Optical Flow for Heading Estimation of a Quadrotor Helicopter

John Stowers,* Andrew Bainbridge-Smith, Michael Hayes and Steven Mills
University of Canterbury, New Zealand

Abstract

This paper demonstrates that computer vision techniques can estimate the heading of a small fixed pitch four rotor helicopter. Heading estimates are computed using the optical flow technique of phase correlation on images captured using a down facing camera. The camera is fitted with an omnidirectional lens and the images are transformed into the log-polar domain before the main computational step. The vision algorithm runs at 10 Hz on a single board computer (SBC) mounted aboard the craft. Experimental performance of this system is compared with results obtained from a traditional inertial measurement unit (IMU). It is found that the yaw rate computed from the optical flow is comparable to the IMU and thus appropriate for use in controlling the helicopter.

1 Introduction

Robot control and navigation require motion and attitude sensing mechanisms. Tasks such as object avoidance, path planning and navigation all benefit from ego-motion estimation and motion estimation of the environment. A robust estimator of robot state must provide timely and accurate information in order for a craft to make the necessary control and navigation decisions. Such a system is typically based on a combination of sensors and an estimator which fuses the data into a more accurate estimate.

In this paper we are interested in controlling a small fixed pitch four rotor helicopter called a quadrotor, Figure 1. Unpiloted aerial vehicles, like this, are usually controlled using an attitude estimate obtained from an inertial measurement unit (IMU) consisting of gyroscopes, magnetometers, accelerometers and barometer. However, we are interested in assessing the performance of a vision system in estimating the heading. Our interest in vision is to overcome problems in IMUs, such as drift, which accumulates, and problems in GPS with loss of signal for prolonged periods of time. If it can be shown that vision systems are capable of similar performance then integrating them into the flight control system seems sensible.

Figure 1: Quadrotor airframe, total flying weight 700 g, typical flight time 15 minutes.

A vision system must not only be capable of providing attitude estimates, but be able to provide these in a timely manner to have effective control, and on hardware small and light enough to be mounted on the vehicle. In this paper we will be looking at such a practical system that employs two separate computer systems, as can be seen in Figure 2. The flight control system is an ARM7 microprocessor based system that includes inputs from the IMU. A separate vision processing system, including camera and lens, is implemented on a Gumstix Overo single board computer (SBC). The vision system communicates to the flight control via a USB interface.

*Email address: jrs89@student.canterbury.ac.nz

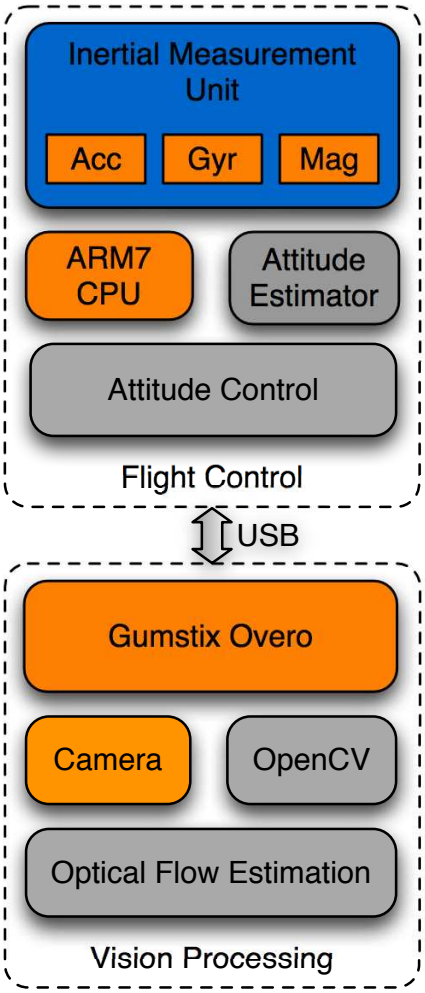


Figure 2: System diagram showing the two quadrotor subsystems, software (grey), hardware (orange) and USB communication between them.

