Sensor Based Control of a Quadrotor

A thesis submitted to the Department of Electrotechnical Engineering in partial fulfilment of the requirements for the degree of Master of Science in Electrical and Computer Engineering

Pedro Santos
Instituto Superior Técnico
pedro.lema.santos@ist.utl.pt

***Abstract*— In specific environments such as indoors, absolute positioning sensors - GPS for instance - can't be used and new alternatives should be investigated. This thesis aims to apply the biological concept of optical flow, which is the visual motion of features in an image, to the navigation of an UAV with specific characteristics, the quadrotor.**

Firstly, quadrotor is introduced and depicted, through the study of kinematics and dynamics.

Afterwards, the way how optical flow can be used is described. Nowadays, the solution of complex optical systems with incorporated cameras and vision algorithms to obtain the information about motion is often used. The alternative suggested in this thesis is based on simple sensors with all the processing incorporated. These optical sensors are the basic component of an optical mice, commonly used by everyone on a daily basis. The presented work encompasses the study of the device and development of all communication with such a sensor, involving areas such as electronics and programming of microcontrollers. Given that these sensors are dimensioned for their use in optical mice, an optics study was needed to extrapolate its use to an aerial vehicle, with tremendously bigger working distances to the floor. A prototype was developed and installed on the quadrotor.

The last stage of this thesis comprised the use of the retrieved data, which needs compensation for rotational errors originated by quadrotor's angular rates. The data were included in a closed control loop for position stabilization - hover. The existing simulator was adapted to the needs of this project and a 3D model was developed for real-time visualization of the simulation results.

***Index Terms*— Unmanned Aerial Vehicle, Quadrotor, Optical Flow, Optical Mice, Hover**

I. Introduction

The state of the art in aerial robotics has moved rapidly from simple systems based on RC models, only able to do basic hover or cruise flights using inertial sensors, to robotic vehicles able to navigate and perform simple missions using GPS, vision sensors, lasers, among others.

Nature comes up as a source of inspiration. Flies and insects are the closest thing to aerial vehicles. The manned aircraft developed over the past 100 years rely on a similar lift to the generated by birds' wings. Aircraft designers have paid little attention, however, to the pilot's visual sensor that finely controls these wings, although it is definitely the most sophisticated avionic sensors ever known to exist. The key issues arising in aerial navigation, like flying through cluttered environments and close to obstructions (obstacle avoidance, take-off and landing), are perfectly solved by flying insects. These are agile creatures that navigate swiftly through most unpredictable environments, despite having limited brain processing capacity. Everything humans sought for in the field of aerial robotics is achieved by these creatures: tracking, docking, manoeuvres, collision avoidance, autonomous navigation, etc.

The last decades evidenced that insects guide themselves by processing the optical flow (OF) generated on their eyes as a consequence of their movement. Systems based on the same principle should be able to replicate the smoothness in insects' locomotion. By incorporating similar sensors on aerial vehicles, one expects to accomplish very positive results in the aforementioned fields of navigation.

Involved in the scope of the Advanced Interactive Robotic Tools for the Inspection of Critical Infrastructures (AIRTICI) project funded by AdI, the work reported in this document intended to use an inexpensive optic flow sensor, as the ones in optical mice, to measure translational displacements of an aerial platform, a quadrotor. The focus of this research is comprised of four main smaller objectives.

First step was to attempt to interface the sensor, developing all the necessary libraries to communicate and extract valid data. The second step was to develop a navigation algorithm that would incorporate these experimental sensors to stabilize the vehicle in hover position and, later on, trying to extend the functionality of the sensor to manoeuvres like landing. The third step was to use the available vehicle simulator to incorporate a simulated sensor and test the navigation algorithms. The final step was to assemble the sensor on the vehicle and test it on an indoor environment.

II. Quadrotor

A. Reference frames

Usually it is necessary to use different coordinate systems for several reasons [6], when working with a vehicle of this kind:

Thanks to Jorge Ribeiro, Henrique Silva and André Silva.

- Forces and torques are applied in the body frame, i.e. the coordinate frame attached to the quadrotor, and so are Newton's equations of motion.
- The majority of sensors used in the vehicle provide information with respect to the same frame. An exception to this is a GPS sensor. Its measurements are with respect to the inertial frame.
- Most missions and objectives planned for these vehicles are based on trajectories and paths specified in the inertial frame. Additionally, map information is also described in the inertial frame.

The transformation between frames is given by two basic operations: rotations and translations.

B. Euler Angles

There are many ways of parametrizing rotations in three dimensions, namely Euler angles, rotation vector (or angle-axis) and quaternions ([5], [15]). Each parametrization has its own advantages and disadvantages, discussed in [4]. The first will be described next.

Euler angles are three angles introduced by Leonhard Euler to describe the orientation of a rigid body. They constitute a mean of representing a spatial orientation of any frame, through a composition of rotations from a frame of reference.

In this representation, rotations are made sequentially over the Euler angles, leaving the other two constant. After each partial rotation, the orientation of the original frame is changed. The popularity of such representation is attached to their physical meaning, since Euler angles constitute a direct match to the angles measured by onboard sensors of inertial navigation systems (INS).

It is important to mention two important reference frames for these platforms: inertial and body. Inertial reference frame $\{I\}$ is one in which the motion of a particle not subject to forces is in a straight line at constant speed. Earth can be considered flat and non-rotating due to its size and the distance and duration of flights. The X-axis is chosen to lie in the Northward direction while Y lies East and Z points down. Hence this is an Earth fixed coordinate system. The body reference frame $\{B\}$ is attached to the navigation platform, i.e. the quadrotor, and its origin is at the center of mass of the vehicle. However, the axes of $\{B\}$ are aligned with the axis of $\{I\}$.

The orientation or attitude of the rigid body, i.e. the quadrotor, is defined by $\boldsymbol{\lambda} = [\psi, \theta, \phi]'$, where yaw ($\psi$), pitch ($\theta$) and roll ($\phi$) describe rotations around Z, Y and X, respectively, as shown in Fig.1.

