

Article

Bio-Inspired Autonomous Visual Vertical and Horizontal Control of a Quadrotor Unmanned Aerial Vehicle

Saul Armendariz *, Victor Becerra  and Nils Bausch

School of Energy and Electronic Engineering, University of Portsmouth, Portsmouth PO1 3DJ, UK; victor.becerra@port.ac.uk (V.B.); nils.bausch@port.ac.uk (N.B.)

* Correspondence: saul.armendarizpuente@myport.ac.uk

Received: 29 December 2018; Accepted: 1 February 2019; Published: 5 February 2019



Abstract: Near-ground manoeuvres, such as landing, are key elements in unmanned aerial vehicle navigation. Traditionally, these manoeuvres have been done using external reference frames to measure or estimate the velocity and the height of the vehicle. Complex near-ground manoeuvres are performed by flying animals with ease. These animals perform these complex manoeuvres by exclusively using the information from their vision and vestibular system. In this paper, we use the Tau theory, a visual strategy that, is believed, is used by many animals to approach objects, as a solution for relative ground distance control for unmanned vehicles. In this paper, it is shown how this approach can be used to perform near-ground manoeuvres in a vertical and horizontal manner on a moving target without the knowledge of height and velocity of either the vehicle or the target. The proposed system is tested with simulations. Here, it is shown that, using the proposed methods, the vehicle is able to perform landing on a moving target, and also they enable the user to choose the dynamic characteristics of the approach.

Keywords: UAV; bio-inspiration; autonomous control; horizontal control; vertical control

1. Introduction

Unmanned Aerial Vehicle (UAV) usage and applications, specially those performed by Micro Aerial Vehicles (MAV), has increased. Now, more than ever they are being used in tasks such as inspection, surveillance, reconnaissance, and search and rescue [1]. This increased use demands for better navigation strategies to tackle more challenging approaches. To successfully accomplish this, UAV technologies need to be further advanced.

Navigation in unmanned vehicles is commonly performed using an external reference frames, such as global positioning systems and other sensors. This reliance on external reference frames severely hinders their autonomy. Constant changes in the mission context make it difficult for an autonomous vehicle to adapt to its changing environment. Near ground manoeuvres are vital to complete any flight mission successfully. Accurate velocity control of the vehicle at touchdown is critical. A combination of positioning systems, range finding sensors and image sensors have been popular tools in navigation strategies to accomplish autonomous landing [2].

Biologically inspired controllers in robots, unlike traditional controllers, emulate animals to achieve complex tasks. Flying animals control mechanisms have been optimized through millions of years of natural evolution, allowing them to navigate complex environments with ease, without relying on any external reference frame.

Tau theory, as the base of a bio-inspired controller, has been used in [3] to generate trajectories during UAV perching using information from external reference sensors, such as Global Positioning

System (GPS). Landing on a moving platform without knowledge of the vehicle's height or velocity has been achieved in [4] where the previously known size of the landing platform is used to estimate the position of the quadrotor body frame and generate an adequate landing trajectory.

The key contribution of this paper is a novel bio-inspired vertical and horizontal control system on-board the UAV to achieve near-ground manoeuvres on a moving target. This paper is organized as follows: The basics of Tau theory and its variants are described in Section 2. A body-centric control model is presented in Section 3 that it is complemented with a high-level control system described in Section 4. The estimation of visual motion is described in Section 5, followed by the objective tracking description in Section 6. Finally we perform simulations in Section 7, that we discuss in Section 8 and provide conclusions in Section 9.

2. Tau Theory

2.1. Flying Navigation Strategies in Nature

Flying insects have captured the attention of visual navigation researchers due to their ability to navigate complex and changing environments. Their large eyes with wide Field-of-View (FoV) suggest that they use optic flow to regulate motor actions. Flying bees, despite having two eyes, are not believed to use any depth perception information as their eyes separation does not allow them to capture this information [5]. This means that bees navigate using exclusively the optic flow patterns generated from their own motion. In [6], it was proposed that bees use a measure of image angular velocity ω_z , named the ventral flow, given by:

$$\omega_z = \frac{v_z(t)}{z(t)} \quad (1)$$

where $v_z(t)$ is the velocity and $z(t)$ is the distance to the objective at a given moment in time. When performing landing, it has been found that bees always land with a zero horizontal velocity at touch down [7]. This is achieved without knowledge of height or forward velocity, rather using their ratio, which is the image angular velocity in the vertical direction. While it descends towards the objective, the ventral flow increases due to the decrease in height. By holding constant the ventral flow while performing landing ($\omega_z = C$), both the velocity and the height decrease, until zero forward velocity is achieved at touch down. This has been named as the constant ventral flow strategy [6].

2.2. Biological Evidence of Tau Theory

When flying animals approach an object to land, capture or perch, as if they use predictive timing information linked to visual cues of their surrounding to guide and adjust their actions. Time-to-contact (TTC), sometimes referred to as time-to-collision, is defined as the remaining time before an anticipated contact between the approaching animal and the target. Based on the TTC, Lee introduced Tau theory [7]. He proposed that the variable Tau could be used to represent the TTC in the animals' visual systems. It is defined as the inverse of the target's relative rate of expansion on the animal's retina. In addition, Lee also proposed a general Tau theory, which states that the information from Tau is used in the guidance of the general movements of animals, not only on their perceptual mechanisms. This theory has been verified mathematically and experimentally, inspiring robotics researchers to apply Tau theory. In this project we use Tau theory to perform near-ground manoeuvres in a MAV.

Lee proposed that the animal movement is goal-directed. If a motion gap is defined as the difference between the animal's current motion state and its target state, then all the intended control actions are made for the purpose of closing the motion gap. If an object is at a distance $z > 0$ along some axis, then the Time-to-Contact to the object is defined as

$$TTC(t) = -\frac{z(t)}{\dot{z}(t)} \quad (2)$$

This can only be true when $\dot{z} \neq 0$. As the subject moves towards the target, the retinal image of the object in the subject’s eyes will dilate and the features of the target inside the subject’s retina will move radially. This image dilation is caused by the reduction of the relative distance between the subject and the target. It has been demonstrated [8] that the time-to-contact is the reciprocal of the image dilation and can be registered optically from the targets’s image features in the subject’s retina, such that:

$$TTC(t) = -\frac{\Phi(t)}{\dot{\Phi}(t)} \tag{3}$$

where $\Phi(rad)$ is the angle in the object’s retinal image. This shows that the time-to-contact can be registered optically without knowledge of the distance to the object or the relative velocity. The Time-to-Contact and Tau (τ) are connected as follows:

$$\tau = \frac{z(t)}{\dot{z}(t)} = -TTC(t) \tag{4}$$

2.3. Basic Tau Strategies

Assuming that the UAV has arrived at the desired location for landing and it is ready to descend, with Tau it is possible to initiate a descending trajectory, starting from the initial location at non-zero speed and ending right upon the target with a zero speed for a no impact landing. The only information needed to control an on-going descent action is the time rate of tau. It has been observed that animals tend to keep the time rate of tau constant as they close the gap towards their target [7].

$$\dot{\tau}(t) = k \tag{5}$$

where k is a constant. Integrating the previous equation we obtain

$$\tau(t) = kt + \tau_0 \tag{6}$$

where τ_0 is the initial constant value, which is:

$$\tau_0 = x_0 / \dot{x}_0 < 0 \tag{7}$$

where x_0 and \dot{x}_0 are the initial position and velocity of the vehicle, respectively. Substituting, we obtain:

$$x(t) / \dot{x}(t) = kt + \tau_0 \tag{8}$$

solving for $x(t)$, $\dot{x}(t)$ and $\ddot{x}(t)$ we obtain:

$$\begin{aligned} x(t) &= x_0(1 + kt\dot{x}_0/x_0)^{\frac{1}{k}} \\ \dot{x}(t) &= \dot{x}_0(1 + kt\dot{x}_0/x_0)^{\frac{1-k}{k}} \\ \ddot{x}(t) &= \frac{\dot{x}_0^2}{x_0}(1 - k)\dot{x}_0(1 + kt\dot{x}_0/x_0)^{\frac{1-2k}{k}} \end{aligned} \tag{9}$$

To visualize the effects of k independently from initial conditions, namely position, velocity and acceleration, each of the equations are normalized and the results are displayed in Table 1.

