Bioinspiration & Biomimetics

PAPER

UAS stealth: target pursuit at constant distance using a bio-inspired motion camouflage guidance law

To cite this article: Reuben Strydom and Mandyam V Srinivasan 2017 *Bioinspir. Biomim.* **12** 055002

View the article online for updates and enhancements.

Related content

- Bio-inspired vision based robot control using featureless estimations of time-to-contact
  Haijie Zhang and Jianguo Zhao
- An autonomous robot inspired by insect neurophysiology pursues moving features in natural environments
  Zahra M Bagheri, Benjamin S Cazzolato, Steven Grainger et al.
- A biomimetic vision-based hovercraft accounts for bees' complex behaviour in various corridors
  Frédéric L Roubieu, Julien R Serres, Fabien Colonnier et al.

This content was downloaded from IP address 130.102.42.98 on 31/10/2017 at 02:13

Bioinspiration & Biomimetics

PAPER

UAS stealth: target pursuit at constant distance using a bio-inspired motion camouflage guidance law

RECEIVED
25 November 2016

REVISED
2 July 2017

ACCEPTED FOR PUBLICATION
4 July 2017

PUBLISHED
20 September 2017

Reuben Strydom and Mandyam V Srinivasan

The Queensland Brain Institute and the School of Information Technology and Electrical Engineering at The University of Queensland, Brisbane, Australia

E-mail: r.strydom@uq.edu.au

Keywords: motion camouflage, vision-based guidance, stealth, unmanned aircraft systems (UAS), pursuit, interception

Supplementary material for this article is available online

Abstract

The aim of this study is to derive a guidance law by which an unmanned aerial system(s) (UAS) can pursue a moving target at a constant distance, while concealing its own motion. We derive a closed-form solution for the trajectory of the UAS by imposing two key constraints: (1) the shadower moves in such a way as to be perceived as a stationary object by the shadowee, and (2) the distance between the shadower and shadowee is kept constant. Additionally, the theory presented in this paper considers constraints on the maximum achievable speed and acceleration of the shadower. Our theory is tested through Matlab simulations, which validate the camouflage strategy for both 2D and 3D conditions. Furthermore, experiments using a realistic vision-based implementation are conducted in a virtual environment, where the results demonstrate that even with noisy state information it is possible to remain well camouflaged using the constant distance motion camouflage technique.

1. Introduction

Currently, there is growing interest in increasing the autonomy of robots and unmanned vehicles. To this end, the field of aerial navigation and guidance has adopted numerous strategies inspired by nature, such as optic flow [23], view-based snapshot localisation [9] and simultaneous localisation and mapping (SLAM) [16]. Although the robotics community has benefited greatly from biology thus far, there are still many lessons to learn through studying nature.

Animals, in particular insects, have demonstrated an incredible ability to navigate through complex environments, pursue mates and even camouflage their motion from the eyes of potential prey. 'Motion camouflage'—a term coined by Srinivasan and Davey [22]—is the principle where a shadower (the agent chasing) conceals its motion with respect to a shadowee (the agent being chased). The shadower accomplishes visual motion camouflage by moving in such a way as to make the motion of its image indistinguishable from the motion of the image of a stationary object in the retina of the moving shadowee. Behavioural studies by Mizutani, *et al* [17] have shown that dragonflies use motion camouflage while pursuing prey, even with their limited sensory and computational resources. Although motion camouflage is simple in concept, Anderson and McOwan [4] demonstrated that even humans, with our advanced visual system, can be fooled by this clever technique.

Since the majority of animals rely on vision as their primary sense, the robotics community (especially UAS research) has in recent years focused on vision-based strategies for navigation, situational awareness and guidance. Cameras are multi-purpose sensors that provide an abundance of information with the added advantages of being low cost, low power, and small in form-factor.

When one pairs vision-based sensing with the rapidly increasing popularity of UAS, Motion Camouflage provides the potential for numerous innovative military (e.g. surveillance, pursuit and interception, and target following) and civilian applications (e.g. law enforcement surveillance, animal tracking, and sports footage). In particular, surveillance tasks and video capture of sporting activities would benefit from a constant distance following technique that visually conceals the motion of the UAS.

There are a number of potential applications for the use of motion camouflage, ranging from defence-related applications, through surveillance to unobtrusive animal observation. However, the research

© 2017 IOP Publishing Ltd

community has paid relatively little attention to this concept thus far. We present a novel bio-inspired motion camouflage guidance law, which is developed theoretically, and tested in simulation in Matlab, as well as on a UAS platform in a virtual environment. This motion camouflage strategy provides a closed-form solution for guidance of the pursuer, by specifying the distance and direction of the motion to be executed by the shadower at each time-step. This is accomplished by satisfying the following two constraints: (1) the shadower moves in such a way as to be perceived as a stationary object on the shadowee's retina (image frame), and (2) the distance between the shadower and shadowee is kept constant throughout the pursuit. Furthermore, a theoretical framework is developed to ensure that the trajectory of the shadower is constrained to its specified maximum achievable speed and acceleration.

The motivation for a constant distance motion camouflage (CDMC) guidance law stems from the fact that many monocular machine vision algorithms (e.g. for the detection of collision avoidance) are designed to be sensitive to looming (expansion) cues. Thus, even if the traditional motion camouflage constraint—as defined by Srinivasan and Davey [22]—is obeyed, the shadowee may detect the expansion or contraction of the shadower's image—thus disrupting the shadower's visual concealment.

The outline of this paper is as follows: firstly, section 2 discusses related work in the field. Section 3 presents the problem description, assumptions and theoretical framework for the motion camouflage law presented here, including the guidance constraints for the aircraft (e.g. maximum acceleration and achievable velocities). Section 4 outlines the virtual environment used to test a realistic vision-based implementation of the CDMC algorithm on a rotorcraft UAS. Details of the control architecture and vision-based implementation are given in sections 4.1 and 4.2. The results are outlined in sections 5—5.1 validates the methods using Matlab simulations and section 5.2 describes the results of testing the rotorcraft in a virtual environment. Finally, the discussion and conclusions follow in section 6 and section 7, respectively.

2. The motion camouflage principle and related work

The principles of motion camouflage were first set out by Srinivasan and Davey [22]. That study presents the basic concept of motion camouflage where a shadower 'can camouflage its motion by emulating a stationary object at a point F' [22], where F, in this case, is an arbitrarily chosen fixed point, as illustrated in figure 1. In that study, four scenarios are considered: when *F* is in front of the shadower, behind the shadower, at the shadower's starting location, or at infinity (the last scenario also coincides with the strategy of Constant Bearing interception, if the shadower closes in on the shadowee [8, 18]). Indeed, these motion camouflage strategies can be exploited by an agent for moving toward a target while appearing to be static from the target's point of view, with the exception of generating image expansion cues that can be detected by the target as the agent moves closer [12].

The use of motion camouflage by flying insects is further supported by the study of the behaviour of aggressive bees when they inspect or pursue moving targets, as detailed in the supplementary materials (stacks.iop.org/BB/12/055002/mmedia). If it is assumed that the shadower is a point object, i.e. if the shadowee is not sensitive to looming (image expansion) cues generated by the shadower, then the shadower can camouflage its motion by staying on the camouflage constraint lines (the lines connecting the instantaneous positions of the shadowee with the fixed point). Treating the shadower as a point object may be a valid assumption for a small insect such as the aforementioned honeybee. However, following such a strategy poses two problems for robotic systems: (1) there is ambiguity to the motion camouflage solution, that is, there are an infinite number of trajectories that satisfy the motion camouflage constraint lines (of which only 3 are shown in figure 1 for each scenario); and (2) robots are, in general, too large to be considered as point objects, and can therefore break the camouflage if they generate looming cues by closing in on the shadowee. To overcome the two limitations, a second constraint is imposed, namely, that the shadower pursues the shadowee at a constant distance.

Executing motion camouflage per se does not require knowledge of Cartesian prey velocity and distance to prey. All that one needs to do is line the target up against a landmark on the ground (Srinivasan and Davey [22]). Of course, we do not know if insects actually do it this way but the observation that insects perform motion camouflage has inspired a number of studies to look to biology to provide an unambiguous solution of the motion camouflage strategy for robotic platforms [6, 7, 15]. To provide more realistic guidance laws, motion camouflage has also been analysed in the presence of sensorimotor delay [20, 24]. Although extensive research has been conducted on the use of motion camouflage in nature and in defining the basic theory, its implementation in robotics has been limited.

While motion camouflage is in principle well defined, it is challenging to provide control laws suitable for robots, such as unmanned ground vehicle(s) (UGV) and UAS. Anderson and McOwan [3] stipulated two motion camouflage algorithms. The first is a responsive algorithm, which reacts to the last known state of the target, and the second is a predictive algorithm that requires some prior knowledge of the target's motion. In both cases the algorithm by Anderson and McOwan [3] requires knowledge of the shadowee's direction, the direction of motion of the image, and an accurate path-integrated estimate of the shadower's current position

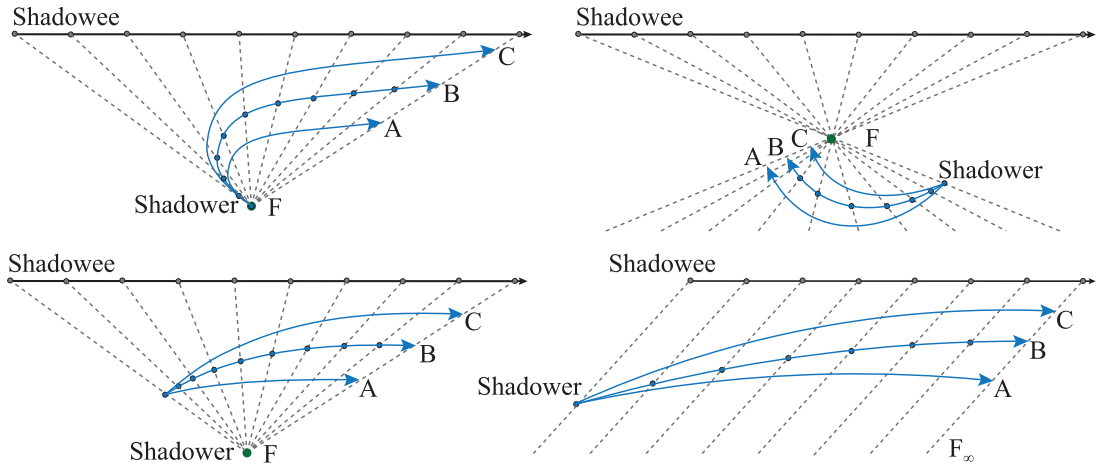


Figure 1. Four motion camouflage scenarios—Adapted with permission from [22]. $\vec{F}$ is an arbitrarily chosen fixed point. In each scenario A, B and C represent three possible camouflage trajectories.