When performing Z, Y and X rotations sequentially, the geometric transformation from $\{I\}$ to $\{B\}$ is a compose rotation and is given by

$$ {}^B_I\mathbf{R} = {}^B_{F2}\mathbf{R}(\phi)\,{}^{F_2}_{F_1}\mathbf{R}(\theta)\,{}^{F_1}_{I}\mathbf{R}(\psi) \tag{1} $$

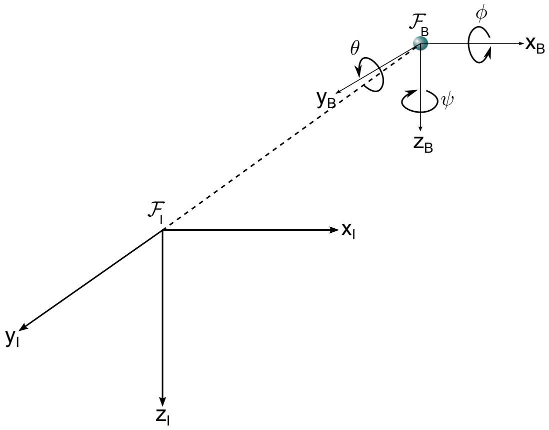


Fig. 1: Inertial to body reference frame transformation.

where $\{F_1\}$ and $\{F_2\}$ are intermediate frames. Following the inverse property

$$ {}^I_B\mathbf{R} = ({}^B_I\mathbf{R})^T = {}^I_{F_1}\mathbf{R}(\psi)\,{}^{F_1}_{F_2}\mathbf{R}(\theta)\,{}^{F_2}_{B}\mathbf{R}(\phi) \tag{2} $$

Replacing each rotation by the corresponding matrix and performing the product gives Eq.3:

$$ {}^I_B\mathbf{R} = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & s\psi s\phi + c\psi s\theta c\phi \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{3} $$

There are many possible representations using Euler angles [16], depending on the order in which one performs the rotations. The one described so far is named Z-Y-X, for obvious reasons.

It is possible to extract the Euler angles Z-Y-X through the entries of the rotation matrix.

C. Dynamics & Kinematics

The expressions derived in this chapter are general to any rigid body, but the notation and coordinate frames used are specific to the quadrotor.

Consider $\boldsymbol{p} = [x, y, z]^T$ as the quadrotor's position in $\{I\}$, $\boldsymbol{v} = [u, v, w]^T$ as the linear velocity of the vehicle expressed in $\{B\}$, $\boldsymbol{\lambda} = [\theta, \phi, \psi]^T$ as the orientation given by the euler angles with respect to $\{F_1\}$, $\{F_2\}$ and $\{I\}$, respectively, and $\boldsymbol{\omega} = [p, q, r]^T$ as the angular velocity expressed in $\{B\}$.

The kinematics of this vehicle relate state variables on different coordinate frames. Since 6 independent coordinates are necessary to determine the position and orientation in space, the relationships between them are described with 6 DOF (degree of freedom) kinematic equations. Each motion DOF is an entry of Table I.

The state model used for this vehicle is composed by the following equations, which relate all the aforementioned variables.

DOF	Linear		
1 2 3	motion in the x-axis direction motion in the y-axis direction motion in the z-axis direction	u v w	x y z
DOF	Angular		
4 5 6	rotation about the x-axis (roll) rotation about the y-axis (pitch) rotation about the z-axis (yaw)	p q r	ϕ θ ψ

TABLE I: 6-DOF Motion

$$\begin{aligned}
\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} &= {}^{I}_{B}\mathbf{R} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \\
\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} &= \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} + {}^{B}_{I}\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + \frac{1}{m} \begin{bmatrix} 0 \\ 0 \\ f_z \end{bmatrix} \\
\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} &= \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi\sec(\theta) & c\phi\sec(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \\
\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} &= \begin{bmatrix} \frac{J_{YY}-J_{ZZ}}{J_{XX}} qr \\ \frac{J_{ZZ}-J_{XX}}{J_{YY}} pr \\ \frac{J_{XX}-J_{YY}}{J_{ZZ}} pq \end{bmatrix} + \begin{bmatrix} \frac{1}{J_{XX}} \tau_{\phi} \\ \frac{1}{J_{YY}} \tau_{\theta} \\ \frac{1}{J_{ZZ}} \tau_{\psi} \end{bmatrix}
\end{aligned} \tag{4}$$

where $\boldsymbol{\tau} = [\tau_{\phi}, \tau_{\theta}, \tau_{\psi}]^{T}$ is the applied torque and

$$\mathbf{J} = \begin{bmatrix} J_{XX} & J_{XY} & J_{XZ} \\ J_{YX} & J_{YY} & J_{YZ} \\ J_{ZX} & J_{ZY} & J_{ZZ} \end{bmatrix} \tag{5}$$

is the inertia matrix. Giving that the adopted approximation for the calculation of this matrix is a sphere with 4 point masses at in each of the rotors, one concludes that

$$\begin{aligned}
J_{XX} &= \frac{2MR^2}{5} + 2ml^2 \\
J_{YY} &= \frac{2MR^2}{5} + 2ml^2 \\
J_{ZZ} &= \frac{2MR^2}{5} + 4ml^2
\end{aligned} \tag{6}$$

The remaining entries are null, due to symmetry.

D. Forces and Moments

A more general approach is commonly necessary in order to better understand how to maneuver the quadrotor. It has four identical rotors and the propellers have a fixed angle of attack. The opposing rotors and paired and each pair rotates in a different direction. In other words, two of the propellers rotate in a clockwise direction and the other two spin counter-clockwise.

Each propeller produces a force, usually named thrust, and a torque, both of which increase with the speed of rotation.