Table 1. Motion with different constant k values.

k	t	x	\dot{x}	\ddot{x}	Final Goal
$k < 0$	$\rightarrow t_d$	$\rightarrow \infty$	$\rightarrow \infty$	$\rightarrow \infty$	Gap not closed
$k = 0$	$\rightarrow t_d$	$= x_0$	$= \dot{x}_0$	$= 0$	Gap not closed
$0 < k < 0.5$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	Zero Touchdown
$k = 0.5$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$= C$	Slight Collision
$0.5 < k < 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow \infty$	Slight Collision
$k = 1$	$\rightarrow t_d$	$\rightarrow 0$	$= C$	$\rightarrow \infty$	Collision
$k > 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow \infty$	$\rightarrow \infty$	Strong Collision

Table 1 and Figure 1 show the values of x , \dot{x} and \ddot{x} with different k values. We can see that only the case with $0.5 \leq k < 1$ achieves a slight collision.

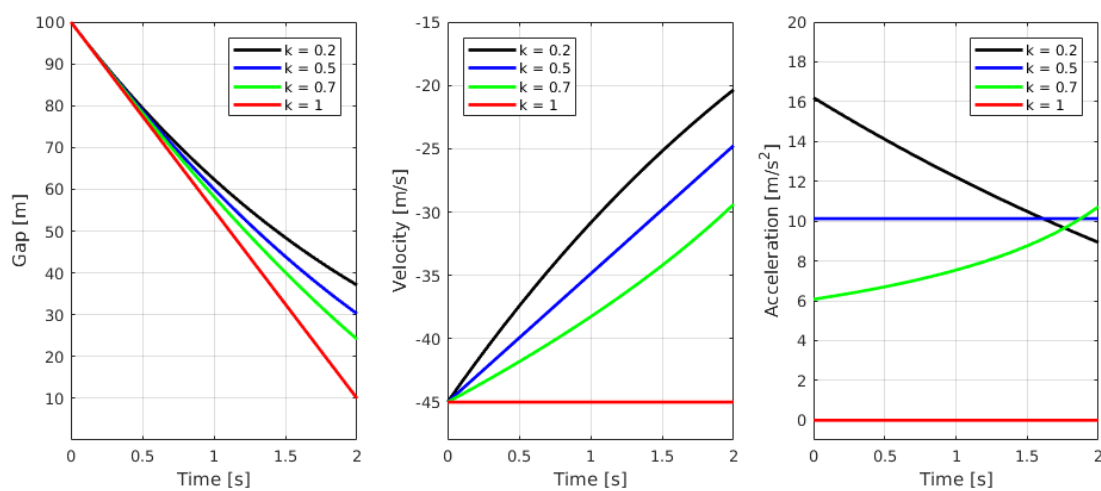


Figure 1. Values of x , \dot{x} and \ddot{x} with different values of k ($k = 0.2, 0.5, 0.7, 1.0$).

2.4. Tau Coupling

In a more realistic scenario, multiple gaps exist when approaching an objective and they all need to be closed simultaneously. Tau coupling [9] can be used for such situations. For example, if we need to close two translational gaps, $\alpha(t)$ and $\beta(t)$, the two corresponding tau variables will be linked by a constant ratio of $k_{\alpha\beta}$ during the course of the approach.

$$\tau_\beta = k_{\alpha\beta} \tau_\alpha \tag{10}$$

Taking this into consideration, we can rewrite Equation (9):

$$\begin{aligned} \beta &= C\alpha^{1/k_{\alpha\beta}} \\ \dot{\beta} &= \frac{C}{k_{\alpha\beta}} \alpha^{\frac{1}{k_{\alpha\beta}}-1} \dot{\alpha} \\ \ddot{\beta} &= \frac{C}{k_{\alpha\beta}} \alpha^{\frac{1}{k_{\alpha\beta}}-2} \left(\frac{1-k_{\alpha\beta}}{k_{\alpha\beta}} \dot{\alpha}^2 + \alpha \ddot{\alpha} \right) \end{aligned} \tag{11}$$

where the constant C is defined as $C = \beta_0/\alpha_0^{1/k_{\alpha\beta}}$. Similarly to the previous case, we can find the motion caused by different values of $k_{\alpha\beta}$.

These results indicate that when $0 < k_{\alpha\beta} \leq 0.5$ or $k_{\alpha\beta} = 1$, the distance, velocity and acceleration of the gap $\beta(t)$ will become zero in parallel to the closure of gap $\alpha(t)$, as seen in Table 2. Just as in the previous case, the gap closure can be modified with constant $k_{\alpha\beta}$ to perform different strategies,

such as: landing with zero velocity at touchdown, never closing the gap or achieving an aggressive gap closure.

Table 2. Motion with different constant $k_{\alpha\beta}$ values in coupling movement.

$k_{\alpha\beta}$	t	α	β	$\dot{\alpha}$	$\ddot{\beta}$	Final Goal
$k_{\alpha\beta} < 0$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow \infty$	$\rightarrow \infty$	$\rightarrow \infty$	Gap y not closed
$k_{\alpha\beta} = 0$	$\rightarrow t_d$	$\rightarrow 0$	$= 0$	$?$	$?$	Error
$0 < k_{\alpha\beta} < 0.5$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	Zero Touchdown
$0.5 \leq k_{\alpha\beta} < 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow \infty$	Slight Collision
$k_{\alpha\beta} = 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	Collision
$k_{\alpha\beta} > 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow \infty$	$\rightarrow \infty$	Strong Collision

2.5. Gravity Guidance Strategy

Previous examples had the disadvantages of requiring a downward velocity in order to be usable for landing. This can be achieved easily when the vehicle is in motion and the near-ground manoeuvre is initialized, but it will not initialize if the vehicle starts with a zero downward velocity. To solve this problem a method called “intrinsic Tau gravity guidance” was developed [7]. This is a special instance of Tau coupling where the $\alpha(t)$ gap is guided by the gravity’s constant vertical acceleration. This manoeuvre can be expressed as:

$$\tau_\alpha(t) = k_{\alpha g} \tau_g(t) \tag{12}$$

where the constant $k_{\alpha g}$ will determine the movement characteristics, and $\tau_g(t)$ specifies the time of the gap to be closed with gravity’s constant acceleration. The gap $x_g(t)$ makes use of $\tau_g(t)$, which can be derived from the free-fall equations under gravitational acceleration:

$$\begin{aligned} x_g(t) &= \frac{1}{2}gt_d^2 - \frac{1}{2}gt^2 \\ \dot{x}_g(t) &= -gt \end{aligned} \tag{13}$$

$$\tau_g(t) = \frac{x_g(t)}{\dot{x}_g(t)} = \frac{1}{2} \left(t - \frac{t_d^2}{t} \right) \tag{14}$$

where t_d is the time duration of the entire operation. Using Tau coupling, we can find the solution for $\alpha(t)$ as follows:

$$\begin{aligned} \alpha(t) &= \frac{\alpha_0}{t_d^{2/k_{\alpha g}}} (t_d^2 - t^2)^{\frac{1}{k_{\alpha g}}} \\ \dot{\alpha}(t) &= \frac{-2\alpha_0 t}{k_{\alpha g} t_d^{2/k_{\alpha g}}} (t_d^2 - t^2)^{\frac{1}{k_{\alpha g}} - 1} \\ \ddot{\alpha}(t) &= \frac{2\alpha_0}{k_{\alpha g} t_d^{2/k_{\alpha g}}} \left(\frac{2t^2}{k_{\alpha g}} - t^2 - t_d^d \right) (t_d^2 - t^2)^{\frac{1}{k_{\alpha g}} - 2} \end{aligned} \tag{15}$$

Table 3 and Figure 2 show the motion of gap closure on α , $\dot{\alpha}$ and $\ddot{\alpha}$ for different values of $k_{\alpha g}$.

Table 3. Motion with different constant $k_{\alpha g}$ values in during intrinsic Tau gravity movement.