For the purposes of this comparative work we shall limit ourselves to measuring only the components of yaw[1] rate, r in the aircraft body-frame, although the calculations also yield altitude Z, from a downward facing camera fitted with a fish eye lens. We shall present the paper in the following order. Firstly, a discussion of the optical flow process for estimating motion from imagery is given in section 2. We also present the phase correlation technique, which we use for making the optical flow estimate. Secondly, in section 3 we look at transforming the normally Cartesian imagery into the log-polar domain as a means of decoupling two of the important attitude components, heading and altitude, before the phase correlation algorithm is applied. Section 4 presents salient information on the implementation of the quadrotor and its vision system, this includes mechanical, optical, electronic and control information. Results from a limited field trial are given in section 5 with concluding remarks in section 7.

[1]The terms yaw and heading are often used interchangeably. Yaw is a rotation about an object's z-axis, and will be used when describing the output of a specific sensor. Heading refers to the direction the aircraft is facing or tracking. Yaw and heading are coincidental and aligned, subject to calibration, in this aircraft.

2 Estimation of Optical Flow

Optical flow estimation methods measure the velocity vectors perceived from the time-varying image intensity pattern captured by a camera. That is, through the image brightness function they measure the 2D projection of the 3D motion of the world scene in the image plane [1].

Given a brightness function $I(x, y, t)$ at a pixel position (x, y) and time t then the brightness conservation condition can be written as,

$$\frac{\mathrm{d}}{\mathrm{d}t} I(x, y, t) = 0. \tag{1}$$

Since $x = x(t)$ and $y = y(t)$, application of the chain-rule yields

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0, \tag{2}$$

or

$$-\frac{\partial I}{\partial t} = \nabla I \cdot \mathbf{v}, \tag{3}$$

where $\mathbf{v} = (v_x, v_y)^T$ is the image flow velocity.

Fundamentally (3) is an underdetermined problem, for which a partial solution in the direction of the normal of the image spatial gradient can be found,

$$|\mathbf{v}_n| = -\frac{\partial I}{\partial t} \times \frac{1}{|\nabla I|}. \tag{4}$$

This normal image velocity, $\mathbf{v}_n$, embodies the aperture problem [1]. To obtain a full solution of the optical flow second order derivatives of the image intensity function must exist, and this can only be achieved if the aperture is large enough. There exists many potential solutions to this problem and regardless of the solution employed the two fundamental assumptions are:

1. the change in the brightness function (image intensity) is due solely to motion, consequently there will be problems with specular reflection, and,

2. the aperture problem must be solved by considering a patch of pixels in the image.

The phase correlation method for determining optical flow estimates the relative shift between two image blocks by means of a normalized cross-correlation function computed in the 2D spatial Fourier domain [2],

$$\bar{C}_{k,k+1}(u,v) = \frac{\mathcal{F}_{k+1}(u,v)\mathcal{F}_k^*(u,v)}{|\mathcal{F}_{k+1}(u,v)\mathcal{F}_k^*(u,v)|}, \quad (5)$$

where $\mathcal{F}_k(\cdot)$ denotes the Fourier image of $I(x, y, k\Delta t)$ in the spatial frequencies of (u, v). This technique was selected because of it resilience to occlusions, global changes in illumination [2], and because a discrete Fourier transform based implementation has a relatively low computational cost.

If we assume that $I_k(x, y)$ undergoes a linear translational motion of (v_x, v_y) and normalising the time between frames ($\Delta t = 1$) then we are able to compute the displacement (x_0, y_0) of the brightness pattern captured on image sensor, such that $I_{k+1}(x, y) = I_k(x - x_0, y - y_0)$. The Fourier images will be related by a simple linear phase translation, i.e.,

$$\mathcal{F}_{k+1}(u,v) = \exp\left(-j2\pi(ux_0 + vy_0)\right)\mathcal{F}_k(u,v), \quad (6)$$

and in this case the cross-correlation function can be stated as,

$$\bar{c}_{k,k+1}(x,y) = \delta(x - x_0, y - y_0), \quad (7)$$

where the location of the impulse in the cross-correlation space gives the optical flow estimate.

It is informative to note that the optical flow, at all points in the flow field, is proportional to translational 3D motion in the plane of the image sensor. But 3D translational motion in the direction of the optical axis (Z) results in a more complex optical flow field expression,

$$|\mathbf{v}| \propto \frac{\Delta Z}{Z^2}. \quad (8)$$

Hence 3D motion in the direction of the optical axis leads to a dilation or contraction (constant scaling) of the optical flow field. Rotational 3D motion about the optical axis leads to a circular optical flow field offset from the centre of the field (principal point) by any translational 3D motion. Rotational motion in the other axes lead to much more complex optical flow fields, which may be difficult to discern from translational 3D motion [3, 4].