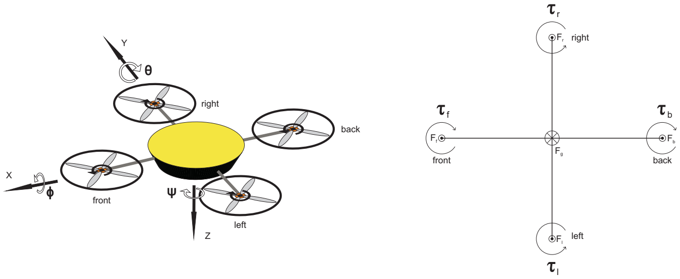


Fig. 2: On the left: perspective view of the quadrotor, with the positive rotations shown for pitch, roll and yaw, as well as the direction of rotation of the propellers. On the right: top view diagram of the quadrotor, with all the torques and forces that actuate on the frame of the vehicle.

Newton's Third Law, which expresses the relation between the mutual forces of action and reaction between two bodies, is what governs the motion of the quadrotor. The propellers apply a force to the air accelerating it downwards. The reaction of the air applies an equal and opposite force on the propellers accelerating them upwards. The same can be transposed to torques: motors exert a torque to the propellers and the propellers provide an equal and opposite torque to the motor. Note that the motors are rigidly attached to the frame of the quadrotor. Thus, the reaction torque is applied directly to the frame. This description is depicted on Fig.2.

There are four non-redundant control inputs, meaning that the quadrotor is an underactuated vehicle. These are:

- **Thrust - F**
  The thrust input results from the sum of the forces produced by each of the rotors, so that

$$\boldsymbol{T} = \boldsymbol{F}_f + \boldsymbol{F}_b + \boldsymbol{F}_l + \boldsymbol{F}_r \tag{7}$$

  In the body frame, the total force acting on the vehicle is given by

$$\boldsymbol{F} = {}^{B}_{I}\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} \tag{8}$$

- **Roll Torque -** τ_{ϕ}
  Torque inputs result from the force differential between two rotors on the same axis.. To create a positive roll torque, the left propeller's speed should increase while the right one is decreased by the same amount. On the contrary, if the intended goal is a negative torque, the right one should increase while the left one is decreased.

$$\tau_{\phi} = l\,(F_l - F_r) \tag{9}$$

- **Pitch Torque -** τ_{θ}
  Similarly, the pitching torque is produced by balancing the forces of the front and back motors, as follows

$$\tau_{\theta} = l\,(F_f - F_b) \tag{10}$$

- **Yaw Torque -** τ_{ψ}

As described previously, the yawing torque is controlled by all the motors. The net thrust and pitch/roll torques can be balanced if the equalities of Eq.7,10,9 are kept constant, i.e., increasing (decreasing) the remaining motors' speed by the same amount while decreasing (increasing) the rest in the same proportion. This creates a positive (negative) yaw torque, making the quadrotor turn around itself.

$$\tau_\psi = c\,(F_f + F_b - F_l - F_r) \tag{11}$$

In summary, the thrust and torque inputs are given by

$$\begin{bmatrix} T \\ \tau_\theta \\ \tau_\phi \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ c & c & -c & -c \end{bmatrix} \begin{bmatrix} F_f \\ F_r \\ F_b \\ F_l \end{bmatrix} \tag{12}$$

E. Aerial Platform

The quadrotor used for this project is shown in Fig.3.

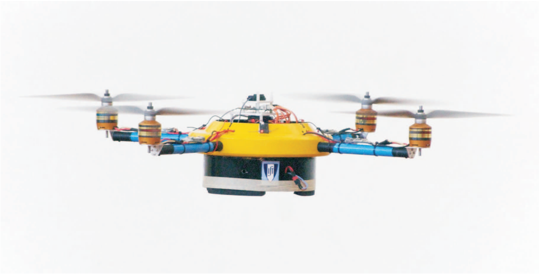


Fig. 3: IST Quadrotor

III. Optical Flow

A. Concept

Applied to Nature, optical flow refers to the apparent movement of textures in the visual field of an insect, while moving. This information can be used to perceive depth and estimate the proximity to objects. On the one side, while looking at the downward direction, the insect estimates its speed from the observed optical flow. Faster optical flow indicates lower altitudes (assuming the insect's velocity doesn't change much). On the other side, obstacle detection can be made by detecting expansion or divergence in the visual field. This is depicted on Fig.4, where a small representation of an insect moving towards a tree is shown.

The downward view is useful for tracking purposes and terrain following. Notice the sudden increase of the field while passing above a stone in the ground.

More rapid expansions in the forward view imply a closer proximity to the obstacle. Usually the origin of the field is called the focus of expansion (FOE) and, if located inside a rapidly expanding region, a collision is imminent. On the contrary, if the FOE is located outside the expanding region, then the insect will move without colliding. These are the principles behind the behaviours of a flying insect, using optical flow.

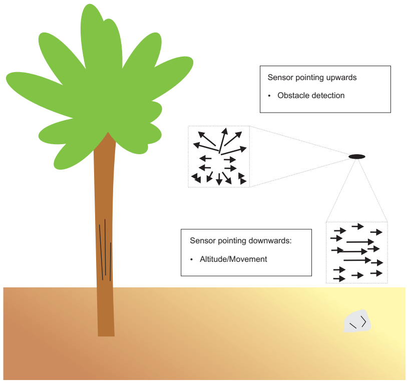


Fig. 4: Visual sensing of an insect

B. Navigation and Flight Control

The question is now how insects use the optical flow field to navigate and control the flight in a stable way. [7] describes a study made and states that the adopted behaviours based on the information received by the sensors are quite simple. Tasks are performed using stratagems which can be seen as reflexes, making them appropriate for control loops.

Some of the described stratagems are:

- **Centering Response -** Two fly a collision-free path between two objects, equalize the optical flow measured on both sides. This heuristic ensures that, for instance, inside a tunnel, an insect flies at the center, maintaining the same distance to the left and right walls.
- **Landing Strategy -** Keep the downwards optical flow constant and the forward speed proportional to the vertical speed. Decreasing the forward speed proportional to the height guarantees a constant downwards optical flow and perfect landing, since when the height approaches zero, so does the flight speed.
- **Hovering Strategy -** Cancelling the image velocity everywhere allows the insect to hover. With zero velocity relative to the objects, the optical flow field is null. If the objects are stopped, so is the insect.