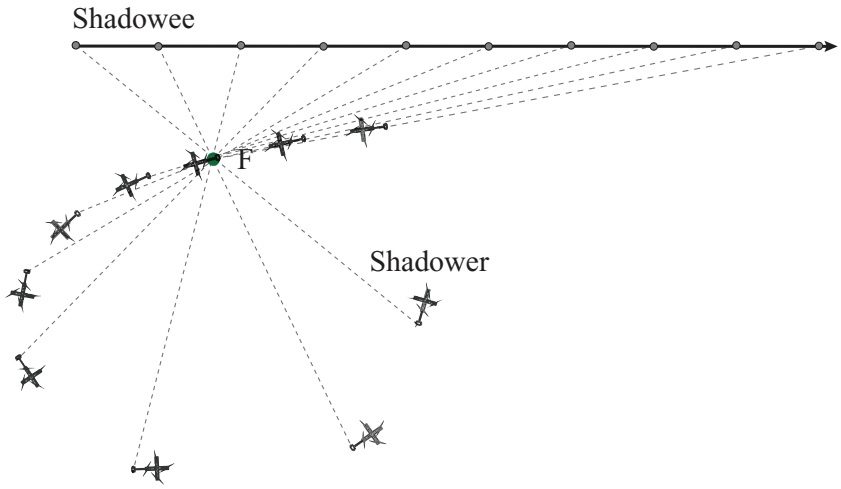


Figure 2. Illustration of CDMC.

in order to generate a guidance command that ensures that the shadower remains on the (continually changing) camouflage constraint line. Although their strategy is an important step towards autonomous motion camouflage, this method has not been implemented and tested on a robotic platform. Ranó and Iglesias [19] present a guidance law utilising a non-linear polynomial controller that learns through the input of motion camouflage data—constructed using a computationally expensive heuristic inspired by the Dynamic Window Approach by Thrun, *et al* [10]. The study by [19] validates their motion camouflage technique with experiments conducted on a UAS. It was found, however, that the performance of the utilised heuristic implementation degraded as the UAS approached the target. Furthermore, depending on the application, this solution may produce undesirable trajectories, as it learns from a number of data sets to define the motion camouflage trajectory. Therefore, the exact path taken by the shadower may be unpredictable.

As well as providing a novel control law for motion camouflage at a constant pursuit distance, the next section will derive the requisite velocity and acceleration profiles of the shadower, which are then used to choose the fixed point based on the control constraints of the UAS (maximum achievable speed and acceleration).

3. Constant distance motion camouflage in 2D and 3D

This section presents the theoretical derivations for the proposed CDMC strategy. Figure 2 shows a planar view of a potential trajectory, and figure 3 illustrates the geometry of the approach.

3.1. Problem description and assumptions

As mentioned previously, motion camouflage generally only constrains the quadrotor to stay on the line passing through both the target and the fixed point (from now on, 'quadrotor' and 'target', and 'shadower' and 'shadowee' will respectively be used interchangeably). However, this broad definition of motion camouflage leaves open an infinite number

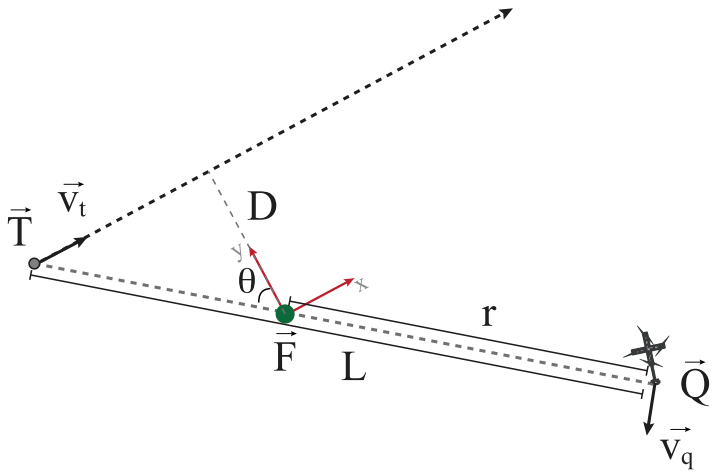


Figure 3. 2D geometry for CDMC. Note that $\vec{T} = (x_t,\ y_t)$ and $\vec{Q} = (x_q, y_q)$. $\vec{V}_t$ and $\vec{V}_q$ represent the velocities of the target and the quadrotor, respectively.

of potential positions that the shadower can occupy along each constraint line. On the other hand, moving a vehicle in such a way as to camouflage its motion requires a specific control command to be issued at each time step—usually a position, velocity or acceleration command. In this proposed strategy, the trajectory of the quadrotor is defined by two constraints:

- The quadrotor should always be positioned on the motion constraint line, defined by the position of the fixed point (F) and target's current position.
- The distance between the target and the quadrotor is kept constant.

These two constraints, along with the assumptions set out below, provide a trajectory with a defined set of commands for the quadrotor to follow from one time step to the next.

There are four assumptions that are made to compute the quadrotor's position, velocity and acceleration throughout such a trajectory. These assumptions are essential for a rational choice of the location of the fixed point, which is the initial step in computing the quadrotor's path. The assumptions are:

(i) The target is moving at a constant velocity.
(ii) The distance to the target is known or can be measured.
(iii) The quadrotor's state (position, velocity and attitude) are known.
(iv) The quadrotor's maximum velocity is greater than that of the target (this is necessary for achieving pursuit at a constant distance).

The above four assumptions provide the necessary information to determine our CDMC strategy, and compute the position, velocity and acceleration profiles of the shadower, thereby determining the appropriate location of the fixed point, given the quadrotor's velocity and acceleration constraints. As derived later in this section, the location of the fixed point determines the position, speed and acceleration profiles of the shadower (and therefore the maximum speed and acceleration). Therefore, the location of the fixed point is chosen so as to ensure that the shadower does not exceed its speed and/or acceleration constraints. It should be noted that the first assumption—constant velocity motion of the target—is only a requirement for pre-computing an optimal location of the fixed point that is consistent with the maximum achievable acceleration and velocity of the UAS. If this assumption is relaxed by requiring only a constant direction but a non-constant speed, the CDMC is still achievable, however, the quadrotor's maximum acceleration and speed throughout the trajectory will be unpredictable and if the velocity or acceleration required from the quadrotor exceeds its capabilities, then the motion camouflage may need to be terminated. In this case, conservative acceleration and velocity thresholds should be applied, which are below the actual maxima of the quadrotor's capabilities.

The following sections provide a theoretical framework and analysis of the proposed CDMC strategy. The derivations are conducted in 2D, however, as will be shown in section 3.7, they also hold for 3D motion camouflage. The reason that the 2D derivation holds for the 3D solution is that the quadrotor's trajectory continues to be constrained by a 2D plane; i.e. a plane defined by the fixed point and the target's linear trajectory.

For a given initial position of the shadower, and a target moving at a known (or inferred) velocity, it is possible to derive the trajectory that the shadower should describe if it is to pursue the target: (i) at a constant distance, and (ii) by camouflaging its own motion, through emulating a stationary object at the prescribed fixed point.

3.2. Geometry of CDMC in 2D

From figure 3, the quadrotor's positional components (x_q, y_q), with respect to the fixed point, can be represented by (1) and (2) in Cartesian coordinates (Cartesian coordinates are used rather than polar coordinates as it simplifies the analysis of the acceleration profile):

$$x_q = r\sin(\theta) \tag{1}$$

$$y_q = -r\cos(\theta) \tag{2}$$

where

$$r = \frac{D}{\cos(\theta)} - L. \tag{3}$$

The rate of change of θ and r will be important for the derivation of the quadrotor's speed and acceleration profiles in the later section—note: D and L are constants; and r is a signed distance, and a change in the polarity of r means that the quadrotor has passed through the fixed point, which will be discussed later. The rate of change of θ can be derived from:

$$\frac{x_t}{D} = \tan(\theta).$$

We define the speed of the target as:

$$\begin{aligned} S_t &= \|\vec{V_t}\| \triangleq \|\frac{\mathrm{d}x_t}{\mathrm{d}t}\| \\ &= \|D\sec^2(\theta)\frac{\mathrm{d}\theta}{\mathrm{d}t}\|. \end{aligned} \tag{4}$$

Therefore, $\frac{\mathrm{d}\theta}{\mathrm{d}t} = \frac{S_t}{D}\cos^2(\theta)$
and the rate of change of r can be derived as:

$$\begin{aligned} \frac{\mathrm{d}r}{\mathrm{d}t} &= \frac{\mathrm{d}r}{\mathrm{d}\theta}\frac{\mathrm{d}\theta}{\mathrm{d}t} \\ &= S_t\sin(\theta). \end{aligned} \tag{5}$$

