining jitter.

6 Prior Work

Multicopters have received a great deal of attention by merit of their simplicity. A great deal of design work has come out of hobbyist communities such as DIY-Drones and

Their work often lacks scientific rigour, but frequently makes the first foray into experimental hardware.