The measurement of optical flow is usually done using a camera. It is a popular subject of investigation, known to be a computationally intensive problem, like many of image processing algorithms. Only in the late few years processors have become fast enough to overcome the overload issues, not jeopardizing the efficiency of the system with the time taken to compute the algorithms.

Another way of calculating the flow has recently popped up, by using electronic microsensors available in optical mice.

C. Sensor - Avago ADNS5050

Avago's ADNS5050 (IC) is commonly used in optical mice. It is a cheap, common and lightweight device, based on a CMOS technology for capturing images, at a maximum rate of 4500 frames per second. It measures changes in position (displacement) by acquiring sequential surface images and determining the direction and magnitude of movement, using internal procedures.

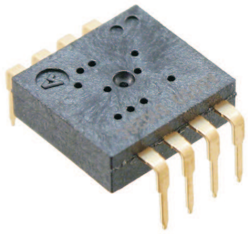


Fig. 5: (a) Avago ADNS5050.

Electronically, the sensor is programmed via registers through a three-wire Serial Peripheral Interface (SPI). It has a set of internal registers which can be read from or written to, through a synchronized transmission over the SDIO line. The synchronization is obtained with the clock line, shared between the sensor and the other devices.

The sensor allows one to extract several types of data, among others:

- Displacement both in x and y coordinates;
- Measure of image quality, in terms of features, named SQUAL;
- Minimum and maximum light intensity in the image; and
- Frame captured by the sensor.

As the internal registers are all 8-bit, the extracted values are in the interval [-128, 127], since they are represented in two's complement.

The optical flow provided by the sensor in both x and y directions is a single value. This means that there is no information about the optical flow field of the frame taken at a certain instant, but rather a mean value of the field in every pixel.

It is important to understand what information about the displacements is extracted. Displacements are measured in cpi (counts per pixel), i.e., the amount of pixels that, in average, the traceable features moved. The sensor accumulates measurements until a read operation is made and the internal registers are cleaned. For control purpose, the optical flow can be converted into a concrete physical value, other than cpis. For this, the notion of field of view has to be introduced. It corresponds to the angular extent of the observable world seen by an optical sensor, whether it is an eye, a camera or any other type of device. On Fig.6, it corresponds to the angular opening which encompasses the whole square image. The field of view can be seen individually for each dimension: there is an horizontal field of view and a vertical field of view, whether one considers x-axis or y-axis.

Assuming a measured displacement in both directions between two time intervals, given by Δx and Δy, the conversion to radians is made by simple applying the following expression

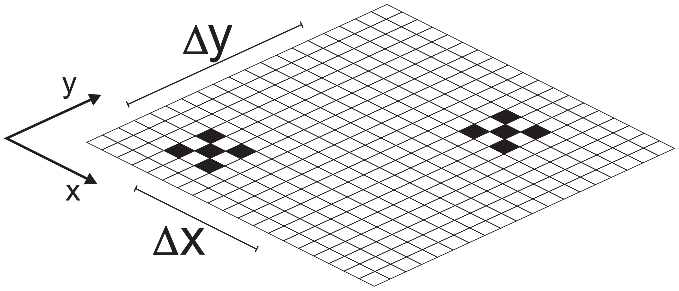


Fig. 6: Converting cpi displacement in radians, by means of the field of view of the sensor

$$\theta_i = \frac{\Delta i}{n_p}\alpha \tag{13}$$

where i is the defined direction (x or y) and α is the field of view in that direction. n_p is the total number of pixels in the desired direction. The angular movement of the sensor correspondent to the detected feature movement in one direction is a fraction of the total field of view in that same direction. Knowing the latter, the first can be determined by the values given by the sensor.

D. Interface

The interface with the sensor was achieved with an ATMEL90CAN128, a microcontroller from ATMEL. It is a low-power CMOS 8-bit microcontroller. It has a considerable number of features like the use of general purpose I/O registers, timers, two USARTs and an SPI serial port.

The microcontroller was programmed to periodically extract data from the sensor.

In order to communicate, the program running in the microcontroller has to be responsible for the direct control of the communication lines (SDIO, SCLK and NCS), as well as the required timings between clock changes. This software solution is called bit banging. In this, all the synchronization, changes and sampling of the communication lines have to be contemplated on the program running in the microcontroller. It is slower than other hardware solutions (usually interrupt-driven) since it is based on an active wait (processor is blocked until something is transmitted/received). It is also more susceptible to noise in the lines.

The protocol necessary to successfully communicate with the sensor is present in [20].

The architectural overview of the system can be seen on Fig.7.

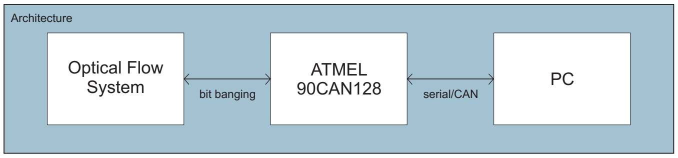


Fig. 7: Overview of the system's architecture

E. *Prototype*

The know-how in optical flow has been extended over latest few years. Pelican quadrotor, from Chemnitz UT, aims at using optical flow as odometer fused with Microsoft Kinect for indoors navigation, with position and velocity control. In Brigham Young University a flying wing airframe was developed and uses optical flow outdoors, for high altitude flights. Another approach - called Optipilot - uses these sensors for landing and take-off control of a swinglet. Despite the existing major differences, this set-up was firstly adopted in this project. It didn't meet the desired requirements and, as such, a new lens was required to increase the working distance.

To decrease the field of view and, consequently, increase the magnification (zoom) of the image, the focal distance should be bigger. However, this means that the amount of light that is captured is reduced and these sensors require a big amount of light to properly work. The increase of the focal distance should then account for this.

A set of lens, with a huge variety of focal distances, is available at Lensation. These M-12 mount lenses are ideal to be used in these projects, as they are lightweight and small. To comply with the necessary requirements, a new lens was incorporated in the optical system - a BT16020 ($f = 16mm$) - with a proper adapter. Additionally, a red 1W Alustar LED module was also acquired. This module has an array of high-intensity red LEDs and incorporates the mount and mirror.