Phase correlation was selected for two reasons. Firstly, there exists high performance implementations of the FFT for the SBC processor in the form of the fftw library [5], and also numerous optimised FFT implementations for the C64 series DSP that is also present, although currently unused, in the SBC processor. Secondly, previous experimental work by the author showed that phase correlation performed the best (compared to other common optical flow calculation methods) when the expected magnitude of image motion, the framerate of the camera, and the computational power of the SBC were considered.

3 Optical flow computation in the log polar domain

In the physical arrangement of our system the camera is mounted in a downward facing direction. Further, in this paper we are interested in measuring both the rotation about the optical axis and translation in the direction of the optical axis. This corresponds to estimating the heading and altitude components of the quadrotor's attitude. As noted, in the Cartesian coordinate system this gives an optical flow field which is the vector sum of a rotational vector field and a dilation vector field. However, if we consider computing the optical flow in the polar coordinate system we are able to decouple these two attitude components [6]. Consider the polar coordinate system (r, α), where r denotes radial distance from optical center (x_c, y_c) and α denotes angle. Any (x, y) point can be presented in polar coordinates,

$$r = \sqrt{(x - x_c)^2 + (y - y_c)^2}, \quad (9)$$

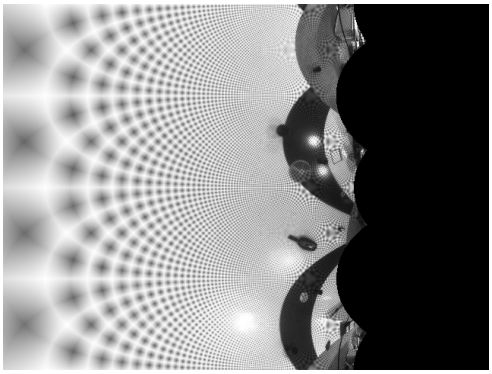
$$\alpha = \tan^{-1}\frac{(y - y_c)}{(x - x_c)}. \quad (10)$$

This transformation maps radial lines in Cartesian space to horizontal lines in polar space [6]. Figure 3 show an image in both its Cartesian space 3(a) and its polar space 3(b).

Application of the cross-correlation function to the polar images $I_k(r, \alpha)$ and $I_{k+1}(r, \alpha + \alpha_0)$, assuming rotation only, will yield an impulse function offset in the y-axis of the cross-correlation space. Similarly a constant image scaling will yield an impulse function offset in the x-axis of the cross-correlation space of the polar images. As noted, a constant image scaling is due to motion in the optical axis, a change of altitude, but the scaling is inversely proportional to Z^2 and consequently the dynamic range over which this motion can be effectively measured can be small. Taking logarithms can improve this situation,

$$\log \Delta r \propto \log \Delta Z - 2\log Z. \quad (11)$$

(a) Original image



(b) After log-polar transformation

Figure 3: Images captured while the quadrotor was in a stable hover above the ground.

Hence the cross-correlation of the log-polar image space, $I(\log r, \alpha)$, is used in practice to determine a change in altitude ($\log \Delta r$) and heading ($\Delta\alpha$).

4 Implementation

The quadrotor system was developed in conjunction with the Paparazzi UAV[2] research group. Work on the flight control hardware was undertaken at ENAC[3] during 2008.

4.1 Mechanical

The quadrotor hardware shown in Figure 1 consists of a cross-frame made of square dural aluminium tubes joined in the center. On this frame are mounted four brushless motor / propeller combinations. The flight control system sits above the center section, while the battery and single board computer sits below.

All control is obtained by manipulating the speed of these propellers. Two opposing pairs of propellers rotate in opposite directions, and are used to create a torque to change the quadrotor heading, yaw. Roll or pitch are manipulated by simultaneously increasing and decreasing opposing propellers. There are no other mechanical parts to control the flight direction and the speed of the quadrotor.