$k_{\alpha g}$	t	α	$\dot{\alpha}$	$\ddot{\alpha}$	Final Goal
$k_{\alpha g} < 0$	$\rightarrow t_d$	$\rightarrow \infty$	$\rightarrow \infty$	$\rightarrow \infty$	Gap not closed
$k_{\alpha g} = 0$	$\rightarrow t_d$	$= 0$	$?$	$?$	Error
$0 < k_{\alpha g} < 0.5$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	Zero Touchdown
$0.5 \leq k_{\alpha g} < 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow \infty$	Slight Collision
$k_{\alpha g} = 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow 0$	$\rightarrow 0$	Collision
$k_{\alpha g} > 1$	$\rightarrow t_d$	$\rightarrow 0$	$\rightarrow \infty$	$\rightarrow \infty$	Strong Collision

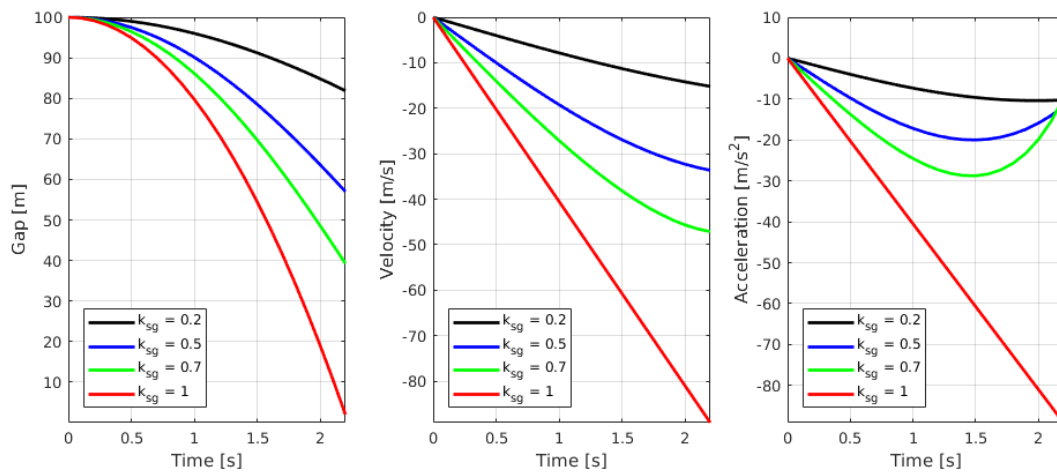


Figure 2. Values of α , $\dot{\alpha}$ and $\ddot{\alpha}$ with different values of $k_{\alpha g}$ ($k_{\alpha g} = 0.2, 0.5, 0.7, 1.0$).

2.6. Tau Theory Link to Constant Optic Flow Approach

Tau strategies have also been found in more developed species, such as birds and mammals, which require more complex visual locomotion strategies than insects with their constant optic flow approach. During vertical landing, using the constant dilation approach [10] for asymptotic closure of vertical gaps, the image dilation ω_z is given by:

$$\omega_z = -\frac{\dot{z}}{z} \tag{16}$$

which is held constant during the execution of the constant dilation strategy. Since the image dilation is the reciprocal of τ :

$$\tau = -\frac{1}{\omega_z} \tag{17}$$

This means that $\dot{\tau} = 0$, making the constant dilation strategy an implementation of the tau control strategy with a constant value of $k = 0$. This creates a soft touch landing with constant deceleration. The constant dilation strategy is a special case of the tau theory.

3. Body-Centric Quadrotor Model

The quadrotor model presented here is similar to the one developed in [11] and taken from [12]. For the purpose of modelling the quadrotor, two Cartesian coordinate frames are defined. The Earth-surface fixed frame, with axes $1_x^e, 1_y^e$ and 1_z^e aligned with north, east and down directions in the Earth frame. The second body frame is a body-fixed frame with its origin at the body centre of mass, and axes $1_x, 1_y$ and 1_z aligned with the forward, starboard (right), and down body orientations. The Earth and body coordinate frames, motor numbering and the rotation directions are illustrated in Figure 3.

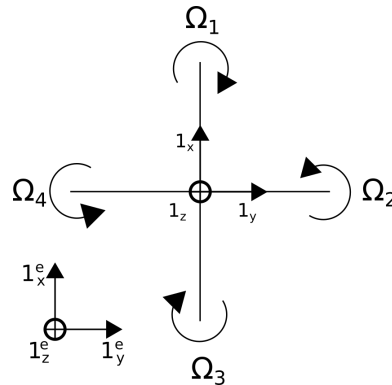


Figure 3. Top view of the quadrotor with the defined coordinate frames, motor numbering and positive motor rotation directions.

3.1. Attitude and Rotation Representation

The body attitude is represented, relative to the Earth frames, by the right-handed rotation sequence (yaw, pitch, roll) with angles ψ , θ , and ϕ about 1_z , 1_y and 1_x axes respectively. These three rotations define the transformation matrix $R_{b/e}$. Consequently, the quadrotor angular velocity in the Earth frame $\omega_{b/e}^e = [\dot{\psi}, \dot{\theta}, \dot{\phi}]$ and in the body frame $\omega_{b/e}^b = [p, q, r]$ are related as follows [13]:

$$\begin{bmatrix} mI_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_q \end{bmatrix} \begin{bmatrix} \dot{V}^b \\ \dot{\omega}_{b/e}^b \end{bmatrix} + \begin{bmatrix} \omega_{b/e}^b \times mV^b \\ \omega_{b/e}^b \times I_q \omega_{b/e}^b \end{bmatrix} = \begin{bmatrix} F^b \\ \tau^b \end{bmatrix} \tag{18}$$

3.2. Quadrotor Body Dynamics

Using Newton’s Euler formalism, the body dynamics are expressed in the body-fixed frame as:

$$\omega_{b/e}^e = \begin{bmatrix} 1 & \tan(\theta) \sin(\phi) & \tan(\theta) \cos(\phi) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \omega_{b/e}^b \tag{19}$$

We assume that the quadrotor is symmetric about its body principal axes, which coincide with the body frame. This assumption cancels all products of inertia and the inertial matrix becomes a diagonal matrix $I_q = \text{diag}(I_{xx}, I_{yy}, I_{zz})$.

The external forces acting on the quadrotor body are the weight force mg and the thrust forces generated by the four propellers T_i . Each thrust force is modelled as:

$$T_i = n\Omega_i^2, \quad i = 1, 2, 3, 4 \tag{20}$$

and the total thrust force $T_a = T_1, T_2, T_3, T_4$ is always aligned with the body 1_z axis in the negative direction. The total torque acting on the quadrotor is composed of the control torques and gyroscopic effect torque. Control torques τ_x and τ_y , which generate a positive rolling and pitching moment, can be expressed as

$$\tau_x = \ell(T_4 - T_2)1_x, \quad \tau_y = \ell(T_1 - T_3)1_y \tag{21}$$

The aerodynamic drag torque Q_i acting on a propeller i is modelled as

$$Q_i = d\Omega_i^2, \quad i = 1, 2, 3, 4 \tag{22}$$

The total drag torque that generates a positive yawing moment is expressed as

$$\tau_z = d(Q_2^2 + Q_4^2 - Q_1^2 - Q_3^2)1_z \tag{23}$$

Body angular rates induce a gyroscopic effect torque τ_J on each of the rotating propellers due to rotor inertia J and the total imbalance Ω_{res} in the propeller angular velocities; τ_J can be expressed as

$$\tau_J = J(\omega_{b/e}^b \times 1_z)\Omega_{res} = \begin{bmatrix} Jq\Omega_{res} \\ -Jp\Omega_{res} \\ 0 \end{bmatrix} \quad (24)$$

where

$$\Omega_{res} = \Omega_2 + \Omega_4 - \Omega_1 - \Omega_3 \quad (25)$$

By defining the following variables

$$\begin{aligned} U_1 &= (\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ U_2 &= (\Omega_4^2 - \Omega_2^2) \\ U_3 &= (\Omega_1^2 - \Omega_3^2) \\ U_4 &= (\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \end{aligned} \quad (26)$$

the quadrotor model dynamic equations $(\dot{p}, \dot{q}, \dot{r}, \dot{v}_x, \dot{v}_y, \dot{v}_z)$ expressed in the body-fixed coordinates frame as well as the local Earth attitude kinematics $(\dot{\psi}, \dot{\theta}, \dot{\phi})$ can be written as

$$\begin{aligned} \dot{p} &= [qr(I_{yy} - I_{zz}) + Jq\Omega_{res} + \ell nU_2]/I_{xx} \\ \dot{q} &= [pr(I_{zz} - I_{xx}) - Jp\Omega_{res} + \ell nU_3]/I_{yy} \\ \dot{r} &= [pq(I_{xx} - I_{yy}) + dU_4]/I_{zz} \\ \dot{v}_x &= rv_y - qv_z - g \sin(\theta) \\ \dot{v}_y &= pv_z - rv_x + g \cos(\theta) \sin(\phi) \\ \dot{v}_z &= qv_x - pv_y + g \cos(\theta) \cos(\phi) - nU_1/m \\ \dot{\phi} &= p + q \tan(\theta) \sin(\phi) + r \tan(\theta) \cos(\phi) \\ \dot{\theta} &= q \cos(\phi) - r \sin(\phi) \\ \dot{\psi} &= q \sin(\phi) / \cos(\theta) + r \cos(\phi) / \cos(\theta) \end{aligned} \quad (27)$$

4. Control Scheme

The quadrotor is an open-loop unstable system with fast rotational dynamics. The proposed control scheme has two parts: a low-level stabilizing controller and a high-level bio-inspired controller in charge of near ground manoeuvres.