To validate the new optical system, the platform was assembled on the top of a terrain robot, a Pioneer 3-AT. The test consisted in a forward straight movement followed by the reverse movement. The initial and final position should coincide. Two different sample rates were tested (5ms and 30 ms), to check its influence on the extracted measurements. The sensor's height was bigger than in the previous tests.

The position in i-axis of the vehicle at time instant k is given by

$$p_i(k) = \int_0^k \Delta i(k)\, dk, \qquad i = x, y \tag{14}$$

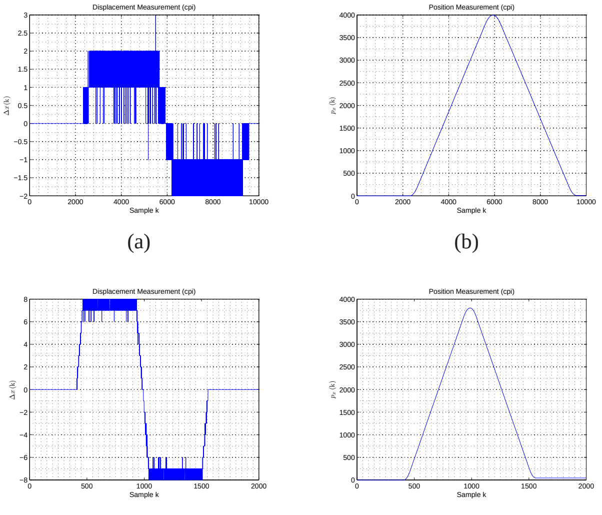
The displacements in x and the position obtained from Eq.14 are shown for both sample rates.

Despite the sampling rate difference, the outcome of the experiment was practically the same.

In terms of quality, both tests had very good results of SQUAL over time, as presented in Fig.9.

The light conditions are of utmost importance. When the LED isn't pointing towards the area targeted by the sensors, the SQUAL value decreases radically and the displacements accumulate a considerable error. By taking a look at an optical mouse, it is straightforward to understand the importance of light, since there should be a reasoning behind the fact that the red LED is fully concentrated below the mouse.

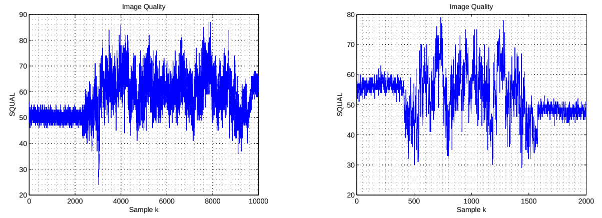
These results were the gateway to the final prototype - Fig.10.



(a) (b)

(c) (d)

Fig. 8: (a) Raw displacement data at a sample rate of 5ms. (b) Position at a sample rate of 5ms. (c) Raw displacement data at a sample rate of 30ms. (d) Position at a sample rate of 30ms.



(a) (b)

Fig. 9: (a) SQUAL at a sample rate of 5ms. (b) SQUAL at a sample rate of 30ms.

IV. Control

Hovering is one of the most challenging processes when controlling a helicopter, such as the quadrotor. The air flux originated by the rotors act against the vehicle's frame, so there is a constant need for adjusting and correcting the control inputs to keep the hovering position.

Given the degrees of freedom of the quadrotor, the optical flow measured by the sensor will have two components: translational and rotational. For hover control, i.e., immobilizing the quadrotor in the air, only the translation component is important. Small rotational movements should have no impact on the response of the vehicle.

A. *Optical Flow Compensation*

The measured optical flow can then be described as

$$\Delta i = \Delta i_{translational} + \delta i_{error} \tag{15}$$

where $i = x, y$.

While the translational component is due to the linear movement of the vehicle relative to the ground, the error

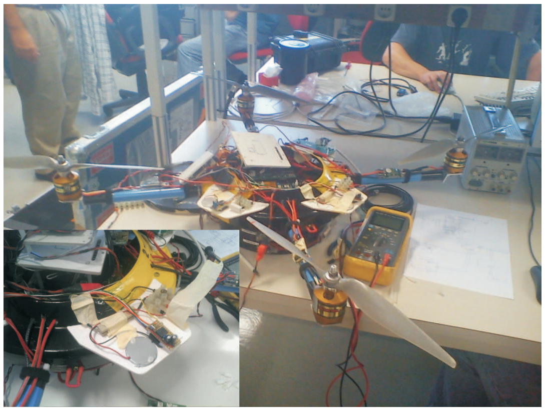


Fig. 10: Quadrotor with the sensor incorporated

comes from angular rates. Given that the sensor is attached to the frame of the quadrotor, rotations around the x-axis and y-axis influence the measured optical flow.

The translational component is directly related to the linear velocity of the vehicle, parallel to the ground. Although the obtained displacement is a measure of motion, it is ambiguous about absolute velocity. The sensor in fact measures a ratio between velocity and height above ground. Consider a angular displacement θ as the one described in Fig. 6, obtained by converting sensor data into radians. Applying simple trigonometric equations, the translational component is given by

$$\theta = \frac{v_g \Delta t}{h_{ag}} \tag{16}$$

where v_g is the velocity relative to the ground, Δt is the sampling time and h_{ag} is the height above ground. The same translational optical flow can be obtained for different velocities, as long as the ratio between velocity and height remains the same.

Besides the ambiguity in velocity, there is also an ambiguity related to the type of movement of the vehicle.

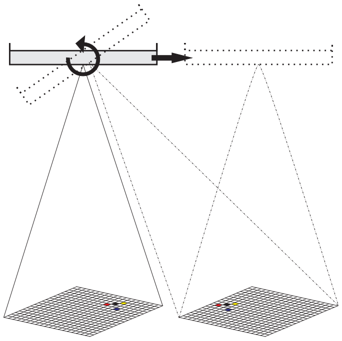


Fig. 11: Two distinct movements - translational and rotational - can result in the same optical flow measurement.

In Fig.11, it is clear how the same optical flow can be obtained by two completely distinct manoeuvres. A rotation can get similar results to an horizontal displacement. However, as the objective is to control the position of the quadrotor, maintaining hover, one has to separate both components and use that information in the feedback loop.