4.2 Optical

Images are captured using a Firefly MV USB connected machine vision camera (figure 5) from Point Grey research. This camera features a 1/3" Micron image sensor with 752×480 pixels and a global shutter, essential for sharp images in high dynamic environments. This camera was chosen due to the availability of high quality Linux drivers that allow efficient image transfer over USB with little CPU usage.

Sharp, motion blur free images are important for optical flow calculation [7, 3] so the images are captured at 640×480 pixels, at 15 times per second. Each image taken with a 15 ms shutter time.

The camera has been fitted with a Sunex DSL219A miniature fisheye lens. This lens has an effective focal length of 1.8 mm, and when mated with the 1/3" image sensor it delivers a $160°$ horizontal field of view image. The camera is mounted to look directly at the ground.

4.3 Omnidirectional imagery

Traditional camera systems have a field of view of approximately $45°$. There have been several methods proposed for expanding the field of view. These methods generally involve either a fixed focus fish-eye lens or a reflective mirror of a distinct shape. The latter are termed catadioptric lenses, and involve placing a hemispherical mirror immediately in front of a planar image sensor, the centre of the mirror aligned with the centre of the sensor.

Stratman [3] et al. and Gluckman [4] et al. both showed that omnidirectional lenses provide a number of benefits over conventional lenses for the computation of optical flow. Larger object surfaces can be perceived, and these remain in the field of view for a longer duration. Both of these properties cause smooth variations in the scene, and facilitate accurate calculation of optical flow.

Sensors with wide field of view make it easier to disambiguate translation motion to that of rotation when the motion is small or in the presence of significant noise. The exaggerated movement of the omnidirectional imagery increases the signal-noise ratio, particularly at high framerates and stable flight, where other techniques would not observe sufficient movement between frames.

[2] http://paparazzi.enac.fr
[3] http://www.enac.fr

4.4 Electronics

Figure 2 shows that the quadrotor features two major electronic subsystems, the hard real-time flight control system, and the soft real-time vision processing system.

The flight control system is based around a 32bit LPC2148 ARM7 micro-controller running at 60 MHz. This processor runs the computationally intensive control and state estimation algorithms. The quadrotor is flown by radio control in stability augmented mode, or autonomously commanded using a remote ground-station.

The ARM7 micro-controller and related hardware are physically arranged in a stacked manner. As shown in figure 4, the top board contains the GPS receiver and antennae, the middle board contains the IMU, and the bottom PCB contains the CPU, power supply, barometer and connectors.

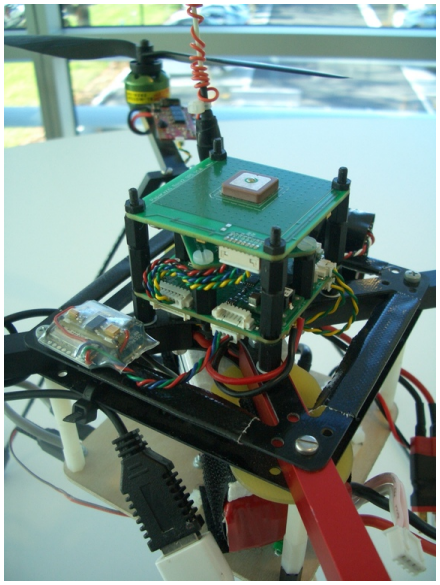


Figure 4: Flight control system, featuring GPS receiver (top PCB), IMU (centre PCB, partially obscured), and CPU (bottom PCB).

Two forms of communication with the quadrotor are present. An XBee 2.4 GHz modem is used for transmission of flight telemetry and commands to the ground-station. A model aircraft radio receiver is used for commanding stability augmented flight using a traditional model aircraft radio transmitter. The receiver signal is analysed directly by the CPU before generating the necessary control signals for the motor controllers - all fight, even manual, is fly-by-wire.

The IMU contains 3 single-axis gyroscopes (ADXR300), 2 dual-axis accelerometers (ADXL320), a 3-axis magnetometer (PNI MS2100) and a barometer (MPX6115). All are sampled at 200 Hz. These sensors all generate errors such as drift, offset and if left uncorrected, a large error in altitude and position estimation can accumulate. Drift and offset in attitude estimates are minimised by using a complementary filter to combine the gyroscope, magnetometer and accelerometer readings [8]. Drifts in position are corrected by using a Kalman filter to combine the GPS readings with the inertial state estimate.