3.3. Derivation of the Quadrotor's speed profile

The quadrotor's speed in the x direction can be computed by differentiating (1):

$$\begin{aligned} \frac{\mathrm{d}x_q}{\mathrm{d}t} &= r(\cos(\theta))\frac{\mathrm{d}\theta}{\mathrm{d}t} + \sin(\theta)\frac{\mathrm{d}r}{\mathrm{d}t} \\ &= S_t\left(1 - \frac{L}{D}\cos^3(\theta)\right) \end{aligned} \tag{6}$$

and the speed in the y direction can be computed by differentiating (2):

$$\begin{aligned} \frac{\mathrm{d}y_q}{\mathrm{d}t} &= r(\sin(\theta))\frac{\mathrm{d}\theta}{\mathrm{d}t} - \cos(\theta)\frac{\mathrm{d}r}{\mathrm{d}t} \\ &= -\frac{S_t}{2}\frac{L}{D}\cos(\theta)\sin(2\theta). \end{aligned} \tag{7}$$

Finally, the quadrotor's speed (S_q) can be expressed as:

$$\begin{aligned} S_q^2 &= \left(\frac{\mathrm{d}x_q}{\mathrm{d}t}\right)^2 + \left(\frac{\mathrm{d}y_q}{\mathrm{d}t}\right)^2 \\ &= S_t^2\left[\frac{L^2}{D^2}\cos^4(\theta) - \frac{2L}{D}\cos^3(\theta) + 1\right]. \end{aligned} \tag{8}$$

3.4. Quadrotor's speed at extrema

To determine whether the quadrotor is capable of performing the desired trajectory as defined by the above equations—in the light of the maximum speed that it can achieve—the maximum value of the requisite speed profile is required. The extrema (maximum and minimum) of (8) occur where $\mathrm{d}S_q^2/\mathrm{d}\theta = 0$. Differentiating (8) with respect to θ yields:

$$\frac{2L}{D}\cos^2(\theta)\sin(\theta)\left[3 - \frac{2L}{D}\cos(\theta)\right] = 0 \tag{9}$$

(9) is valid when:

(i) $\sin(\theta) = 0$, that is $\theta = 0$
(ii) $\cos(\theta) = 0$, that is $\theta = \frac{\pi}{2}$
(iii) $\cos(\theta) = \frac{3D}{2L}$, which only gives a real solution for $\frac{D}{L} < \frac{2}{3}$.

Note: here θ has a range between $-\frac{\pi}{2} \leqslant \theta \leqslant \frac{\pi}{2}$. To check whether the extremum as defined by (8) is a maximum or minimum, the polarity of $\mathrm{d}^2S_q^2/\mathrm{d}\theta^2$ is evaluated. Computing $\mathrm{d}^2S_q^2/\mathrm{d}\theta^2$ from the left hand side (LHS) of (9) simplifies to:

$$\begin{aligned} \frac{\mathrm{d}^2S_q^2}{\mathrm{d}\theta^2} \propto \frac{2L}{D}\Bigg[&3\cos^3(\theta) - \frac{2L}{D}\cos^4(\theta) \\ &- 6\cos(\theta)\sin^2(\theta) \\ &+ \frac{6L}{D}\cos^2(\theta)\sin^2(\theta)\Bigg]. \end{aligned} \tag{10}$$

Let us examine the polarity of (10) for each of the three extrema indicated above.

(i) *Case 1:* $\sin(\theta) = 0; \cos(\theta) = 1$

$$\frac{\mathrm{d}^2S_q^2}{\mathrm{d}\theta^2} = \frac{2L}{D}\left[3 - \frac{2L}{D}\right]. \tag{11}$$

From (11) it is possible to determine that:

$$\frac{\mathrm{d}^2S_q^2}{\mathrm{d}\theta^2} \text{ will} \begin{cases} \text{be positive if } \dfrac{D}{L} > \dfrac{2}{3} \text{ leading to} \\ \text{a minimum of the speed profile.} \\ \text{be negative if } \dfrac{D}{L} < \dfrac{2}{3} \text{ leading to} \\ \text{a maximum of the speed profile.} \end{cases} \tag{12}$$

Speed at this extremum, from (8):

$$\begin{aligned} S_q^2 &= S_t^2\left[\frac{L^2}{D^2} - \frac{2L}{D} + 1\right] \\ S_q &= S_t\left|\frac{L}{D} - 1\right| \end{aligned} \tag{13}$$

(ii) *Case 2:* $\cos(\theta) = 0; \sin(\theta) = 1.$

This gives:

$$\frac{\mathrm{d}^2S_q^2}{\mathrm{d}\theta^2} = 0. \tag{14}$$

In this case the shadower's speed (S_q) asymptotically approaches the value S_t as $t \to \infty$:

$$\lim_{t\to\infty} S_q = S_t. \tag{15}$$

(iii) *Case 3:* $\cos(\theta) = \frac{3D}{2L}$

This is possible only if D/L $<$ 2/3, which gives:

$$S_q^2 = S_t^2\left[\frac{L^2}{D^2}\frac{81}{16}\frac{D^4}{L^4} - \frac{2L}{D}\frac{27}{8}\left(\frac{D}{L}\right)^3 + 1\right]$$

$$S_q = S_t\sqrt{1 - \frac{9}{4}\frac{D^2}{L^2}}, \text{ when } \frac{D}{L} < \frac{2}{3}. \tag{16}$$

3.5. Derivation of the Quadrotor's acceleration profile

Next, the acceleration profile is derived for the shadower. Differentiating (6) with respect to time gives the acceleration in the x direction:

$$\frac{d^2x_q}{dt^2} = 3L\left(\frac{S_t}{D}\right)^2 \sin(\theta)\cos^4(\theta). \tag{17}$$

Differentiating (7) with respect to time provides the acceleration in the y direction:

$$\frac{d^2y_q}{dt^2} = \frac{L}{2}\left(\frac{S_t}{D}\right)^2 \cos^3(\theta)\left[1 - 3\cos(2\theta)\right]. \tag{18}$$

The magnitude of the absolute acceleration can be expressed as:

$$A_q = \sqrt{\left(\frac{d^2x_q}{dt^2}\right)^2 + \left(\frac{d^2y_q}{dt^2}\right)^2}. \tag{19}$$

Substituting from (18) and (17) this can be written as:

$$A_q = L\left(\frac{S_t}{D}\right)^2 \cos^3(\theta)\sqrt{\frac{5 - 3\cos(2\theta)}{2}}. \tag{20}$$

Note that in (20) the shape of the acceleration profile is independent of (S_t/D), although it is scaled by it. This means that the locations of the extrema (of the maximum and minimum values) will occur at the same values of θ, irrespective of the values of S_t or D. It can be shown that A_q is maximum at $\theta = 0$ and minimum occurs at $\theta = \pm\pi/2$. Inserting $\theta = 0$ in (20), we obtain the value for the maximum acceleration as:

$$A_{q_{\max}} = L\left(\frac{S_t}{D}\right)^2. \tag{21}$$

Therefore, as the fixed point moves further away from the target (and thus increasing D) the quadrotor's acceleration is reduced.

3.6. Determining the fixed point

One key remaining requirement that has not previously been considered in the literature is the choice of the location of the fixed point (F).

The first constraint for locating the fixed point is that it must lie on the line connecting the target and the quadrotor (or on the extension of this line, as illustrated in figure 1). The question posed, however, is where should $\vec{F}$ be placed on the initial line connecting the target to the quadrotor?

It is evident from figure 3 that the location of the fixed point on this initial line can be defined uniquely by its perpendicular distance, D, to the target's trajectory. Conveniently, D is the only independent variable in the calculation of the quadrotor's speed and acceleration. This is due to the fact that the quadrotor cannot control the target's velocity (V_t), and L is a constant (usually the initial distance between the target and quadrotor). With this knowledge, the known quadrotor limitations can be used to compute the location of $\vec{F}$ to ensure that the maximum speed and acceleration throughout the quadrotor's trajectory are achievable (if assumption 1 in section 3.1 holds). A two-step process is followed to determine the placement of F. First, compute D for the maximum quadrotor acceleration, using (22) below. Secondly, to ensure that the maximum velocity is within the limits of the quadrotor, substitute D (computed from (22)) into equations (13) and (16). If the trajectory defined by $D_{\text{AccLimited}}$ requires speeds greater than the quadrotor's limitations, then recompute D to satisfy the velocity constraint ($D_{\text{SpeedLimited}}$), as specified by (23).

If the maximum acceleration is constrained to be $A_{q_{\max}}$, we can calculate the required D from (21) as:

$$D_{\text{AccLimited}} = \sqrt{\frac{LS_t^2}{A_{q_{\max}}}}. \tag{22}$$

Once D is computed from (22), the speed limits are checked to ensure that they do not exceed the speed extremum cases 1 and 2 as specified by (13) and (16) respectively. If the calculated speeds are higher than the maximum achievable quadrotor's speed ($s_{\max}$), D is recomputed by rearranging (13) or (16) (depending on which case provides a higher maximum speed), as shown by (23) below. Therefore, the appropriate value of D is then given by:

$$D_{\text{SpeedLimited}} = \begin{cases} \dfrac{L}{\left(\frac{s_{\max}}{S_t} + 1\right)} & \text{if case 1 provides} \\ & \text{max speed for } D_{\text{AccLimit}} \\ L\sqrt{\dfrac{16}{27}\left(1 - \dfrac{s_{\max}}{S_t}\right)} & \text{if case 3} \\ & \text{provides max speed for} \\ & D_{\text{AccLimit}} \end{cases}. \tag{23}$$

Recomputing D in this way then constrains the maximum quadrotor's speed and acceleration throughout the trajectory to the desired limits. If the speed of the target is greater than that of the quadrotor, then the shadower cannot pursue it at a constant distance, let alone execute motion camouflage.

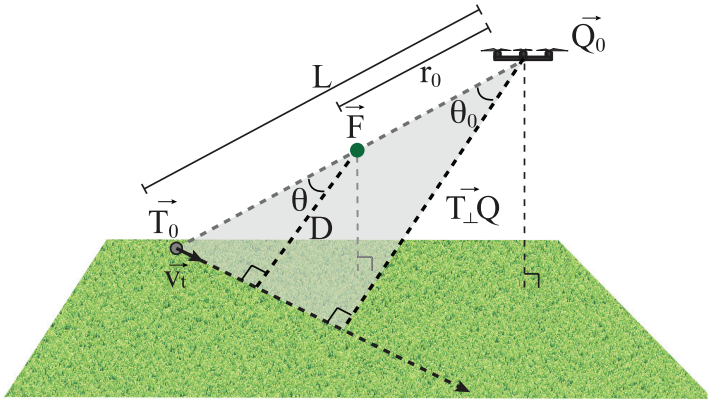


Figure 4. Geometry for CDMC in 3D at $t = 0$.