The rotational motion can be effectively compensated for using the angular velocity $\boldsymbol{\omega}$ measured by the inertial measurement unit (IMU). The error added by rotational movements is given by

$$\begin{cases} \delta x_{error} = \left(\frac{n_p}{\alpha}\right) \omega_y \Delta t \\ \delta y_{error} = \left(\frac{n_p}{\alpha}\right) \omega_x \Delta t \end{cases} \tag{17}$$

Note that the IMU data is given in the body frame. The sensor isn't aligned with the body frame for structural issues, so the information can't be fused without transforming one frame into another. This can be done in a simple way. Consider a generic angle β, which corresponds to the angular shift between the sensor reference frame and the quadrotor body frame, as depicted in Fig.12.

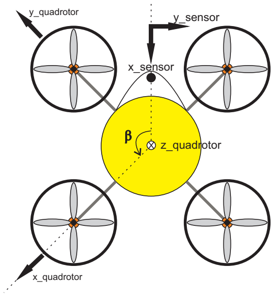


Fig. 12: Difference between the quadrotor's body frame and the sensor reference frame.

The transformation of the displacement data is then given by two rotations. First, a rotation of π radians around y is made, which implies a signal change in the measured x-displacements. Afterwards, a rotation of β radians around z completes the transformation.

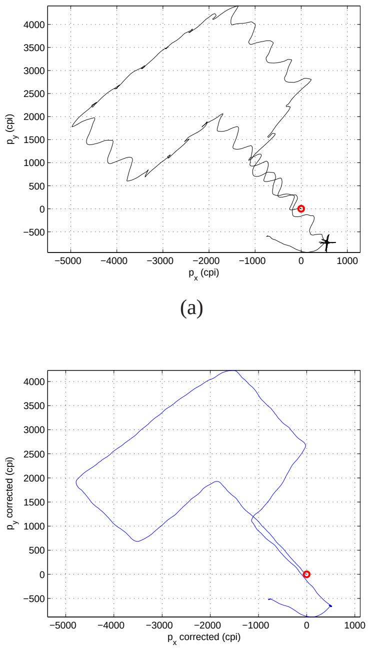
$$\begin{bmatrix} {}^{B}\Delta x \\ {}^{B}\Delta y \\ 0 \end{bmatrix} = \mathbf{R}_z(\beta)\mathbf{R}_y(\pi) \begin{bmatrix} {}^{S}\Delta x \\ {}^{S}\Delta y \\ 0 \end{bmatrix} \tag{18}$$

After this transformation, the data extracted from the sensor is aligned with the body frame of the quadrotor. All the calculations are now coherent, since they are relative to the same reference frame.

To validate the aforementioned corrections, a test was performed on Instituto Superior Técnico's campus. With the motors turned off and an approximate constant height of 1.5 meters, the path started on the marked red circle. A test involved a linear path, but with oscillatory movements in pitch and roll, maintaining the heading. At the end, with a stationary position, consecutive roll and pitch variations were forced, individually. Besides the data extracted from the sensor, the onboard IMU was also used.

Since no moving platform was used and these tests were totally handmade, accuracy is not flawless. For instance, there were some height variations which could eventually lead to a certain bias in the outcome. Remember that height influences the sensitivity of the sensor, in the sense that the bigger the distance to the detected features, the less the optical flow values obtained (for constant velocity). However, the objective was to validate the sensor compensation.

Results presented in Fig.13 are relative to the body frame of the quadrotor.



(a)

(b)

Fig. 13: (a) Raw position. (b) Compensated position.

The second test is definitely more aggressive, since the attitude is constantly changing. The compensation is really effective in this case, almost totally erasing any rotational effect in the optical flow measurements. This is reflected on both position plots shown (Fig.13a and 13b). One should also add that the IMU data is a bit noisy, which leads to some compensation miscalculations. Nevertheless, it is still visible that the corrected position in the linear path test is better than the raw one.

B. Simulator

A simulator of the quadrotor was developed by the DSOR team. This model encompasses three main modules. The first is the quadrotor's model, which respects the dynamics described in the second chapter of this thesis. An RC command is also simulated and constitutes the second module. The last module is related to the control of the vehicle, running as an interface between the RC command and the quadrotor's model.

C. Optical Flow Controller

The next step is to define the process of incorporating the optical flow data in the system, in order to stabilize the quadrotor in hover position.

The concept of stabilizing the quadrotor resides on the control of the thrust vector. If the quadrotor moves left, for instance, the thrust vector should be such as to create a right horizontal force which opposes to the vehicle's velocity, while vertically balancing the gravitational force. This situation is depicted on Fig.14.

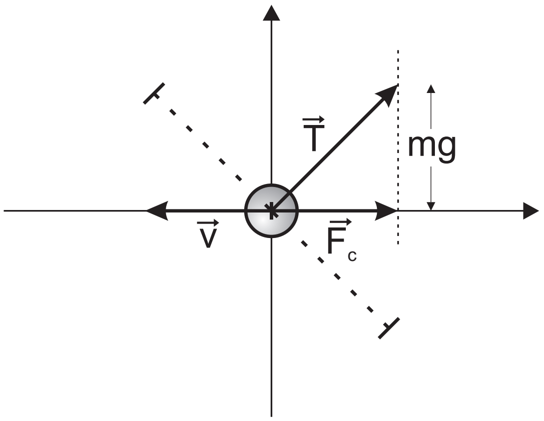


Fig. 14: Method for controlling the quadrotor.

The desired control force $\mathbf{F}_c$ in the body frame is then

$$ {}^{B}\mathbf{F}_c = -K_d {}^{B}\mathbf{v} \tag{19}$$

This force has two components, x and y. The velocity measured in the body frame along x and y is proportional to the measured optical flow (scaled by the sampling period and a unity conversion gain). Hence, the desired control force is proportional to the measured optical flow.

An attitude controller is already running in the quadrotor, developed by the DSOR team. One indicates the desired pitch, roll and yaw angles and the system reacts accordingly. There is also an altitude controller which stabilizes the quadrotor in respect to its height above ground.