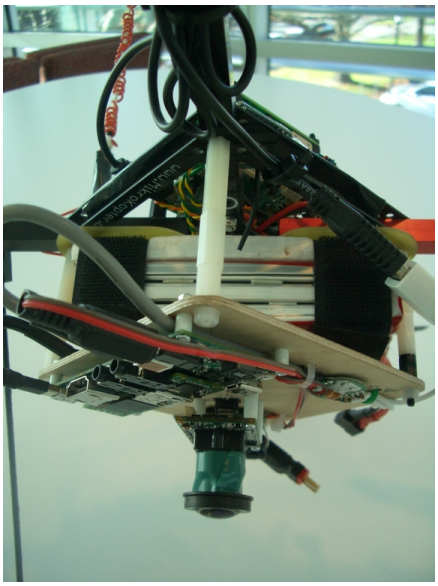


Figure 5: Computer vision system, featuring single board computer (bottom, left) and digital machine vision camera (bottom, right).

The computer vision hardware, shown in Figure 5 uses a Gumstix Overo Earth single board computer (SBC). This features a TI OMAP3503 processor, clocked at 600 MHz and also includes 256 MB of flash and RAM. The SBC is mounted on a Gumstix development board attached to the quadrotor below the battery.

4.5 Image processing

Images are captured from the camera and passed into OpenCV using the libdc1394 library. The image is transformed into log-polar coordinate space, a Gaussian blur is applied and the image is broken into several regions. The phase correlation in each region is computed via the result of a discrete Fourier transform. The discrete Fourier trans-

form is performed using the open source fftw3 library, which provides excellent performance [5].

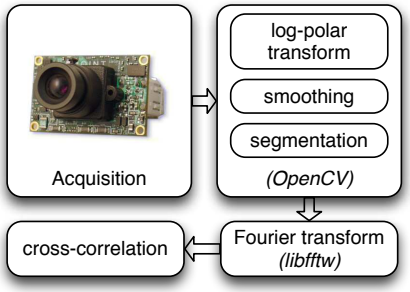


Figure 6: The image processing pipeline, showing the major steps and software used (italics).

Simultainously, the inertial estimate is received over USB from the flight control system. Both the inertial and computed visual yaw results are recorded on a non-volatile SD card.

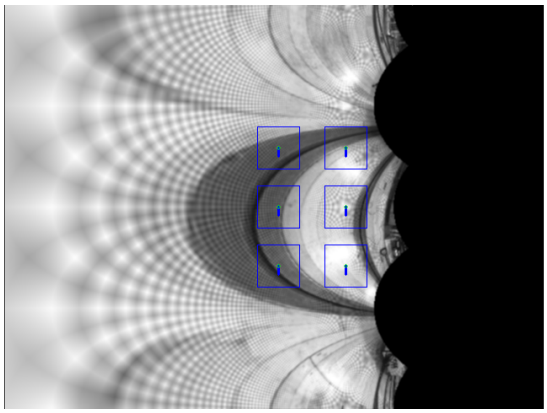


Figure 7: Log-polar image with regions of calculation for phase correlation. The vertical lines, enclosed in boxes indicating the calculated region, represent the magnitude vector of the image motion from the previous frame over that region. The vectors shown in this image have been scaled for readability.

4.6 *Computation of heading and altitude*

The log-polar images are processed on a small number of sub-images or regions for computational efficiency reasons. The image region locations shown in Figure 7 were selected based on the likelihood of significant change of the image data in order for the phase correlation to work effectively. The omnidirectional lens magnified rotational change sufficiently to make this happen.

The attitude (heading and altitude) is obtained by averaging the result—the impulse location, from each region estimate.

4.7 *Control*

The real-time control is derived from the Paparazzi project non-linear back-stepping controller, implemented according to Drouin et al. [9] Of relevance is that this is a rate based, not position based, controller. The control system features multiple control modes where pilot commands sent over the radio control transmitter are transformed into deviations from an equilibrium. All experiments were performed using attitude control mode, the craft is commanded through deviations from zero attitude. This means the craft is stable in hover without pilot input.