4.1. Low-Level Controller

For the low-level controller a discrete time linear regulator with a direct feed-through matrix [14] is selected to perform stabilizing control on the quadrotor. The controller takes as input a vector of references

$$y_r = [\psi_r, a_{xr}, a_{yr}, a_{zr}]^T \quad (28)$$

and a state vector

$$x = [\phi, \theta, \psi, p, q, r]^T \quad (29)$$

Finally, it outputs a control vector

$$u = [\Omega_1, \Omega_2, \Omega_3, \Omega_4]^T \quad (30)$$

The controller is designed with basis on the previous Jacobian linearised dynamic model (27), about the equilibrium point $x_{eq} = [0, 0, 0, 0, 0, 0]^T$ and $u_{eq} = [\Omega_h, \Omega_h, \Omega_h, \Omega_h]^T$, where Ω_h is the necessary speed in rad/s to maintain hover. The low-level control method is taken from [12] and uses a linear quadratic tracker approach. The control is given by

$$u(n) = -Kx(n) + Fy_r(n + 1) \quad (31)$$

where matrices K and F are the state feedback and reference feed-forward gains, respectively. The purpose of the low-level controller is to stabilize the quadrotor's fast rotational dynamics by tracking a body acceleration and heading reference signal (28). This complements the high-level controller whose purpose is to use Tau theory to command the low-level controller with a suitable reference signal. The values of matrices K and F can be found on Appendix A.

4.2. High-Level Controller

The high-level controller will be in charge of supplying the low-level controller with suitable reference signals based on Tau theory. This can be achieved knowing that the vertical image dilation ω_z is equal to the reciprocal of Lee's basic Tau law [10]:

$$\omega_{zr}(t) = -\frac{1}{\tau(t)} \quad (32)$$

Substituting Tau, we obtain:

$$\omega_{zr}(t) = -\frac{1}{kt + \tau_0} \quad (33)$$

This means that regulating the visually registered image dilation to track $\omega_{zr}(t)$ becomes equivalent to enforcing the original Tau theory [7] with a constant k value that reflects the manoeuvre we wish to accomplish. Looking at Equation (7) we can see that, in order for this implementation to be viable, a downward vertical (negative) velocity is necessary be properly initialized. This limits τ_0 to only negative values, otherwise the control law will cause the quadrotor to open the gap and fly away from the ground. Additionally, t needs to satisfy

$$t < \frac{\tau_0}{k} \quad \text{if} \quad k < 0 \quad \text{and} \quad t > -\frac{\tau_0}{k} \quad \text{if} \quad k > 0 \quad (34)$$

A simple solution to this problem, to perform near ground manoeuvres from hover, is to substitute the basic Tau implementation with its intrinsic tau gravity guidance counterpart [15]:

$$\omega_{zr}(t) = -\frac{1}{k_{\alpha g} \tau_g(t)} \quad (35)$$

Two values need to be defined for this implementation, the constant $k_{\alpha g}$ and t_d . Constant $k_{\alpha g}$ will dictate the approach that the quadrotor will take regarding the manoeuvre. As indicated in Table 3, the constant chosen will modify the way the action will be performed, from a zero velocity manoeuvre at touchdown, to a strong collision. The choice of t_d will dictate the manoeuvre execution time.

4.3. High-Level Control for Horizontal Manoeuvres

When multiple gaps need to be closed simultaneously, namely, $\alpha(t)$ and $\beta(t)$, the Tau coupling strategy [15] can be implemented. During this operation, the two corresponding Tau will be kept at a constant ratio $k_{\alpha\beta}$. This can be used to close vertical and horizontal gaps simultaneously. As previously discussed, the value of the $k_{\alpha\beta}$ constant will dictate the characteristics of the manoeuvre.

4.4. High-Level Vertical and Horizontal Control Implementation

Tau is controlled by tracking a time-varying image dilation reference signal obtained from the intrinsic tau gravity $\omega_{zr}(t)$ using Equation (35). The visual on-board processing system registers the value of the image dilation $\omega_z(n)$ with a sampling time T_s at a discrete time step n . A PI controller is used to regulate the dilation error e_{ω_z}

$$e_{\omega_z}(n) = \omega_{zr}(n) - \omega_z(n) \tag{36}$$

by providing a suitable reference signal (a_{zr})

$$a_{zr}(n) = K_P e_{\omega_z}(n) + K_I \sum_{i=0}^n e_{\omega_z}(i) \tag{37}$$

to the low level controller, where K_I and K_P are the PI controller gains. The values of the control gains can be found on Appendix A.

To achieve horizontal landing and tracking, the tau coupling strategy is used. If we have three gaps that need to be closed simultaneously, namely $z(t)$, $x(t)$ and $y(t)$, that coincide with the quadrotor 1_z , 1_x and 1_y respectively, it is possible to use the Tau coupling strategy in Equation (11) to find a suitable $\ddot{x}(t)$ and $\ddot{y}(t)$ linked to a_{zr} that will provide body acceleration reference signals, a_{xr} and a_{yr} respectively, for the low-level controller to be input into the reference vector (28) as follows

$$\begin{aligned} a_{xr} &= \ddot{x}(t) \\ a_{yr} &= \ddot{y}(t) \end{aligned} \tag{38}$$

The previous reference values will only be useful when they align with the vehicle reference frames 1_x and 1_y respectively. In order for the reference signals to point towards a target, they would need to be updated. As explained on Section 6.1: Equation (38) would need to be updated as Equation (60). The heading reference component ψ_r in the reference input vector (28) is set to a predefined constant value to hold the heading value while the manoeuvre is performed.

5. Estimation of Visual Motion Parameters

Optic flow corresponds to the image velocities (u, v) in the patterns of apparent motion of objects on frame caused by the relative motion between the subject and a scene. This includes three translational velocities (v_x^c, v_y^c, v_z^c) and three angular velocities (p^c, q^c, r^c), the depth of the observed objective Z , and the cameras focal length f . This can be expressed as:

$$\begin{aligned} u &= -f \left(\frac{v_x^c}{Z} + q^c \right) + x \frac{v_z^c}{Z} + yr^c - x^2 \frac{q^c}{f} + xy \frac{p^c}{f} \\ v &= -f \left(\frac{v_y^c}{Z} - p^c \right) + y \frac{v_z^c}{Z} - xr^c + y^2 \frac{p^c}{f} - xy \frac{q^c}{f} \end{aligned} \tag{39}$$

The translational and angular velocities are given in the camera frame, rigidly attached to the camera where its 1_x^c and 1_y^c axes are aligned with the image horizontal and vertical frames, and the 1_z^c axis is aligned with the optical axis flow towards the scene. This estimation will find the visual motion parameters, namely, the Focus of Expansion (FOE), the camera frame dilation ω_z^c and the ventral flows ω_x^c and ω_y^c . This will be used to control the quadrotor during the near ground manoeuvres. The system implemented here is taken from [16].

5.1. Simultaneous Visual Motion Parameters Estimation

By removing the rotational component of the optic flow from Equation (39), the translational components of the optic flow, u_T, v_T can be expressed as

$$\begin{aligned}
 u_T &= -f \frac{v_x^c}{Z} + x \frac{v_z^c}{Z} \\
 v_T &= -f \frac{v_y^c}{Z} + y \frac{v_z^c}{Z}
 \end{aligned}
 \tag{40}$$

We rewrite the previous equation in terms of the visual motion parameters $\omega_x^c, \omega_y^c, \omega_z^c$, keeping in mind that $v_z^c = \dot{Z}$, the image dilation can be described as

$$\omega_z^c = \frac{v_z^c}{Z}
 \tag{41}$$

Using Equations (40) and (41), the translational optic flow components can be rewritten as:

$$\begin{aligned}
 u_T &= -f\omega_x^c + x\omega_z^c \\
 v_T &= -f\omega_y^c + x\omega_z^c
 \end{aligned}
 \tag{42}$$

In addition, the image frame coordinates of the Focus of Expansion (FOE), x_{FOE}, y_{FOE} can be calculated as

$$\begin{aligned}
 x_{FOE} &= \frac{v_x^c}{v_z^c} \\
 y_{FOE} &= \frac{v_y^c}{v_z^c}
 \end{aligned}
 \tag{43}$$

Note that the FOE can only exist when $v_z^c \neq 0$.