It is possible to adapt the current system to the objective proposed of hover control. Imagine the quadrotor is moving at a constant positive speed v_x, relative to the body frame. By increasing the pitch angle, an horizontal backwards force in x is created, decelerating the vehicle - the same doesn't apply to v_y and roll. As z-axis points down, a positive roll tends to increase the y-component of ${}^{B}\mathbf{F}_c$. The idea is that instead of controlling forces through optical flow, the measured data can be fed into a loop as attitude deviations, as shown in Fig.15.

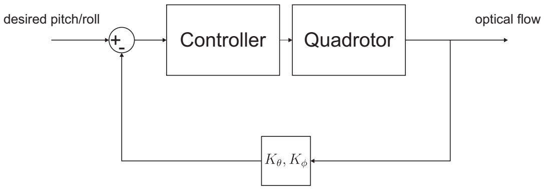


Fig. 15: Implemented proportional controller.

The gains differ in sign. For the reasons described above, K_θ is negative and K_ϕ is positive.

The process of using optical flow can then be described as:

- At time instant k, extract displacements in x and y, as well as the $SQUAL$ value;
- Compensate rotational motions, by incorporating data from the IMU;
- Convert the translational optical flow in pitch/roll angles;
- Feed the RC Remote block with the desired angles.

The proposed method works as a guidance process supported by the attitude controller running on the vehicle. Fig. 16 shows the simulation made using the described feedback process.

The controller was turned on/off four times, which explains the steps seen on the results. The idea was to let the quadrotor gain speed on both x and y (with different) rates and, periodically, let the controller actuate, to see how the system reacted.

The influence of the gains is perfectly clear by inspecting the shown plots. When $K_\theta, K_\phi = 5$ the system is very slow while stabilizing the position of the quadrotor. The increase of both gains improves the performance of the system, in the sense that it is quicker to react and stop the quadrotor's motion. However, if the gains are too high, one can be creating an oscillatory behaviour around the stabilized position. For $K_\theta, K_\phi = 25$, that is what happens.

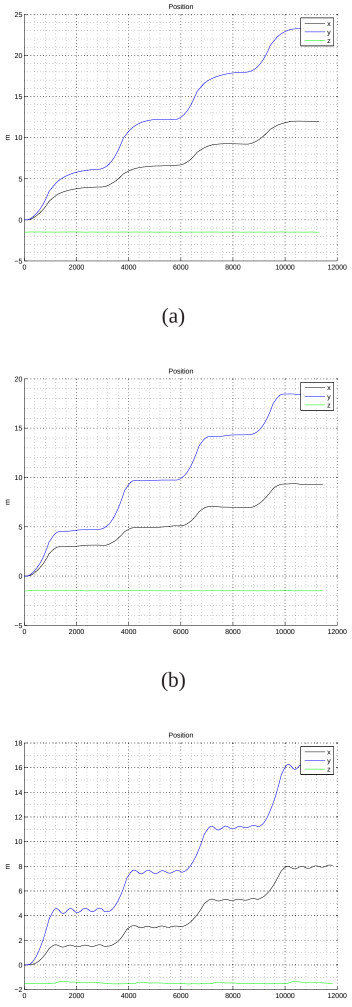
D. 3D Model

The simulator lacked a way to visualize the simulation results in real time, besides the signal processing that could be done afterwards. For this reason, a 3D representation of the quadrotor was included in the Simulink model. Besides that, a Joystick Input block was also added, to engage the real scenarios. This block reads USB ports and automatically extracts all the information about the axis and buttons. It is only necessary to select the desired entries and use them at will. By connecting the correct axis to the remote RC block, the quadrotor can be freely guided by a human operator in the simulation, while visualizing the changes in a 3D representation. Interpreting the outcome of the process becomes easier to the human eye. Fig.17 shows the aforementioned 3D representation.

E. Multiple Sensor Fusion

Considering the use of more than one sensor, the matter of fusing the data arises. As stated previously, these sensors also provide the quality measure - SQUAL - of the image for the user, which is directly related to the uncertainty of its information.

There are several approaches for this problem, one of each is to chose, from the set of working sensors, the one who presents the highest quality factor at a certain time instant. Alternatively, a weighted average can be computed. Considering a general case with N sensors, the displacement estimate in i direction at instant k is given by:



(a)

(b)

(c)

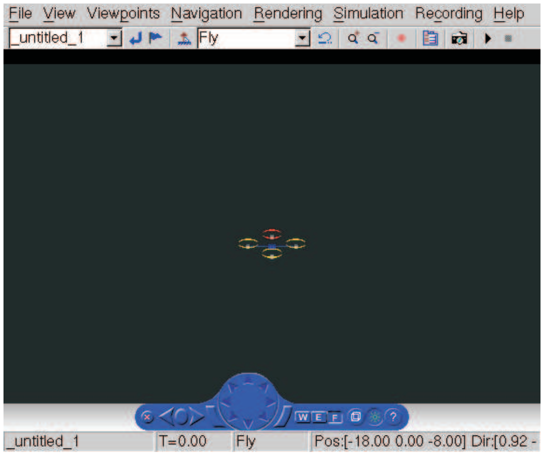
Fig. 16: Simulation results for $|K_\theta|, |K_\phi| = 5$. (a) Optical flow simulated signal. (b) State euler angles. (c) Linear velocity. (d) Angular velocity. (e) Position.

$$\widehat{\Delta i}(k) = \frac{\sum_{j=0}^{N} SQUAL_j(k)\Delta i_j(k)}{\sum_{j=0}^{N} SQUAL_j(k)} \qquad (20)$$

V. Conclusions and Future Work

The present document aimed at proposing a new way of controlling a quadrotor, using a lightweight cheap sensor, embedded in optical mice. The presented study in optics is good to dimensioning the lens for a certain application.