5 Results

To calibrate the system the quadrotor was first mounted on a mechanical platform, whose rotational velocity was known and constant, Figure 8(a). The optical flow estimate was calculated online by the SBC at 10 Hz, limited by the computational power of the OMAP3503 processor. The inertial estimate was calculated at 100 Hz by the CPU and sent to the SBC, where it was recorded for comparison with the optical estimate. Comparison of the two techniques was performed offline.

Figure 8 shows the estimated yaw rate of the quadrotor over the flight. The inertial estimate is measured using the rotational velocity obtained from the yaw-axis gyroscope. The visual estimate is obtained from the total Y-displacement of phase correlated samples in the log polar image domain.

The visual estimate is related to the inertial baseline through a constant, whose value was determined by fitting the visual estimate to the inertial measurements taken when the quadrotor was mounted on the rotating platform, Figure 8(a). This constant relates the perceived change in image brightness through equations (4) and (10) yielding the yaw rate ego-motion of the quadrotor.

Following the initial calibration, the quadrotor was then flown manually in a stable hover. Changes in altitude and translation were kept at a minimum while yawing the quadrotor fore and back, Figure 8(b) and 8(c).

The same scaling constant was applied to all results and shows that the visual estimate of the yaw rate corresponds with the actual yaw rate as reported by the inertial measurement system.

6 Future Work

Future work will focus on expanding the technique to estimate the quadrotor altitude, as an estimate of the optical flow in this directions is already calculated in the process of the yaw rate estimation. The most appropriate filter for combining the yaw rate with other sensors, in order to obtain a full heading and attitude estimate will be explored. Further flights will also be undertaken to confirm the relationship between the sequences described in equations (4),(8) and (11).

There is sufficient computational head-room remaining in the SBC and CPU that the yaw and altitude rate measurements computed from the image analysis can be integrated with the existing control loop. Taking yaw rate as an example, further integration may be either of two forms:

1. An improved heading estimate calculated by the SBC. Here, the SBC would combine the reading from the gyroscope with the visual estimate, hopefully producing a measurement that had better drift and noise performance.

2. A direct replacement for the yaw rate measurement provided from the yaw-axis gyroscope.

Similar approaches could also be employed for the altitude estimate; this time replacing the z-axis accelerometer (integrated to position) measurement with the visually computed one.

7 Conclusions

Results obtained in real time, using phase correlation on log-polar transformed images can provide yaw rate estimates for a quadrotor helicopter. It is expected that these real-time estimates could be fed back to the quadrotor control system as an aid, or possibly a replacement, for measurements traditionally obtained from an inertial measurement unit.

8 References

[1] B. K. P. Horn, *Robot vision*. Cambridge, MA, USA: MIT Press, 1986.

[2] F. Pla and M. Bober, "Estimating Translation/Deformation Motion through Phase Correlation," in *Proc. of the 9th International Conference on Image Analysis and Processing-Volume I*. London, UK: Springer-Verlag, 1997, pp. 653–660.

[3] I. Stratmann, "Omnidirectional imaging and optical flow," in *Proc. of the Third Workshop on Omnidirectional Vision*, 2002, pp. 104–111.

[4] J. Gluckman and S. K. Nayar, "Ego-motion and omnidirectional cameras," in *Proc. of the Sixth International Conference on Computer Vision*, Jan 1998, pp. 999–1005.

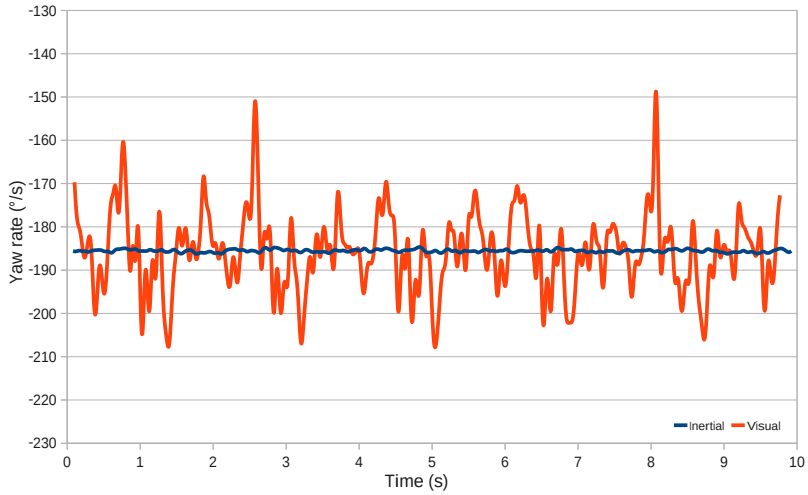
[5] M. Frigo and S. G. Johnson, "The Design and Implementation of FFTW3," *Proc. of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005, Special issue on "Program Generation, Optimization, and Platform Adaptation".