Due to the high number of points where the optic flow can be evaluated, a parametric model can be used to simultaneously calculate the visual motion parameters. The translational components (42) can be represented using the following model:

$$\begin{aligned}
 u_T &= a_1 + a_2x \\
 v_T &= a_3 + a_2y
 \end{aligned}
 \tag{44}$$

Then, the optic flow calculations are used to form a least-square regression problem. In this way, Equation (44) can be rewritten as

$$\begin{bmatrix} u_{T1} \\ v_{T1} \\ u_{T2} \\ v_{T2} \\ \vdots \\ u_{Tn} \\ v_{Tn} \end{bmatrix} = \begin{bmatrix} 1 & x_1 & 0 \\ 0 & y_1 & 1 \\ 1 & x_2 & 0 \\ 0 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ 1 & x_n & 0 \\ 0 & y_n & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}
 \tag{45}$$

This can be solved using least squares to find the estimated model parameters \hat{a}_1, \hat{a}_2 and \hat{a}_3 . Then, the image dilation in camera frame, ventral flows and FOE can be found with:

$$\begin{aligned}
 \omega_z^c &= \hat{a}_2 \\
 \omega_x^c &= -\frac{\hat{a}_1}{f} \\
 \omega_y^c &= -\frac{\hat{a}_3}{f} \\
 x_{FOE} &= -\frac{\hat{a}_1}{\hat{a}_2} \\
 y_{FOE} &= -\frac{\hat{a}_3}{\hat{a}_2}
 \end{aligned}
 \tag{46}$$

We need to consider that the camera is attached to the quadrotor in such way that the 1_z^c axis coincides with the body 1_z axis, while the camera 1_x^c and 1_y^c axes are rotated with an angle ψ_c about the body 1_z axis with respect to the body axes 1_x and 1_y , respectively. This means that while the image dilation in the camera and body frame are equal, the ventral flows need to be adjusted. This can be done as follows:

$$\begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix} = \begin{bmatrix} \cos(\psi_c) & -\sin(\psi_c) \\ \sin(\psi_c) & \cos(\psi_c) \end{bmatrix} \begin{bmatrix} \omega_x^c \\ \omega_y^c \end{bmatrix} \tag{47}$$

5.2. Outlier Rejection

The proposed method for visual motion parameters estimation has been shown to produce accurate results [16], however, the raw estimates obtained from Equation (47) can exhibit outliers, caused by the temporary violation of assumptions made by the optic flow method and due to the noisy nature of digital visual information. To deal with this issue, the outliers need to be eliminated in real time. A median filter is a good robust statistical filter that can be used for outlier rejection. The running median filter presented in [17] is used to reject outliers over a window of previous values.

5.3. Sensor Fusion: IMU Aided Estimation of Visual Motion Parameters (VMP)

Images captured from cameras are naturally noisy, reducing their accuracy during its processing. Even with the fast real-time method used here, image capture update rate is low compared to the dynamics of aerial vehicles. In order to use this information in a control system with a higher sampling rate, it is necessary to estimate the visual information between sampling instants. Even after the application of the outlier rejection filter, the resulting estimates will contain noise. Inter-sampling estimation can be achieved using a stochastic model-based estimation algorithm, such as a Kalman Filter. A dynamic filter of the visual motion parameters is derived for this purpose.

If a downward-looking camera is rigidly mounted on a quadrotor, the height of the quadrotor z and the scene depth at the centre of the image Z are related by their attitude angles. If we assume small attitude angles, it is possible to use the approximation $z \approx Z$.

Defining x_d as:

$$x_d = \frac{1}{z} \tag{48}$$

Taking its time derivative yields

$$\dot{x}_d = -\frac{\dot{z}}{z^2} \tag{49}$$

using a previously defined definition of image dilation (41), the derivative can be rewritten as

$$\dot{x}_d = \omega_z x_d \tag{50}$$

Taking the time derivative of (16) and assuming $z \neq 0$ give

$$\dot{\omega}_x = \frac{\dot{v}_x}{z} - \frac{\dot{z}}{z} \frac{v_x}{z} \tag{51}$$

Revising Equation (27), the acceleration component in the 1_x axis of the body frame is $a_x = -g \sin(\theta)$, and $\dot{\omega}_z$ can be rewritten as

$$\dot{\omega}_x = r \frac{v_y}{z} - q \frac{v_z}{z} + \frac{a_x}{z} - \frac{\dot{z}}{z} \frac{v_x}{z} \tag{52}$$

which can be also be written in the following form, taking into consideration Equations (1) and (48)

$$\dot{\omega}_x = r \omega_y - q \omega_z + \omega_x \omega_y + a_x x_d \tag{53}$$

Given that the body-frame starboard and downward accelerations are $a_y = g \cos(\theta) \sin(\phi)$ and $a_z = g \cos(\theta) \cos(\phi) - T_a/m$, the equations for $\dot{\omega}_y$ and $\dot{\omega}_z$ can be derived as well

$$\begin{aligned}\dot{\omega}_y &= p\omega_z - r\omega_x + \omega_y\omega_z + a_yx_d \\ \dot{\omega}_z &= q\omega_x - p\omega_y + \omega_z^2 + a_zx_d\end{aligned}\quad (54)$$

Using Equations (50), (53) and (54), the dynamic system for the visual motion parameters defined by the state vector $x_f = [\omega_x, \omega_y, \omega_z, x_d]^T$ which will be used in the Kalman Filter. The filter will predict the visual motion parameters at a higher rate to allow a more responsive high-level control.

Similar examples of data fusion can be observed in nature. Visual and non visual cues, such as gravito-inertial senses and efferent copies all play a collaborative role in forming the perception of motion [18]. With this information the brain is capable of build an estimate based on the information available. This is further supported by [19], where a mismatch between the expected and received motion cues can trigger motion sickness in humans.

The Cubature Kalman Filter [20], a variation of the Unscented Kalman Filter (UKF) [21] with a spherical-radial cubature rule is used here. It has been proved to be superior to the Extended Kalman Filter (EKF) [22] and, in some cases, superior to the UKF [23].

The CKF will output a state vector $x_f = [\omega_x, \omega_y, \omega_z, x_d]^T$, while the input vector $u_f = [p, q, r, a_x, a_y, a_z]^T$ is provided by the IMU, and the measurement vector $y_f = [\omega_x, \omega_y, \omega_z]$ is provided by the visual system. This system will be implemented by discretizing Equations (50), (53) and (54). The CKF produces estimates of \hat{x}_f at the same rate as the IMU readings, to be used by the high-level controller, to enable a smoother and more responsive control at a higher rate. The CKF uses a constant process covariance matrix Q_f that can be chosen manually to take into account the unmodelled input noise. Additionally, a time variant noise covariance R_f is defined as

$$R_f = \text{diag} \left(\frac{1}{\sigma_v^2}, \frac{1}{\sigma_v^2}, \frac{1}{\sigma_v^2} \right) \quad (55)$$

where σ_v is calculated from the root mean square of the optic flow residuals in the fitting process described in Equation (45).

6. Objective Tracking

In order to perform near ground manoeuvres on a moving target, first we need to be able to accurately detect and track the object. For this work we have decided to use AprilTags [24]. AprilTags are open source fiducial marks; artificial visual features designed for automatic detection. Initially used for augmented reality applications, they have since been widely adopted by the robotics community for uses such as: ground truth, pose estimation, and object detection and tracking. AprilTags are black-and-white square tags with an encoded binary payload.

In experiments [24], these tags have proved to have high accuracy, low false positive rate and inexpensive computation time. The main drawback of using any fiducial mark is the need to perform camera calibration to take into consideration the camera's focal length, principal point and radial distortion coefficients for each camera model.

6.1. Adjust Body Reference to Target Location

To use the body accelerations as reference signals a_{xr} and a_{yr} from the high-level into the low-level controller to point towards out target, we need to know the objective's quadrant in the camera's Cartesian system. Using the AprilTags, we can extract the target coordinates in the camera frame in pixels, namely u^c and v^c . If we consider the centre of the camera frame, u_0^c and v_0^c in pixels, as the origin of the Cartesian system, then we can calculate the distance from the origin to the objective as:

$$\begin{aligned}u_\Delta &= u_0^c - u^c \\ v_\Delta &= v_0^c - v^c\end{aligned}\quad (56)$$

Then, we can proceed to determine the angle of the objective from the centre of the camera in polar coordinates with

$$\lambda^c = \tan^{-1} \left(\frac{v_\Delta}{-u_\Delta} \right) \tag{57}$$

Finally, we separate the angle into its x and y components

$$\begin{aligned} u_\lambda &= -\cos(\lambda^c) \\ v_\lambda &= -\sin(\lambda^c) \end{aligned} \tag{58}$$

Each of the components will have a value that will range from -1 to 1 , depending on the position of the target in the camera Cartesian system. This value will be multiplied by the reference body acceleration, this way the reference body acceleration signal will always move the vehicle, in the body 1_x and 1_y axes to the location of the objective (See Figure 4).