All alone, the sensor is insufficient, in the sense that it requires additional components to be able to successfully measure optical flow. On the one side, the illumination conditions must be sufficient enough to extract visible features. On the other side, the adapted lens is of utmost importance, since it determines the field of view of the sensor and, as such, it constrains the visible area. By diminishing the visible area, it


File View Viewpoints Navigation Rendering Simulation Recording Help
_untitled_1
Fly
W E F

_untitled_1 T=0.00 Fly Pos:[-18.00 0.00 -8.00] Dir:[0.92 -

Fig. 17: 3D Model added to the simulator

increases magnification and the size of features in the image, losing the amount of incident light. The trade-off should be balanced, in order not to jeopardize the effectiveness of the sensor.

All in all, the potential of such sensors is amazing, in the sense that the whole image processing is available for free. The whole concept of camera plus computer can in fact be eliminated, when resorting to these devices. They are cheaper, lighter, quicker and less complex.

It is possible to control the vehicle in position using these sensors, as shown by the presented simulation results. Note that the control laws applied in this thesis were very basic, as was the guidance system. The main guideline of this thesis was to build a solid baseline in the use of these types of sensors in the quadrotor. Hence, the developed work is extensive and methodical concerning the ADNS-5050. By simply compensating the data extracted from the sensor, it can be directly used to detect translational motion and act accordingly.

As future goals, the controller should be ported to the real system and tested, to validate the simulation outcome. Not only the sensor should be used for hover control, but also for landing and obstacle avoidance. New structural dispositions of the sensor should be studied in order to support the control of such manoeuvring. To avoid erroneous measurements due to error prone prototypes, new printed circuit boards (PCB) should be designed specially for the electronics in the presented system. This way, lens/sensor alignment errors are avoided.

The capabilities of the optical flow sensor should also be increased, by migrating the current system to the ADNS-3080, an optical flow sensor much more powerful than the ADNS-5050.

References

[1] T. Bresciani, "Modelling, identification and control of a quadrotor helicopter," Master's thesis, Lund University, 2008.

[2] S. Griffiths, "Remote terrain navigation for unmanned air vehicles," Master's thesis, Brigham Young University, 2006.

[3] G. Barrows, "Mixed-mode vlsi optical flow sensors for micro air vehicles," Master's thesis, University of Maryland, 1999.

[4] S. C. Kriel, "A comparison of control systems for the flight transition of vtol unmanned aerial vehicles," Master's thesis, University of Stellenbosch, March 2008.

[5] B. Cardeira, "Arquitecturas para navegação inercial/gps com aplicaçao a veículos autónomos," Master's thesis, Universidade Técnica de Lisboa, Instituto Superior Técnico, February 2009.

[6] R. W. Beard, "Quadrotor dynamics and control," Brigham Young University, Tech. Rep., 2008.

[7] G. Barrows, J. Chahl, and M. Srinivasan, "Biomimetic visual sensing and flight control," *The Aeronautical Journal*, 2003.

[8] A. Beyeler, J.-C. Zufferey, and D. Floreano, "optipilot: control of take-off and landing using optic flow," *Proceedings of EMAV 2009*, 2009.

[9] S. Zingg, D. Scaramuzza, S. Weiss, and R. Siegwart, "Mav navigation through indoor corridors using optical flow," *IEEE International Conference*, 2010.

[10] J.-C. Zufferey and D. Floreano, "Optic-flow-based steering and altitude control for ultra-light indoor aircraft," *Infoscience*, 2004.

[11] S. Hrabar and G. Sukhatme, "A comparison of two camera configurations for optic-flow based navigation of a uav through urban canyons," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004.

[12] B. Herisse, S. Oustrieres, T. Hamel, R. Mahony, and F.-X. Russotto, "A general optical flow based terrain-following strategy for a vtol uav using multiple views," *IEEE International Conference on Robotics and Automation*, 2010.

[13] R. Chan, A. Mulla, and K. Stol, "Characterisation of low-cost optical flow sensors," *IEEE International Conference on Robotics and Automation*, 2010.

[14] N. Franceschini, F. Ruffier, J. Serres, and S. Viollet, *Optic flow based visual guidance: from flying insects to miniature aerial vehicles*. InTech, 2009, pp. 747–770.

[15] "Euler angles," http://en.wikipedia.org/wiki/Euler_angles, July 2011.

[16] "Rotation (mathematics)," http://en.wikipedia.org/wiki/Rotation_(mathematics), July 2011.

[17] F. J., *Integrated Aircraft Navigation*. New York: Academic Press Inc, 1976.

[18] J. Kim and G. Brambley, "Dual optic-flow integrated navigation for small-scale flying robots," *ACT 0200*, 2007.

[19] J. E. Bortz, "A new mathematical formulation for strapdown inertial navigation," *IEEE Transactions on Aerospace and Electronic Systems*, 1970.

[20] A. Technologies, *ADNS-5050 - Optical Mouse Sensor*.

[21] ATMEL, *8-bit Microcontroller with 32K/64K/128K Bytes of ISP Flash and CAN Controller - AT90CAN32/AT90CAN64/AT90CAN128*.

[22] D. Flubacher, "Characterisation of an optical flow sensor for off-road robot application," Master's thesis, Swiss Federal Institute of Technology Zurich, 2008.

[23] X. Song, L. D. Seneviratne, and K. Althoefer, "A kalman filter-integrated optical flow method for velocity sensing of mobile robots," *IEEE/ASME Transactions on Mechatronics*, 2011.

[24] D. Sekimori and F. Miyazaki, "Self-localization for indoor mobile robots based on optical mouse sensor values and simple global camera information," *IEEE International Conference on Robotics and Biomimetics*, 2005.

[25] S. Kim and S. Lee, "Robust mobile robot velocity estimation using redundant number of optical mice," *Proceedings of the 2008 IEEE International Conference on Information and Automation*, 2008.

[26] M. Dille, B. Grocholsky, and S. Singh, "Outdoor downward-facing optical flow odometry with commodity sensors," *Proceedings of Field & Service Robotics*, 2009.

[27] A. Bonarini, M. Matteucci, and M. Restelli, "Automatic error detection and reduction for an odometric sensor based on two optical mice," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005.

[28] P. Pounds, R. Mahony, and P. Corke, "Modelling and control of a quad-rotor robot," *CSIRO ICT Centre*, 2005.

[29] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007.