3.7. Constant distance motion camouflage in 3D

Here, the problem of executing CDMC is extended to 3D, as illustrated in figure 4. The steps are fundamentally similar to the 2D case:

(i) Measure the distance between the quadrotor and the target (L).
(ii) Estimate the target's velocity (V_t) and the distance between the quadrotor and the target (L).
(iii) Compute D, defined by the speed and acceleration constraints.
(iv) Determine the location of the fixed point (F).

The behaviour of the 3D trajectory is similar to that of the 2D case, as it is still confined to a 2D plane that is defined by the location of the fixed point and the target's trajectory, as shown by the shaded plane in figure 4. Therefore, the derivation for the optimal placement of $\vec{F}$ also applies to a 3D situation.

3.7.1. *Computing the location of $\vec{F}$ for the 3D case*

D can be computed in 3D in a similar manner as in the 2D case. Referring to figure 4, the fixed point location is computed as:

$$\begin{aligned}\vec{F} &= \vec{Q}_0 + \frac{\vec{T}_0 - \vec{Q}_0}{\|\vec{T}_0 - \vec{Q}_0\|} r_0 \\ &= \vec{Q}_0 + \frac{\vec{T}_0 - \vec{Q}_0}{\|\vec{T}_0 - \vec{Q}_0\|}\left[\left(\frac{D}{\cos(\theta_0)}\right) - L\right] \end{aligned} \quad (24)$$

where

$$\theta_0 = \cos^{-1}\left[\frac{\vec{T_\perp Q}\cdot(\vec{T}_0 - \vec{Q}_0)}{\|\vec{T_\perp Q}\|\|\vec{T}_0 - \vec{Q}_0\|}\right] \quad (25)$$

and

$$\vec{T_\perp Q} = \vec{V_t}\frac{(\vec{Q}_0 - \vec{T}_0)\cdot\vec{V_t}}{\|V_t\|^2} - (\vec{Q}_0 - \vec{T}_0). \quad (26)$$

3.7.2. *Computing the Quadrotor's 3D trajectory*

Once location of the fixed point $\vec{F}$ is determined, the quadrotor's position as a function of time (t) can be represented as follows:

$$\vec{Q}(t) = \vec{T}(t) + \frac{\vec{F} - \vec{T}(t)}{\|\vec{F} - \vec{T}(t)\|}L. \quad (27)$$

The next position of the quadrotor is calculated by inserting the next position of the target into (27):

$$\begin{aligned}\vec{Q}(t+1) = [\vec{T}(t) + \vec{V_t}(t)\Delta t] \\ + \frac{\vec{F} - [\vec{T}(t) + \vec{V_t}(t)\Delta t]}{\|\vec{F} - [\vec{T}(t) + \vec{V_t}(t)\Delta t]\|}L.\end{aligned} \quad (28)$$

The final step is to compute the required velocity command to send to the flight controller. This is the difference between the predicted position at the next time step and the current position, divided by the time step, as shown in (29):

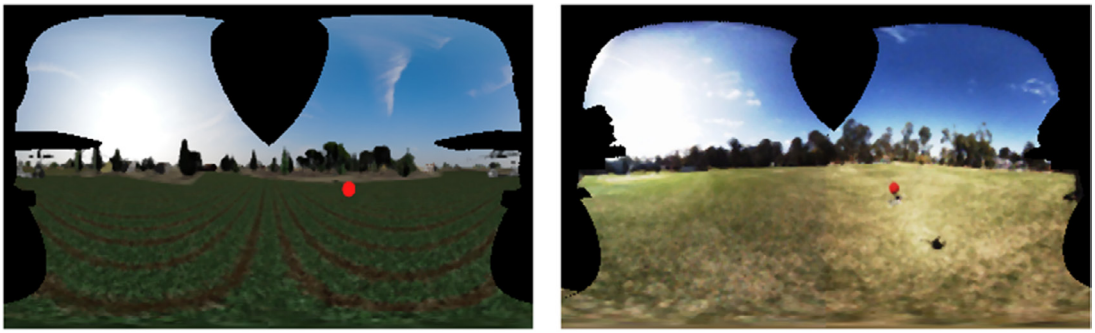
$$\vec{V_q}(t+1) = \frac{\vec{Q}(t+1) - \vec{Q}(t)}{\Delta t}. \quad (29)$$

To validate the derivation of the CDMC guidance law Matlab simulations were conducted—see section 5.1 for the results of these simulations. Furthermore, a realistic vision-based implementation is tested in a virtual environment—the next section provides the implementation details and the experimental results are demonstrated in section 5.

4. Implementation in a realistic UAS virtual environment

To test our CDMC strategy under more realistic conditions, one needs to incorporate aircraft dynamics, sensorimotor delay and image noise (when using a vision sensor). The open scene graph (OSG) virtual environment was selected for this purpose. The OSG environment is used to realistically simulate the custom built quadrotor platform and unique bio-inspired vision sensor used for the outdoor field tests as outlined in [23].

The quadrotor implementation in the OSG environment utilises a realistic flight dynamics model—similar to that discussed in [2]—where the coefficients and constants have been chosen to be representative of the real platform. These include the



(a) (b)

Figure 5. A video frame captured by the onboard bio-inspired vision sensor in (a) the OSG virtual environment and (b) real outdoor field tests.

mass, moments of inertia, drag coefficients, propeller diameter and sensorimotor delays. Furthermore, the high-level autopilot architecture and image processing algorithms are identical for the simulated and real platforms to ensure that the OSG environment will provide similar results compared to real outdoor field tests (excluding external disturbances such as wind and illumination changes). The frame rate used to update the virtual camera system is 25 Hz, which is the frame rate of the vision sensor onboard the outdoor experimental platform. Figure 5 demonstrates the similarities between the simulated environment and the the real platform by showing a comparison between the real and simulated onboard views.

The advantages of testing the algorithm in the virtual environment are: (1) it provides a safe environment in which to test new control algorithms, (2) a perfect ground truth is available, and (3) no external disturbances (e.g. illumination differences, wind, etc) influence the control strategy.

The next section will describe the high-level control architecture used in the virtual environment.

4.1. Control architecture

The structure implemented to control the quadrotor UAS uses either a single velocity proportional-integral-derivative (PID) controller or a cascaded position and velocity PID architecture, as shown in figure 6. The functions of the two different structures are to provide flexibility between using waypoint-based control during take-off, landing and general navigation; and velocity only control when the CDMC algorithm is in operation.

4.1.1. Position-based control structure

In this study, the position-based navigation controller is used for take-off, landing and navigation when the CDMC algorithm is offline. A 3D position setpoint, and the path integrated optic-flow-based position estimate, constitute the inputs to the position PID controller, where the discrepancy between the setpoint and measured position is used to output a 3D velocity command. The velocity command is then compared to the actual (vision-estimated) velocity to compute a velocity error signal that determines the roll, pitch and throttle commands that are required to drive the velocity error to zero. These throttle commands, in combination with the quadrotor's flight dynamics, govern the motion of the aircraft.

4.1.2. CDMC control structure

The CDMC control structure utilises the velocity PID controller but forgoes the need for the first-stage position-based controller. In this case, the CDMC algorithm defines a velocity setpoint. To determine the required velocity setpoint, first, the vision-based egomotion estimate and target state are provided to the CDMC algorithm. The CDMC algorithm then computes the required velocity, which is fed into the velocity PID controller. Finally, the velocity controller outputs the necessary roll, pitch and throttle commands, which determine the motion of the UAS.

4.2. Vision-based implementation details

To accomplish CDMC in reality, three main processes are required: (1) estimation of UAS state (i.e. position, velocity and attitude); (2) object detection; and (3) target state estimation. Furthermore, due to the noisy measurements that are used to compute the target state, a final post-processing step is employed.

Note that in this work the implementation of the CDMC algorithm is a vision-based solution to demonstrate that motion camouflage is possible with a low-cost camera setup. Moreover, note that the purpose is to demonstrate that the CDMC guidance law achieves effective motion camouflage by concealing the motion of the shadower. The aim is not to demonstrate, or to achieve state of the art, in egomotion estimation or target detection.

4.2.1. Egomotion estimation and path integration

A vision-based approach is used to determine the state (position and velocity) of the rotorcraft. Optic flow—the relative motion that an observer experiences when moving through the environment—is used to estimate the egomotion of the aircraft (the

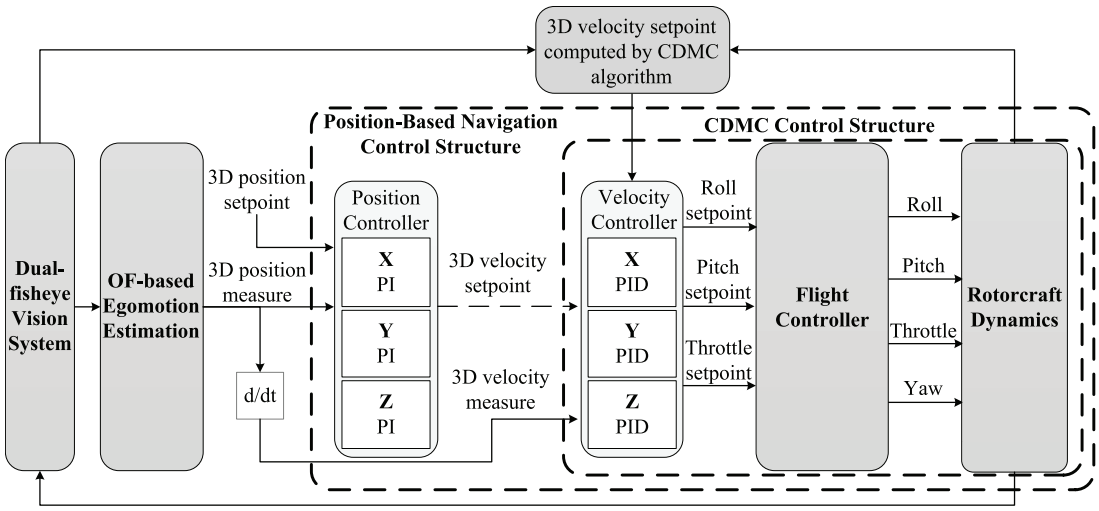


Figure 6. Control architecture for position-based and CDMC control structures.

translation and rotation) from one frame to the next. The computed frame-to-frame egomotion is fed to a path integrator to estimate the current position of the quadrotor. The full state of the aircraft at each time frame is determined through the use of path integrated optic flow and the aircraft's attitude. The optic flow algorithm implemented is a 400-point iterative block matching technique, which is computed on the panoramic images provided by the onboard vision system—for a detailed explanation of the egomotion and path integration algorithms, see [27, 28]. Once the state of the quadrotor is known, the next step is to detect the target.