[6] G. Wolberg and S. Zokai, "Robust Image Registration Using Log-Polar Transform," in *Proc. of the IEEE Int. Conf. Image Processing*, vol. 1, 2000, pp. 493–496.

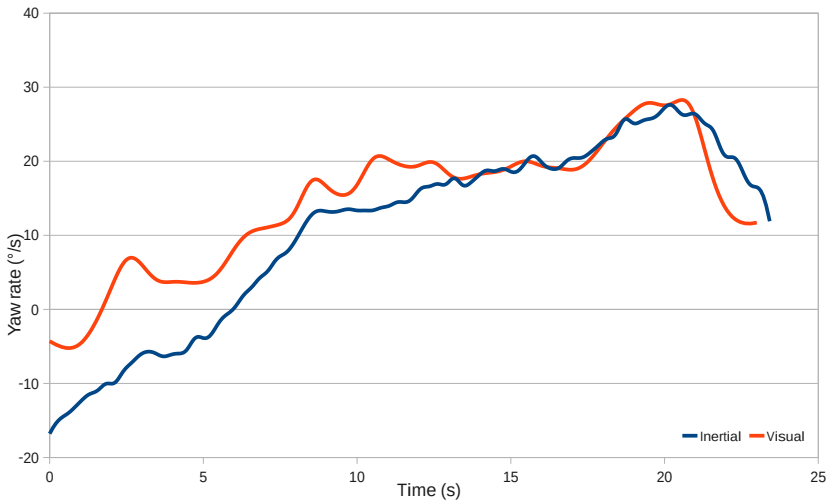
[7] P. J. Myerscough, M. S. Nixon, and J. N. Carter, "Guiding Optical Flow Estimation," in *Proc. of the British Machine Vision Conference*, Norwich, 2003.

[8] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, "A complementary filter for attitude estimation of a fixed-wing UAV," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2008, pp. 340–345.

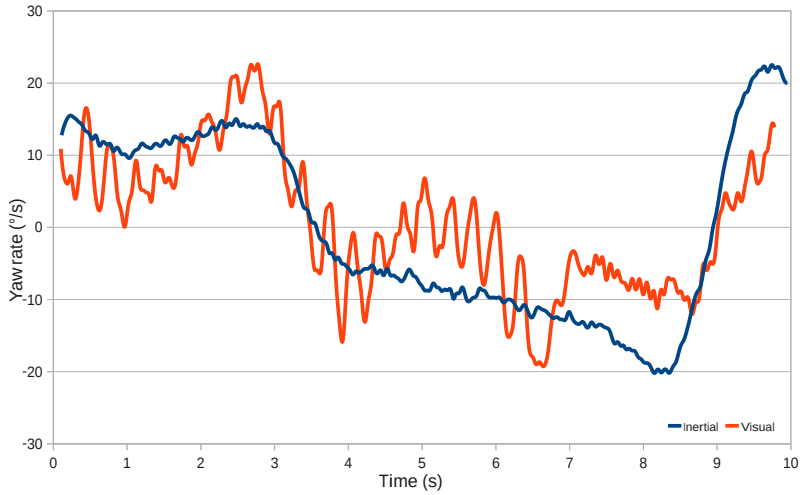
[9] A. Drouin, T. Miquel, and F. Mora-Camino, "Non-Linear Control Structures for Rotorcraft Positioning," in *Proc. of the AIAA Guidance, Navigation and Control Conference and Exhibition*, August 2008.



(a) Quadrotor mounted on rotating platform



(b) Quadrotor flown in stable flight



(c) Quadrotor flown in agressive flight in a busy environment

Figure 8: Estimation of quadrotor yaw rate from inertial measurement unit, and image processing pipeline.