Since the position of the camera does not align with the body 1_x and 1_y axes, we need to rotate the camera Cartesian system with an angle ψ_c . Subtracting ψ_c from Equation (58), we obtain:

$$\begin{aligned} u_\lambda &= -\cos(\lambda^c - \psi_c) \\ v_\lambda &= -\sin(\lambda^c - \psi_c) \end{aligned} \tag{59}$$

Finally, the acceleration reference signal that will be feed into the input vector (28) to control the horizontal movement of the vehicle towards the objective is

$$\begin{aligned} a_{xr} &= \ddot{x}(t)u_\lambda \\ a_{yr} &= \ddot{y}(t)v_\lambda \end{aligned} \tag{60}$$

Equation (60) is an update on Equation (38) that takes into consideration the position of the camera in the vehicle and corrects the reference signals to point towards the objective.

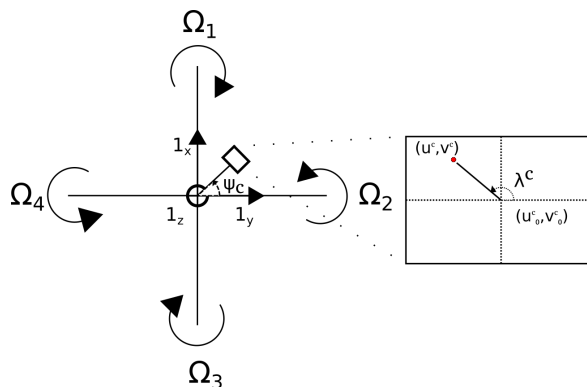


Figure 4. Top view of the quadrotor with the defined coordinate frames, camera location and camera frame conventions.

7. Simulations

7.1. Simulation Environment

Simulations are performed using the Robot Operating System (ROS) [25], a flexible framework for writing robot software. ROS includes a collection of tools, libraries, and conventions to design complex and robust robots. ROS is used in conjunction with Gazebo [26] a simulation environment to rapidly test algorithms using realistic scenarios. The algorithms prototyping will be tested using RotorS [27], a modular gazebo MAV simulator framework. The estimation of motion parameters makes use of the OpenCV [28] software library and the Eigen [29] C++ template library for linear algebra.

RotorS provides us with multi-rotor models such as the AscTec Hummingbird, the AscTec Pelican, and the AscTec Firefly, with the possibility to build custom multi-rotors and even fixed-wing unmanned vehicles. For our experiments we will be using the AscTec Pelican. The Pelican is a flexible research UAV platform that allows us to perform all the computer vision and high-level processes on-board, without the need of any external computing units.

The simulated Pelican incorporates a variety of sensors, including the IMU (three-axes accelerometer and rate gyroscopes), and attitude and heading reference system (AHRS), and one downward-looking camera with a resolution of 720×480 pixels and a focal length of 49 degrees of vertical field of view.

To simulate a moving target, we will use a Husky [30] Unmanned Ground Vehicle (UGV), a field robotics platform that support ROS and can be loaded with a variety of sensors. In our case, a 2×2 m square platform will be placed on top of the Husky UGV with an AprilTag of the same dimensions, in order for the quadrotor to see the moving platform. Simulations were performed on a Ubuntu 16.04 computer with an AMD Ryzen 3 1200 CPU with 8 GB of RAM and a Nvidia GTX 1050Ti GPU. The Gazebo simulation can be seen in Figure 5.

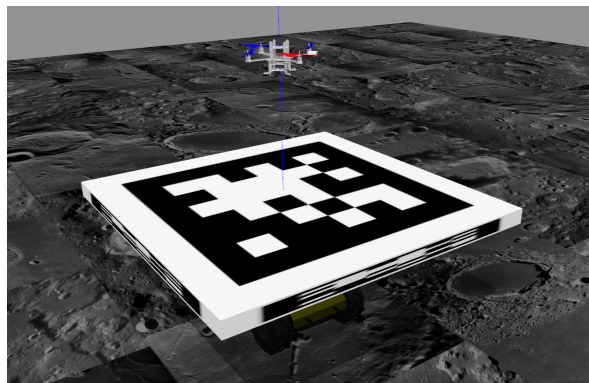


Figure 5. Pelican UAV on simulated environment with AprilTag on a platform on top of a Husky UGV.

7.2. Autonomous Tau-Based Control Simulation

Simulations are performed on Gazebo, the simulation environment, using ROS and RotorS with an AscTec Pelican. To accurately use the optic flow during the visual motion parameters calculation, the ground is covered with a print of randomly assembled lunar images taken by the personal telescope of Wes Higgins [31] (See Figure 5). The main source of light on the simulation has Gazebo default position and values. The quadrotor is flown manually to a 4 m height in the simulated environment, while the Husky is set in different location within camera frame. Four simulations were performed with different k_g , k_{gx} and k_{xy} constants and start from a hover position. The value of t_d is set to 4 across all simulations and x_0 is set to 10.

In Simulation 1, $k_g = 0.4$ while $k_{gx} = k_{xy} = 1.0$. The position of the Pelican and the Husky can be seen in the graphs on Figure 6. It can be observed that the Pelican is capable of tracking the objective and land on it, with a soft landing, when it is above it.

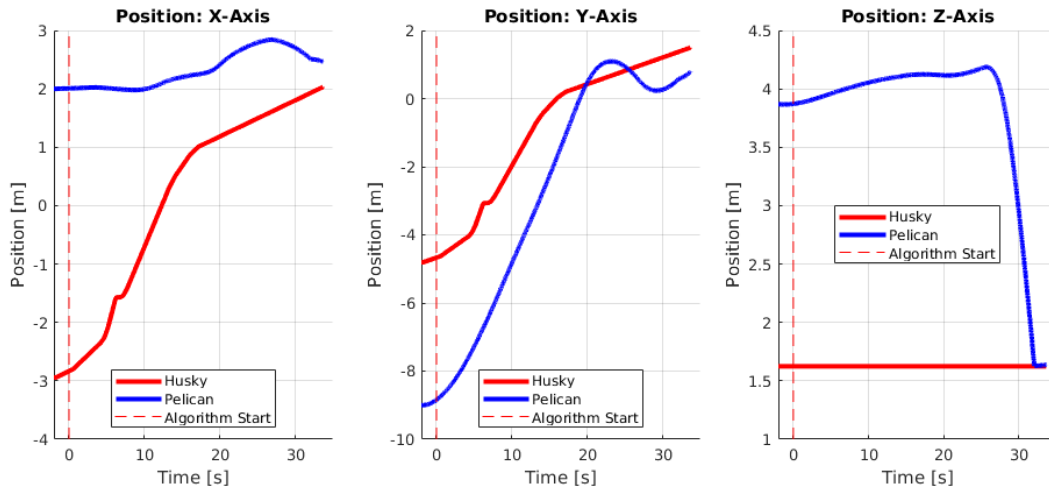


Figure 6. Position, in Gazebo’s reference frame, of the Pelican UAV and Husky UGV over time during simulation, with k values of $k_g = 0.4$ and $k_{gx} = k_{gy} = 1.0$.

On Simulation 2 the values are set to: $k_g = 1.0$ and $k_{gx} = k_{gy} = 1.0$. The position of the Pelican and the Husky can be seen in the graphs on Figure 7. This manoeuvre is similar to the previously described, but due to the choice of k_g , landing is achieved with a higher vertical velocity. As previously discussed, this kind of movement can be useful during perching operations, just like some birds do to catch preys.

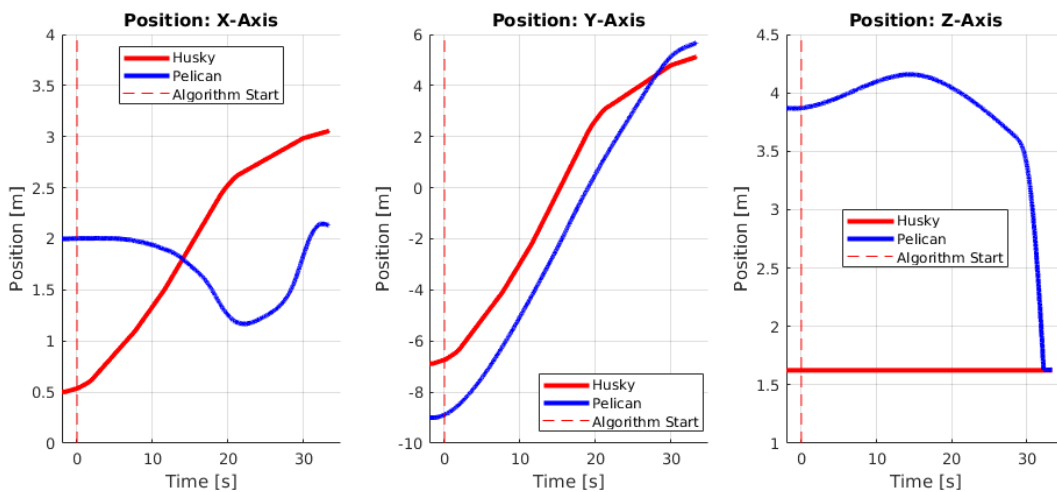


Figure 7. Position, in Gazebo’s reference frame, of the Pelican UAV and Husky UGV over time during simulation, with k values of $k_g = 1.0$ and $k_{gx} = k_{gy} = 1.0$.