4.2.2. *Target detection*

There are a multitude of target detection sensors, which can be broken into three main categories: proximity sensors (e.g. ultrasonic) motion detectors (e.g. Infrared (IR), ultrasound or radar) and imaging sensors (e.g. digital cameras using charge coupled device (CCD) or complementary metal-oxide-semiconductor (CMOS) sensors). In this work the target is detected using an image sensor. Even among techniques that use imaging sensors for target detection, there are numerous algorithms based around fiducial markers [11], dense optic flow [30], background subtraction [21] or colour information [1]. Each approach has its advantages and limitations, however, in this study there were two key requirements for detecting the target: (1) the target should be detectable regardless of whether it is stationary or moving; and (2) the target is detected in real-time. With these two requirements in mind, a fast colour-based seeded-region growing algorithm is employed to detect the target (a red ball) reliably and in real-time. The region-growing algorithm provides robustness to changes in the overall illumination of the scene, and to minor colour gradients across the target. The details of the algorithm are provided in section 3 of Strydom, *et al* [25] also provides a description of the target detection technique. In the description of the detection in [25], the detection algorithm is broken into four steps: the reference colour computation, a segmentation phase, a validation phase and a refinement step. For this current study only the segmentation and validation phases are required; the validation phase compares the new target position (in image space) and size (number or pixels) to the previous estimate to check whether it is the same target.

As the purpose of this study is to demonstrate the concept of this new guidance law for motion camouflage, a highly visible target was chosen, to facilitate accurate and reliable tracking. However, if required, this process could be replaced by a computationally efficient bio-inspired target detector [5]. For example, it has been shown that a moving target can be detected using strategies inspired by insects [13, 29].

4.2.3. *Target state estimation*

The final process required for the CDMC described here is to determine the target distance, and to compute its state (3D position and velocity). This can again be accomplished by a number of different methods. For example, proximity sensors or stereo-based methods are popular for determining the distance to a target. However, proximity sensors are often limited by range and stereovision is not always available. As the stereo overlap of the camera system is only within a small portion of the overall field of view (FOV), a more direct approach was utilised, where the target's size is known *a priori*, thus allowing its range to be computed from the size of its image in the quadrotor's vision system.

4.2.4. *Signal post-processing*

To reduce the noise in the target's state estimates and increase the stability of the quadrotor controller, two additional filtering steps were implemented: (i) application of a moving average window filter to

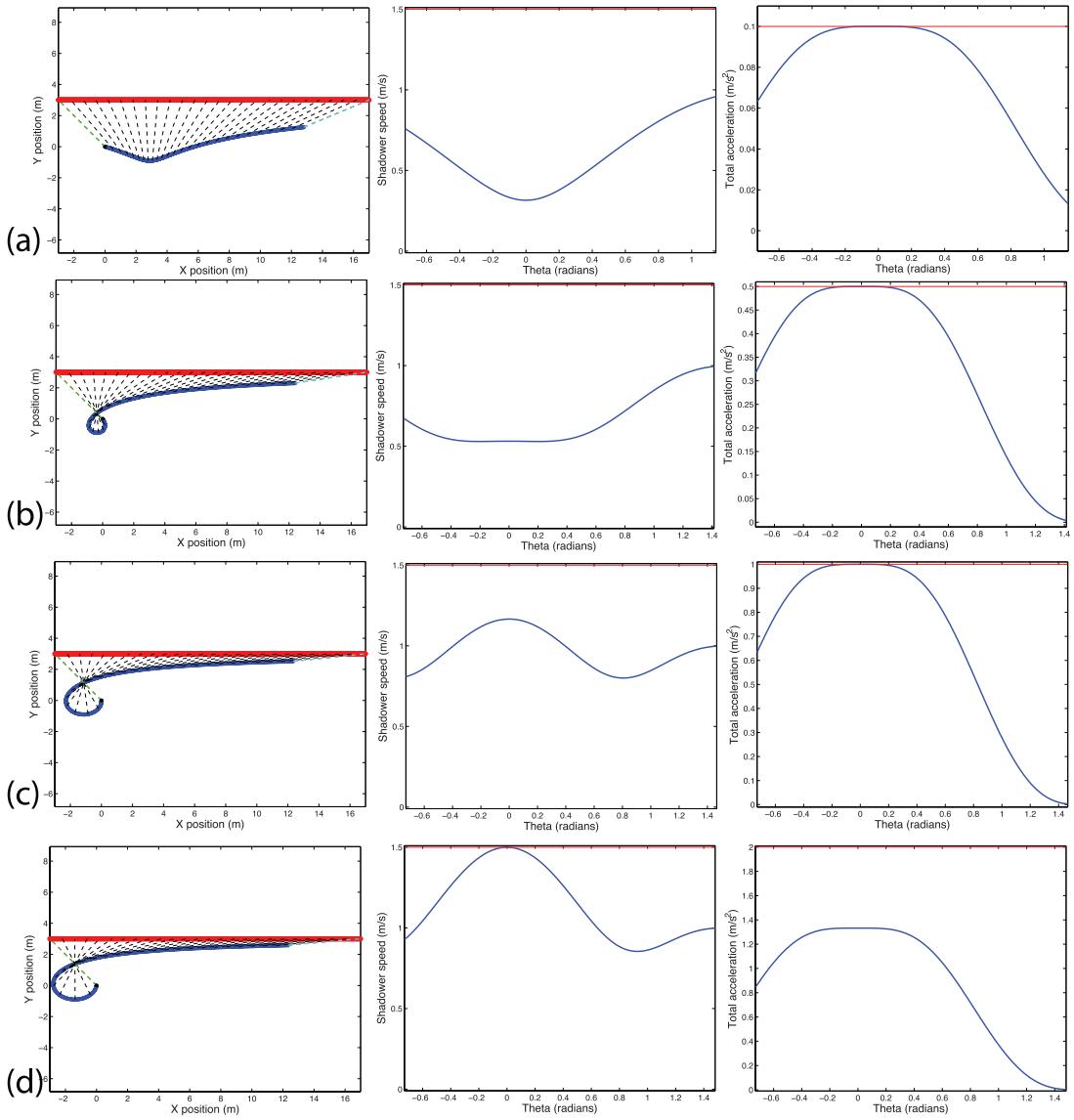


Figure 7. Different CDMC trajectories defined by maximum acceleration and velocity. Each set of three horizontal panels shows, from left to right, the 2D planar view of the 3D trajectory, the speed profile, and the total absolute acceleration profile. In the first panels, the dashed lines represent the lines connecting the instantaneous positions of the two agents, and the start and end points of the target and quadrotor trajectory are defined by the green and cyan dashed lines connecting the two paths, respectively. The horizontal red lines in the second and third panels indicate the maximum permissible speed and acceleration. Trajectory (a) is limited by a maximum acceleration of 0.1 $\mathrm{m \cdot s^{-2}}$, (b) is limited by a maximum acceleration of 0.5 $\mathrm{m \cdot s^{-2}}$, (c) is limited by a maximum acceleration of 1.0 $\mathrm{m \cdot s^{-2}}$, and (d) is limited by a maximum speed of 1.5 $\mathrm{m \cdot s^{-1}}$. The 3D representation of these trajectories is shown in figure 8.

the estimated target velocity and (ii) increasing the prediction time-step when computing the projected quadrotor state.

The first step, a moving average filter, reduces the noise in the estimated target velocity in the x, y and z directions. It was found that averaging velocities over a 25-frame window provided reliable results. Of course, if more accurate measurements are achievable, then the filter window could be reduced or even removed all together.