Note that the difference in distance in axes x and y at touchdown is due to the location of the IMU inside the Husky. The sensor is located on the vehicle’s centre while the platform is a 2×2 m square. This explains why in the previously mentioned axes is not uncommon to end the manoeuvre with a difference of up to 1 m.

Velocities during the landing manoeuvres in Simulations 1 and 2 can be compared in Figure 8. They have a mean downward velocity of -0.0154 and -0.0035 m/s, and an execution time of 5.52 and 3.28 s, respectively. This confirms that the values of constant k_g modifies the vehicle dynamics during landing.

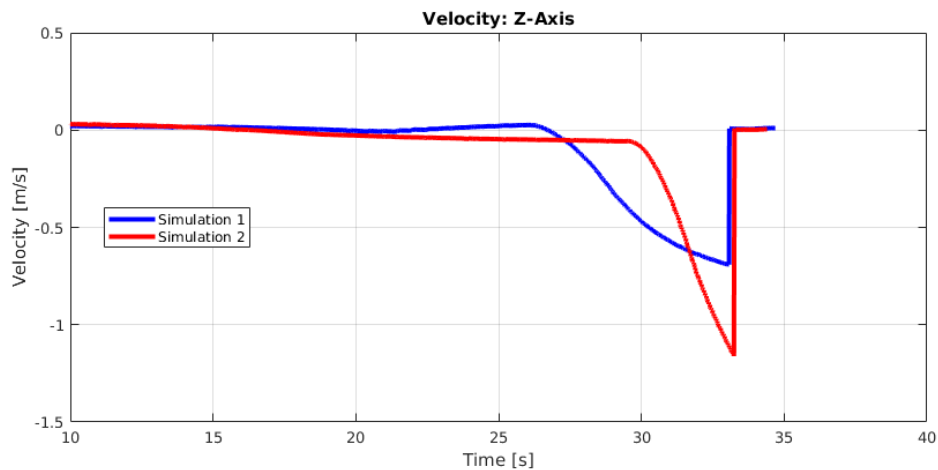


Figure 8. Velocity in the Z-Axis during Simulations 1 and 2.

Simulation 3 is performed with values of $k_g = 0.4$ and $k_{gx} = k_{xy} = 0.5$. The position of the Pelican and the Husky can be seen in the graphs on Figure 9. In this simulation, the Pelican will follow the Husky while keeping its distance, this is achieved due to the value of the constant k_{xy} and k_{gx} . This manoeuvre showcases the flexibility that Tau can achieve during near ground navigation. Just like observed in birds of prey, the quadrotor is capable of give chase to a target. During the simulation the Pelican and the Husky had a mean difference in distance of 0.7, 0.9 and 2.5 m in the x , y and z axes, respectively.

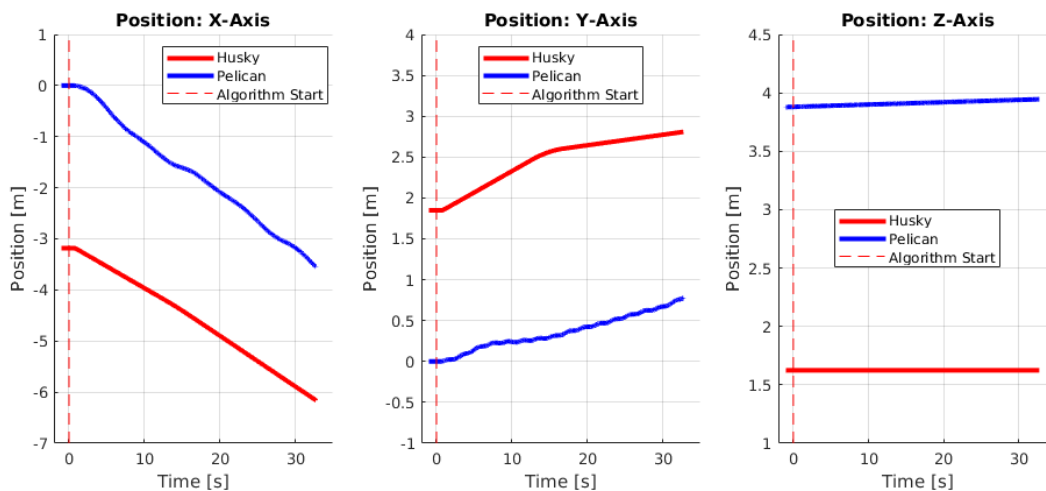


Figure 9. Position, in Gazebo's reference frame, of the Pelican UAV and Husky UGV over time during simulation, with k values of $k_g = 1.0$ and $k_{gx} = k_{xy} = 0.5$.

Finally, Simulation 4 is performed with the same values as Simulation 3 ($k_g = 0.4$ and $k_{gx} = k_{xy} = 0.5$). The position of the Pelican and the Husky can be seen in the graphs on Figure 10. In this simulation, just as in the previous one, the Pelican will follow the Husky while keeping its distance. During simulations, the Pelican and the Husky had a mean difference in distance of 2.48, 1.58 and 2.3 m in the x , y and z axes, respectively.

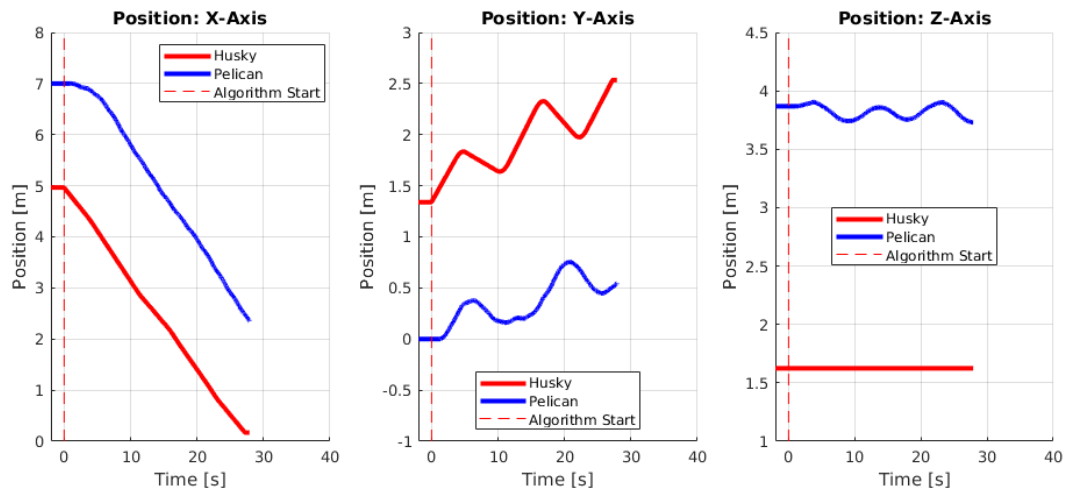


Figure 10. Position, in Gazebo’s reference frame, of the Pelican UAV and Husky UGV over time during simulation, with k values of $k_g = 1.0$ and $k_{gx} = k_{xy} = 0.5$.

8. Discussion

From the simulations, it is clear that the proposed Tau theory based strategy for the control scheme is flexible enough to achieve different types of near-ground manoeuvres. In simulations 1 and 2, the quadrotor successfully performed a the detection, tracking, and, eventual, landing on a moving platform with different touchdown speeds. Simulations 3 and 4 showcase the flexibility of Tau, where the quadrotor is capable of follow the platform and keep itself at a viewing distance from the target without initializing landing. All this experiments start form hover, a new addition that, to the knowledge of the authors, has never been used during Tau theory based visual autonomous landing.