The second filtering step pertains to predicting the position of the target. In the theory, equations (28) and (29) are modified to:

$$\vec{Q}(t+n\Delta t) = [\vec{T}(t) + \vec{V}_t(t)n\Delta t] + \frac{\vec{F} - [\vec{T}(t) + \vec{V}_t(t)n\Delta t]}{\|\vec{F} - [\vec{T}(t) + \vec{V}_t(t)n\Delta t]\|}L \tag{30}$$

$$\vec{V}_q(t+1) = \frac{\vec{Q}(t+n\Delta t) - \vec{Q}(t)}{n\Delta t} \tag{31}$$

where these equations now predict the position and velocity required of the quadrotor in time steps into the future. The resulting quadrotor trajectory is more stable for larger n; through experimentation it was found that in the tested conditions $n = 25$ provided a stable quadrotor trajectory, even with

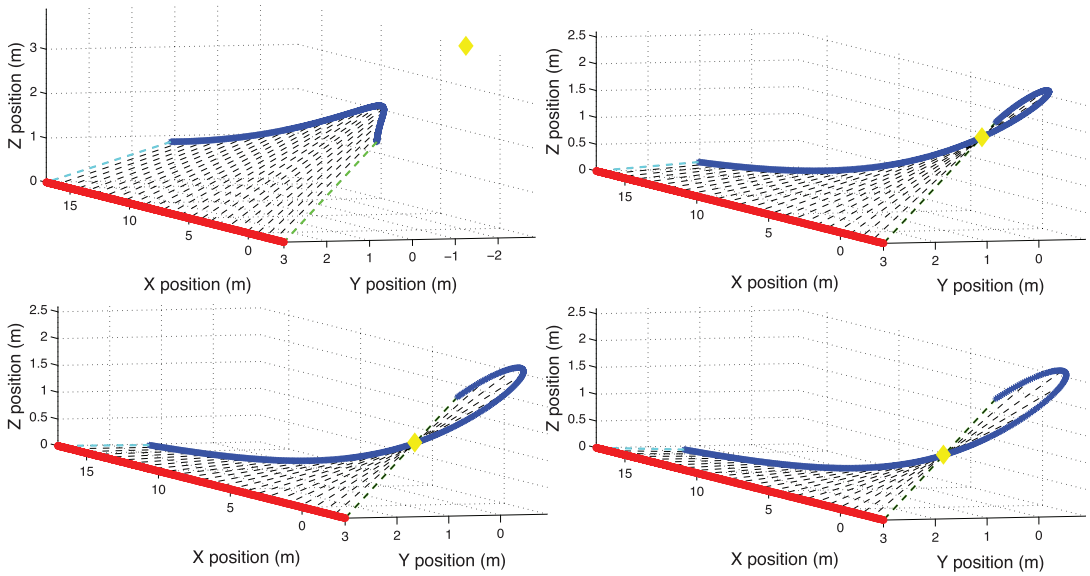


Figure 8. Different 3D CDMC trajectories defined by maximum acceleration and velocity. The quadrotor and target paths are shown by the blue and red points respectively, the fixed point is defined by the yellow diamond, the dashed lines represent the lines connecting the instantaneous positions of the two agents, and the start and end points of the trajectories are illustrated by the green and cyan dashed lines connecting the two paths respectively. Trajectory (a) is limited by a maximum acceleration of 0.1 $m \cdot s^{-2}$, (b) is limited by a maximum acceleration of 0.5 $m \cdot s^{-2}$, (c) is limited by a maximum acceleration of 1.0 $m \cdot s^{-2}$, and (d) is limited by a maximum velocity of 1.5 $m \cdot s^{-1}$. For the 2D (planar view) of the trajectory, velocity profile, and acceleration profile, see figure 7.

noisy state estimation. Then, (31) is used to compute the required quadrotor velocity for the next time-step.

The next section provides a validation of the CDMC algorithms in Matlab and demonstrates the performance of a vision-based implementation in a virtual environment.

5. Results from Matlab simulations and virtual environment tests

5.1. Validation of method

To validate the theory in sections 3 and 3.7, a number of Matlab simulations are performed. The aim of the Matlab simulations is to check that the theory holds in the 3D case for various velocity and acceleration limits. As such, these simulations are not intended to fully simulate reality with realistic noise—this will be demonstrated in section 4 using a virtual environment. The Matlab simulation provides a perfect, noise-free representation of the theory. It is used to validate the theory underlying the selection of the fixed point, and to demonstrate that the 2D analysis also holds for 3D trajectories.

Figure 7 shows four different trajectories computed using the proposed motion camouflage strategy. It illustrates a planar view of the target (red) and quadrotor (blue) trajectory; and the velocity and total absolute acceleration profiles. The position of the quadrotor as seen in figures 7 and 8 is computed using (28). The velocity and acceleration profiles are determined by equations (8) and (20) respectively. It is evident from figure 7 that, in each case, the maximum velocity and acceleration limits constrain the quadrotor's path to ensure that they are not exceeded. The corresponding 3D trajectories are illustrated in figure 8. The 3D trajectory of the shadower always lies in the plane defined by the trajectory of the prey and the fixed point, although this may occasionally appear not to be the case because the plots show a projective view of the trajectory.

In figure 7(a), the maximum acceleration was constrained to 0.1 $m \cdot s^{-2}$. For the quadrotor to maintain camouflage, but concurrently limit its acceleration to 0.1 $m \cdot s^{-2}$, the fixed point is placed behind the quadrotor's starting point. On the other hand, for figures 7(b)–(d) the acceleration limits are large enough that the fixed point can be placed in front of the quadrotor—i.e. between the target and the quadrotor. As expected, an increase in the maximum allowable acceleration requires the fixed point to be placed closer to the target—see section 3.6 for details on how the fixed point changes as a function of the target's speed and acceleration. There are two distinct advantages to placing the fixed point closer to the target: (1) the quadrotor will move behind the target earlier in time; and (2) once the initial turn is complete, the quadrotor's trajectory will be at a shallower angle to the target's motion, which may be beneficial when transitioning to an interception strategy (e.g. pursuit, constant bearing, etc [14]).

Although the Matlab simulations have validated the CDMC theory for 3D trajectories, the next step is to demonstrate that it is possible to camouflage the motion of a quadrotor UAS in a more realistic scenario.

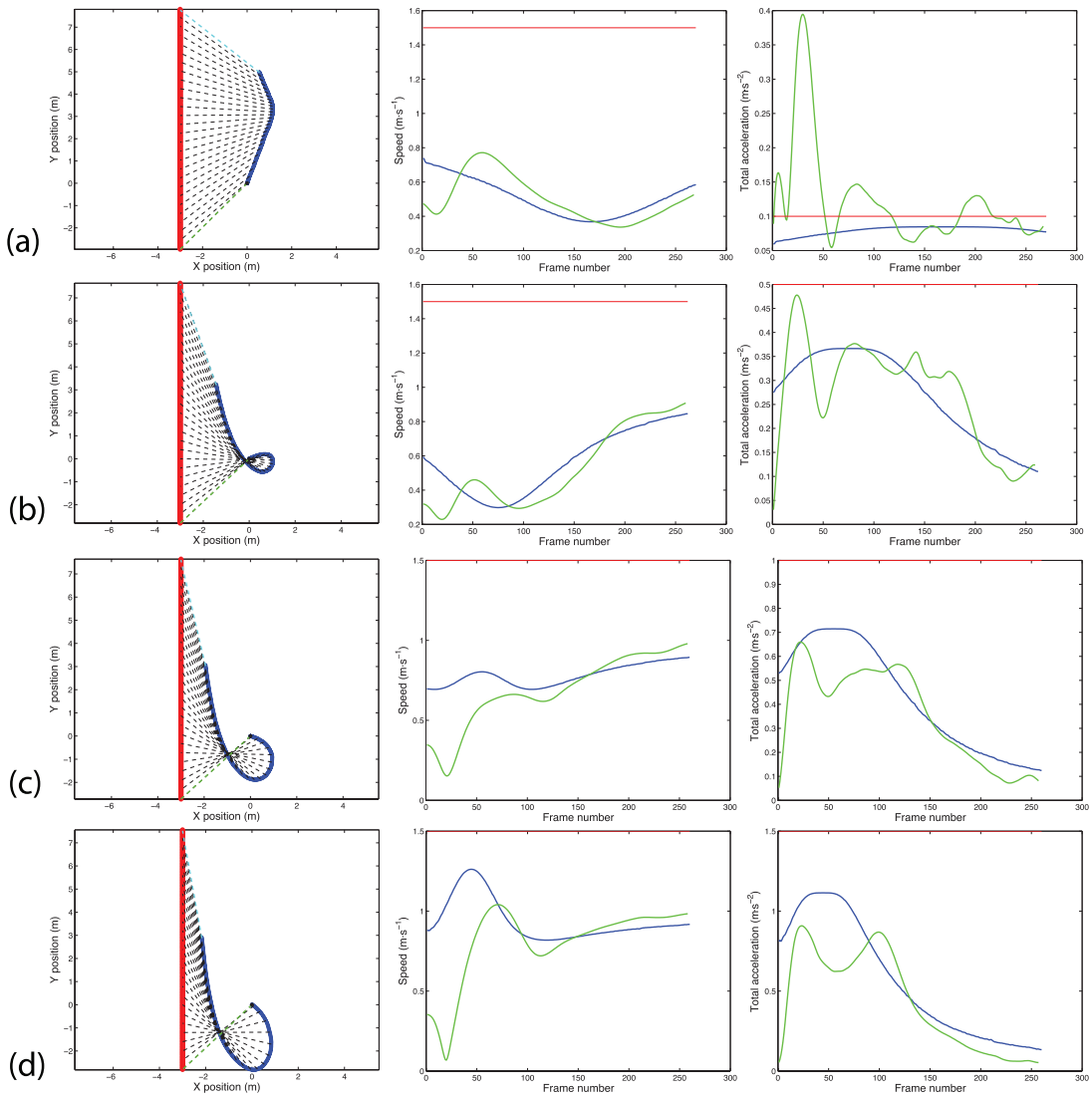


Figure 9. Different CDMC trajectories defined by maximum acceleration and velocity. Each set of three horizontal panels represents, from left to right: the 2D planar view of the 3D trajectory, as shown in figure 10 (the start and end points of the target and quadrotor trajectory are defined by the green and cyan dashed lines connecting the two paths, respectively). The speed profile, and the total absolute acceleration profile (shown by the blue (theoretical) and green (measured) curves, where the red line illustrates the maximum velocity or acceleration). Trajectory (a) is set to a maximum acceleration of 0.1 $m \cdot s^{-2}$, (b) is set to a maximum acceleration of 0.5 $m \cdot s^{-2}$, (c) is set to a maximum acceleration of 1.0 $m \cdot s^{-2}$, and (d) is set to a maximum velocity of 1.5 $m \cdot s^{-1}$.

5.2. Vision-based CDMC results

To demonstrate the performance of the CDMC algorithm onboard a UAS, the same tests were performed as in the Matlab validation (see section 5.1). Figures 9 and 10 in this section are the counterparts of figure 7 and figure 8, in section 5.1.

One way to evaluate the accuracy of motion camouflage would be to examine how precisely the quadrotor maintains the visual bearing of the fictive fixed point, from the target's point of view. This is evaluated in figure 11, which displays the angular error—the discrepancy between the desired and the actual bearing of the quadrotor, from the target's point of view. Table 1 shows the root mean square error (RMSE) of the angular error (as perceived by the target), and the distance, speed and acceleration.

Analogous to figures 7–10 show the four different trajectories using the proposed vision-based CDMC implementation in the OSG virtual environment. The four tests demonstrate the CDMC solution when the acceleration is limited to 0.1 $m \cdot s^{-2}$, 0.5 $m \cdot s^{-2}$, 1.0 $m \cdot s^{-2}$ and 1.5 $m \cdot s^{-2}$. Figure 9 shows planar views of the trajectories, and the speed and acceleration profiles. For the corresponding 3D trajectories, see figure 10.

The measured speed and acceleration profiles in figures 9(a)–(d) are well correlated with the theory; with the exception of the acceleration profile for the scenario where the acceleration was limited to 0.1 $m \cdot s^{-2}$. It is seen in figure 9(a) that the measured acceleration overshoots the desired 0.1 $m \cdot s^{-2}$ acceleration constraint. Examination of the velocity and acceleration profiles reveals a delay slight between the theor-

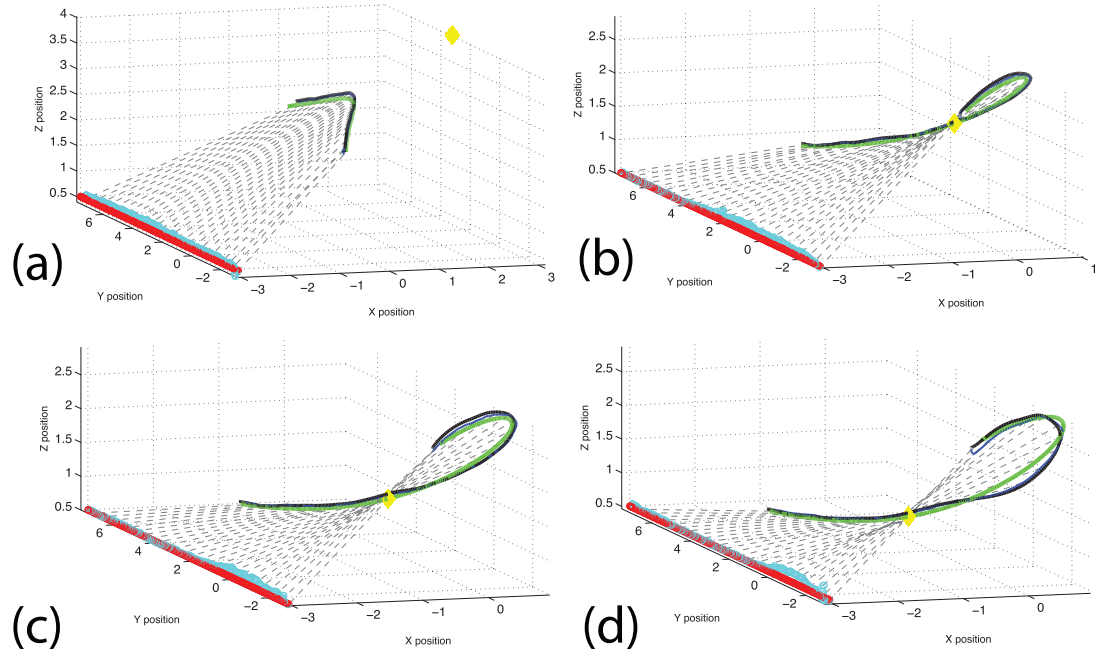


Figure 10. Different 3D CDMC trajectories defined by maximum acceleration and velocity in virtual environment. The quadrotor trajectory is displayed by the blue (vision-based estimate), green (theoretical) and the black curve (ground truth position). The target position is shown by the cyan (vision-based estimate) and red (actual) points, respectively. The yellow diamond defines the fixed point and the dashed lines represent the line connecting the instantaneous positions. Trajectory (a) is set to a maximum acceleration of 0.1 m · s^{-2}, (b) is set to a maximum acceleration of 0.5 m · s^{-2}, (c) is set to a maximum acceleration of 1.0 m · s^{-2}, and (d) is set to a maximum velocity of 1.5 m · s^{-1}. For the 2D (planar view) of the trajectory, velocity profile, and acceleration profile, see figure 9.

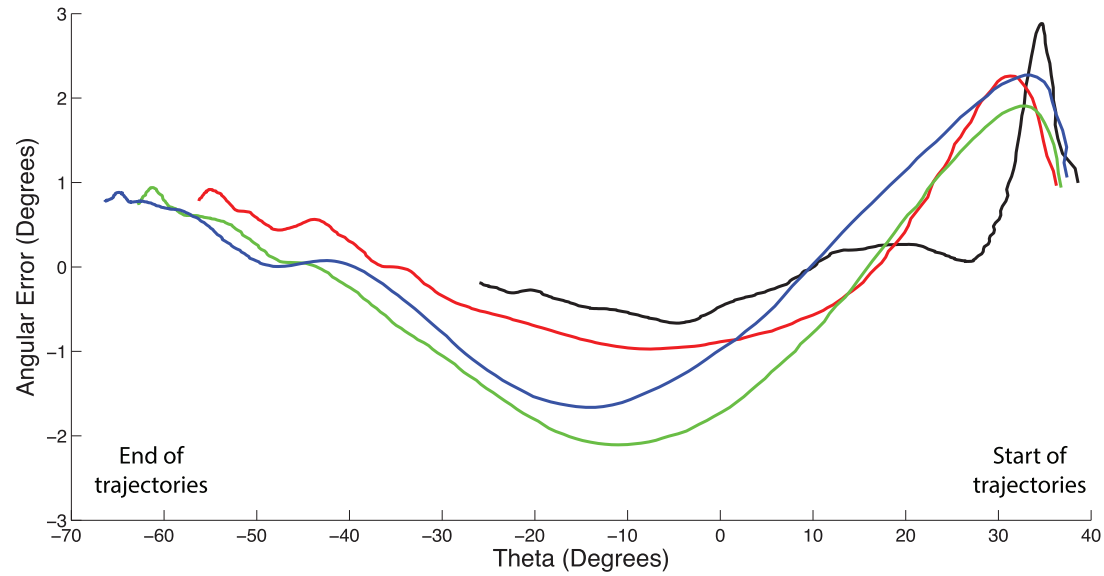


Figure 11. CDMC angular error as perceived by the target for the trajectories shown in figures 9 and 10. The four curves each represent the maximum acceleration of 0.1 m · s^{-2} (black), 0.5 m · s^{-2} (red), 1.0 m · s^{-2} (green), and the maximum velocity of 1.5 m · s^{-1} (blue). Here θ represents the angle between D and the $\vec{T}_0 - \vec{Q}_0$, as shown in figure 4.

etical and measured profiles, which will be discussed further in section 6.

Figure 10 shows the 3D paths of the quadrotor while performing the CDMC algorithm in the OSG virtual environment. It is clear that the vision-based estimate of the quadrotor trajectory is in good agreement with the ground truth, in all cases. However both of these trajectories deviate slightly from the theoretical solution. Additionally, the vision-based estimated position of the target is imperfect, which further increases the error between the measured and theoretical trajectory. Even though the desired and actual quadrotor trajectories are not perfectly aligned with the theory, it is evident that the line of sight (LOS)

Table 1. Individual CDMC test performance metrics conducted in the OSG virtual environment.

	Root mean squared errors (RMSE)			
Desired Max. Acc.	Angular error (Deg)	Dist. error (m) (%)	Speed error ($m \cdot s^{-1}$)	Acc. error ($m \cdot s^{-2}$)
0.10	0.87	0.15 [3.3]	0.12	0.09
0.20	0.96	0.17 [3.3]	0.14	0.08
0.30	0.82	0.15 [3.2]	0.14	0.06
0.40	0.95	0.15 [3.0]	0.13	0.06
0.50	0.87	0.15 [2.9]	0.11	0.07
0.60	1.09	0.15 [3.1]	0.13	0.09
0.70	1.11	0.16 [3.0]	0.15	0.10
0.80	1.08	0.15 [2.8]	0.17	0.11
0.90	1.06	0.16 [2.8]	0.20	0.12
1.00	1.05	0.16 [2.9]	0.20	0.13
1.20	1.00	0.19 [3.0]	0.28	0.19
1.40	1.00	0.21 [3.7]	0.33	0.24
1.60	0.92	0.25 [4.1]	0.34	0.25
1.80	1.00	0.29 [4.6]	0.38	0.29
2.00	0.98	0.28 [4.8]	0.41	0.34

vectors between the target and quadrotor pass very close to the desired fixed point, which is the main requirement for motion camouflage.

Table 1 further reiterates that the vision-based implementation and post-processing steps provide an accurate solution to camouflage the motion of a quadrotor at a set distance from a target. To quantify the performance of the CDMC strategy, the RMS errors in the angular direction of the quadrotor (as perceived by the target), in the distance to the target, and in the speed and acceleration were computed. The percentage distance error is computed as the mean of the percentage errors at each time step.

The results outlined in figures 9, 10 and table 1 will be analysed in greater depth in the Discussion, where the inconsistencies between the real and theoretical results are explored further.

6. Discussion

This research sets out to design a guidance law for motion camouflage—as theorised in section 3—to accomplish three main objectives: (1) the shadower moves in such a way as to be perceived as a stationary object by the shadowee, (2) the shadower pursues the shadowee at a constant distance, and (3) additional constraints are incorporated in the theoretical framework to ensure that the maximum speed and acceleration of the shadower do not exceed specified values.

To verify the CDMC theory, a number of Matlab simulations were conducted. The simulations demonstrate that the theory does indeed:

- Achieve camouflage of the quadrotor's motion.
- Ensure a constant distance to the target.
- Constrain the maximum acceleration and velocity of the quadrotor to prescribed limits.