9. Conclusions

This paper shows a bio-inspired controller using Tau theory to achieve flexible visual autonomous vertical and horizontal control of a multi-rotor vehicle. The simulations confirm that near-ground manoeuvres, such as landing, and tracking of an objective can be performed visually without knowledge of the vehicle’s height or objective’s velocity. Practical applications of this method include target approach to perform inspection, tracking or landing; followed by perching or fly away based on the chosen constant values. Practical applications of the preposed method can be expanded into other VTOL vehicles, UGV and even spacecraft. Further work is required to automate the choice of manoeuvre parameters based on the vehicle’s objective and context awareness.

Author Contributions: S.A.: analysis and design of control methods, programming and simulations, writing up. V.B.: conceptualization, supervision, proof reading, technical corrections; N.B.: supervision, proof reading, technical corrections.

Funding: The author Saul Armendariz is grateful for the funding from the “Consejo Nacional de Ciencia y Tecnologia” (CONACYT, Mexico). This research was funded by CONACYT, grant number 440239.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

EKF	Extended Kalman Filter
UKF	Unscented Kalman Filter
CKF	Cubature Kalman Filter
RMS	Root Mean Square
FOE	Focus of Expansion
GPS	Global Positioning System
VMP	Visual Motion Parameters
ROS	Robot Operating System
UGV	Unmanned Ground Vehicle
FoV	Field of View
TTC	Time-to-Contact
UAV	Unmanned Aerial Vehicle

Appendix A

Low-level controller state feedback and reference feed-forward gains for Equation (31).

$$K = \begin{bmatrix} 0 & 613.77 & -472.22 & 0 & 63.01 & -51.003 \\ -613.77 & 0 & 472.22 & -63.01 & 0 & 51.003 \\ 0 & -613.77 & -472.22 & 0 & -63.01 & -51.003 \\ 613.77 & 0 & 472.22 & 63.01 & 0 & 51.003 \end{bmatrix} \quad (\text{A1})$$

$$F = \begin{bmatrix} -211.61 & -54.09 & 0 & -21.52 \\ 211.61 & 0 & -54.05 & -21.52 \\ -211.61 & 54.09 & 0 & -21.52 \\ 211.61 & 0 & 54.05 & -21.52 \end{bmatrix} \quad (\text{A2})$$

High-Level PI Controller Parameters in Equation (37)

$$K_P = 2.788 \quad (\text{A3})$$

$$K_I = 0.067 \quad (\text{A4})$$

References

1. Gupte, S.; Mohandas, P.I.T.; Conrad, J.M. A survey of quadrotor unmanned aerial vehicles. In Proceedings of the 2012 Proceedings of IEEE Southeastcon, Orlando, FL, USA, 15–18 March 2012; pp. 1–6.
2. Kendoul, F. Survey of Advances in Guidance, Navigation, and Control of Unmanned Rotorcraft Systems. *J. Field Robot.* **2011**, *23*, 245–267. [[CrossRef](#)]
3. Zhang, Z.; Xie, P.; Ma, O. Bio-inspired trajectory generation for UAV perching. In Proceedings of the 2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics: Mechatronics for Human Wellbeing (AIM 2013), Wollongong, Australia, 9–12 July 2013; pp. 997–1002.
4. Falanga, D.; Zanchettin, A.; Simovic, A.; Delmerico, J.; Scaramuzza, D. Vision-based autonomous quadrotor landing on a moving platform. In Proceedings of the 15th IEEE International Symposium on Safety, Security and Rescue Robotics, Conference (SSRR 2017), Shanghai, China, 11–13 October 2017; pp. 200–207.
5. Horridge, G.A. The evolution of visual processing and the construction of seeing systems. *Proc. R. Soc. Lond. Ser. B* **1987**, *230*, 279–292. [[CrossRef](#)]
6. Srinivasan, M.; Zhang, S.; Lehrer, M.; Collett, T. Honeybee navigation en route to the goal: Visual flight control and odometry. *J. Exp. Biol.* **1996**, *199*, 237–244. [[CrossRef](#)] [[PubMed](#)]
7. Lee, D.N. A Theory of Visual Control of Braking Based on Information about Time-to-Collision. *Perception* **1976**, *5*, 437–459. [[CrossRef](#)] [[PubMed](#)]

8. Lee, D.N.; Young, D.S.; Rewt, D. How do somersaulters land on their feet? *J. Exp. Psychol.* **1992**, *18*, 1195. [[CrossRef](#)]
9. Lee, D.N. Guiding movement by coupling taus. *Ecol. Psychol.* **1998**, *10*, 221–250. [[CrossRef](#)]
10. Herisse, B.; Russotto, F.X.; Hamel, T.; Mahony, R. Hovering flight and vertical landing control of a VTOL Unmanned Aerial Vehicle using optical flow. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Nice, France, 22–26 September 2008.
11. Bresciani, T. Modelling, Identification and Control of a Quadrotor Helicopter. Master's Theses, Lund University, Lund, Sweden, 2008.
12. Alkowitz, M.T.; Becerra, V.M.; Holderbaum, W. Body-centric Modelling, Identification, and Acceleration Tracking Control of a Quadrotor UAV. *Int. J. Model. Ident. Control* **2014**. [[CrossRef](#)]
13. Stevens, B.L.; Lewis, F.L.; Johnson, E.N. *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*; Wiley: Hoboken, NJ, USA, 2016; p. 768.
14. Wang, J.H.; Sheng, J.; Tsai, H.; Chen, Y.C.; Guo, S.M.; Shieh, L.S. An Active Low-Order Fault-Tolerant State Space Self-Tuner for the Unknown Sample-Data Linear Regular System with an Input-Output Direct Feedthrough Term. *Appl. Math. Sci.* **2012**, *6*, 4813–4855.
15. Lee, D.N. General Tau Theory: Evolution to date. *Perception* **2009**, *38*, 837. [[CrossRef](#)] [[PubMed](#)]
16. Alkowitz, M.T.; Becerra, V.M.; Holderbaum, W. Estimation of Visual Motion Parameters Used for Bio-inspired Navigation. *J. Image Graphics* **2013**, *1*, 120–124. [[CrossRef](#)]
17. Menold, P.H.; Pearson, R.K.; Allgower, F. Online outlier detection and removal. In Proceedings of the 7th IEEE Mediterranean Conference on Control and Automation (MED '99), Haifa, Israel, 28–30 June 1999; pp. 1110–1133.
18. Harris, L.R.; Jenkin, M.R.; Zikovitz, D.; Redlick, F.; Jaekl, P.; Jasiobedzka, U.T.; Jenkin, H.L.; Allison, R.S. Simulating Self-Motion I: Cues for the Perception of Motion. *Virtual Real.* **2002**, *6*, 75–85. [[CrossRef](#)]
19. Hain, T.C.; Helminski, J.O. Anatomy and physiology of the normal vestibular system. *Vestibular Rehabil.* **2007**, *11*, 2–18.
20. Arasaratnam, I.; Haykin, S.; Hurd, T.R. Cubature Kalman filtering for continuous-discrete systems: Theory and simulations. *IEEE Trans. Signal Process.* **2010**, *58*, 4977–4993. [[CrossRef](#)]
21. Wan, E.A.; Van Der Merwe, R. The unscented Kalman filter for nonlinear estimation. *Technology* **2000**, 153–158. [[CrossRef](#)]
22. Dai, H.-D.; Dai, S.-W.; Cong, Y.-C.; Wu, G.-B. Performance Comparison of EKF/UKF/CKF for the Tracking of Ballistic Target. *Telecommun. Comput. Electron. Control.* **2012**, *10*, 1537–1542.
23. Chandra, K.; Gu, D. Cubature Kalman Filter based Localization and Mapping. In Proceedings of the 18th IFAC World Congress, Milano, Italy, 28 August–2 September 2011; pp. 2121–2125.
24. Wang, J.; Olson, E. AprilTag 2: Efficient and robust fiducial detection. *IEEE Int. Conf. Intell. Robots Syst.* **2016**, 4193–4198. [[CrossRef](#)]
25. Available online: <http://www.ros.org/> (accessed on 29 December 2018).
26. Available online: <http://gazebosim.org/> (accessed on 29 December 2018).
27. Furrer, F.; Burri, M.; Achtelik, M.; Siegwart, R. RotorS—A Modular Gazebo MAV Simulator Framework. In *Robot Operating System (ROS): The Complete Reference (Volume 1)*; Koubaa, A., Ed.; Springer International Publishing: Cham, Switzerland, 2016; pp. 595–625.
28. Available online: <https://opencv.org/> (accessed on 29 December 2018).
29. Available online: <http://eigen.tuxfamily.org/> (accessed on 29 December 2018).
30. Available online: <http://www.clearpathrobotics.com/assets/guides/husky/> (accessed on 29 December 2018).
31. Available online: <http://higginsandsons.com/astro/> (accessed on 29 December 2018).