The Matlab tests demonstrate the validity of the CDMC algorithm under perfect conditions, without considering any sensory limitations. Further testing using an OSG virtual environment demonstrates a vision-based solution onboard a quadrotor UAS to illustrate a realistic implementation of the proposed CDMC strategy. Throughout the virtual environment experiments, the quadrotor and target states were estimated utilising only a bio-inspired panoramic vision system and attitude information.

The performance of the CDMC algorithm is outlined in figures 9, 10 and table 1. To provide quantitative performance metrics, four key variables were taken into consideration: the angular error perceived by the target (i.e. the deviation of the bearing of the quadrotor from the bearing of the fixed point, as viewed by the target), the error in holding the target distance constant, and the errors in the velocity and acceleration profiles. In general, it is shown that the quadrotor trajectories generated in the OSG virtual environment closely match the theoretical trajectories obtained from the Matlab simulations—with three exceptions: (i) there is a slight overshoot in the acceleration profile for the 0.1 $m \cdot s^{-2}$ test, (ii) there is a slight delay between the theory and the measured velocity and acceleration profiles, and (iii) the 3D trajectory undershoots the desired theoretical profile.

The overshoot in the acceleration profile for the 0.1 $m \cdot s^{-2}$ test—as shown in figure 9—may be attributed partly to the PID controller, which does not produce roll, pitch and thrust outputs that correspond exactly to the desired velocities as computed by the CDMC algorithm. Furthermore, to reduce the complexity of the current theory, it is assumed that the quadrotor can instantaneously achieve the required initial speed, which may not always be the case (e.g. when the quadrotor may be initially stationary, or moving in a direction and/or speed that is substantially different from what is required at the commencement of the trajectory).

As expected, the system exhibits about a 1 second delay when comparing the measured and theoretical speeds and acceleration profiles in the OSG virtual environment implementation (see figure 9). This delay is due to the 25-frame moving average window, and the 25-frame time-step prediction used to determine the future target position, as explained in section 4.2.4. This 25-frame prediction is useful to increase the stability of the vision-based implementation by extrapolating the target position further into the future, which allows the desired state to be achieved more easily and accurately by the shadower. However, the consequence of predicting further into the future is that the rotorcraft's trajectory is slightly different from the ideal camouflage trajectory.

Even with the slight mismatch between the theory and the virtual vision-based implementation of the CDMC strategy, the quadrotor is well camouflaged,

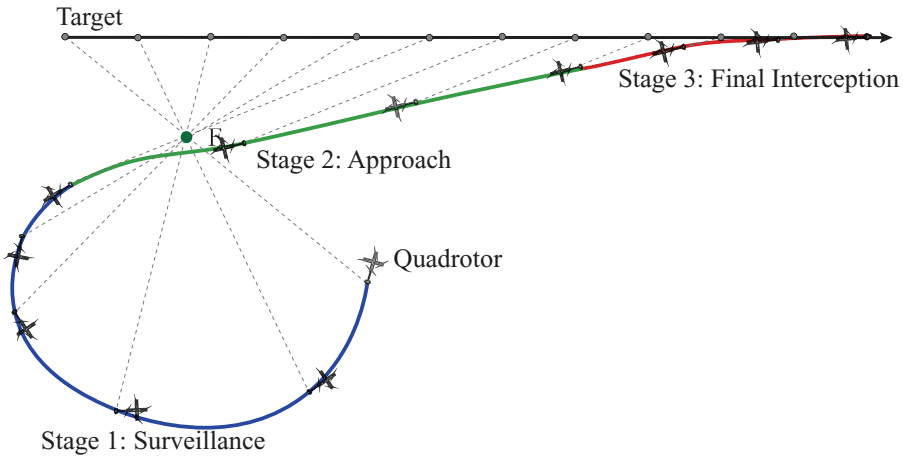


Figure 12. A three-stage approach to target pursuit.

as illustrated by the quantitative metrics in table 1. The angular of the quadrotor's bearing over 15 tests, when considered in relation to the visual acuity of the target's vision system, should provide sufficient camouflage to be undetected by most vision systems. Figure 11 shows the variation of the angular error in the quadrotor's bearing with time. As expected, the maximum error occurs at the start of the trajectory and at approximately $\theta = 0$, which is theoretically when the quadrotor's maximum acceleration occurs (in this case there is a slight delay due to the post-processing steps).

Now that it is demonstrated that the CDMC theory provides a well-defined trajectory for motion camouflage, it is important to understand the limitations and potential applications.

6.1. Limitations

The assumptions that were made to simplify the challenging problem of visual motion concealment also introduce certain limitations. In particular, the algorithm makes two assumptions in computing the extrema of the velocity and acceleration profiles, which are used to determine the best placement of the fixed point. These assumptions are that:

- The distance to the target is known.
- The target's trajectory is linear.

Although the above restrictions are required to determine the optimal position of the fixed point, it is still possible to perform CDMC by visually servoing in on the target in such a way that the image size of the target remains constant (implying a constant target range) and that the UAS trajectory conforms to the requirements of motion camouflage. The main drawback of this strategy—where the target distance is unknown or its trajectory is non-linear—is that the maximum velocity and acceleration of the UAS cannot be predetermined. Therefore, if the maximum velocity or acceleration is exceeded, the camouflage may need to be terminated unexpectedly, although pursuit of the target can continue in an uncamouflaged manner using strategies such as those described in [14], and/or at a progressively increasing distance.

Strictly speaking, if the shadower is to emulate a stationary object at the fixed point F, then the size of the shadower's image in the shadowee's retina needs to vary with time, in accordance with the distance of the shadowee from the fixed point F. This requires the size of the shadower to vary in an appropriate manner with time. Because it is difficult to vary the physical size of the shadower in practice, we investigate a strategy where the shadower pursues the shadowee at a constant, sufficiently close distance, without generating any looming cues in the retina of the shadowee. Furthermore, it is noted that if the shadowee is endowed with a range sensing capability (e.g. stereovision, 3D Time of Flight (ToF) camera, radar, etc) it cannot be 'fooled' by motion camouflage techniques. As stated before, however, many robotic platforms utilise a monocular vision system, which our technique could exploit to camouflage ones' motion.

6.2. Applications

The benefits of stealth technologies have been well established in military contexts, where covertness was accomplished predominantly through reducing acoustic, infrared, visual and emissions footprints [14]. Recently, however, the significant trend towards vision-based navigation and guidance, in both the military and civil arenas, has introduced new challenges in the quest to ensure that the motion of an aircraft is concealed with respect to other ground or air vehicles. The growing popularity of UAS and micro aerial vehicles (MAV) have paved the way for numerous civilian applications in industries such as law enforcement surveillance, animal tracking and sports footage.

Advantages of CDMC are:

- Many monocular vision-based detection algorithms for detecting approaching objects look for expansion (e.g. [26]), which the CDMC method will not generate.
- For surveillance tasks that require the use of high optical zoom for detailed inspection, the target distance will remain constant, and thus the optical zoom setting can remain unchanged while the aircraft is achieving motion camouflage.

Additionally, in the context of pursuit and interception, the CDMC algorithm provides a covert method to move behind the target when the start position is initially in front of the target—where the position (in front or behind) is relative to the target's direction of travel. Figure 12 illustrates a potential three-stage pursuit strategy initially performing CDMC then Constant Bearing interception and finally simple pursuit. In general, pursuit and interception methods are more likely to be successful when the target is pursued from behind rather than approached from the front, especially when the target has the capability to actively evade (or attack) the prospective pursuer. Furthermore, our CDMC strategy can be modified to be an interception or avoidance strategy by progressively decreasing or increasing the distance to the target.

7. Conclusions

In this paper a novel constant distance motion camouflage (CDMC) guidance law is derived, which additionally constrains the trajectory to the velocity and acceleration limits of the shadower. The theory is validated in Matlab simulations, and a vision-based implementation is tested in a virtual environment—demonstrating that, even with noisy state information, it is possible to remain well camouflaged using the presented CDMC technique. Finally, this is the first time that a motion camouflage strategy has been implemented and tested on a rotorcraft unmanned aerial system(s) in a realistic virtual environment.

Acknowledgments

We thank the anonymous referees for their valuable comments and suggestions for improving the manuscript.

The research described here was supported partly by the ARC Centre of Excellence in Vision Science (CE0561903), by Boeing Defence Australia Grant SMPBRT-11-044, ARC Linkage Grant LP130100483, ARC Discovery Grant DP140100896, a Queensland Premier's Fellowship, and an ARC Distinguished Outstanding Researcher Award (DP140100914). We thank Aymeric Denuelle for his assistance with the colour detection algorithm.

References

[1] Adams R and Bischof L 1994 Seeded region growing *IEEE Trans. Pattern Anal. Mach. Intell.* **16** 641–7
[2] Altug E, Ostrowski J P and Taylor C J 2003 Quadrotor control using dual camera visual feedback *Proc. IEEE Int. Conf. on Robotics and Automation* vol 3 (IEEE) pp 4294–9
[3] Anderson A J and McOwan P W 2002 Towards an autonomous motion camouflage control system *Proc. of the Int. Joint Conf. on Neural Networks* vol 3 pp 2006–11
[4] Anderson A J and McOwan P W 2003 Humans deceived by predatory stealth strategy camouflaging motion *Proc. R. Soc.* B **270** S18–20
[5] Bagheri Z M, Wiederman S D, Cazzolato B S, Grainger S and O'Carroll D C 2014 A biologically inspired facilitation mechanism enhances the detection and pursuit of targets of varying contrast *Int. Conf. IEEE Digital Image Computing: Techniques and Applications* pp 1–5
[6] Carey N and Srinivasan M 2008 Energy-efficient motion camouflage in three dimensions (arXiv:0806.1785)
[7] Carey N E, Ford J J and Chahl J S 2004 Biologically inspired guidance for motion camouflage *Control Conf. 5th Asian* vol 3 (IEEE) pp 1793–9
[8] Collett T S and Land M F 1978 How hoverflies compute interception courses *J. Comparative Physiol.* **125** 191–204
[9] Denuelle A and Srinivasan M V 2016 A sparse snapshot-based navigation strategy for UAS guidance in natural environments *IEEE Int. Conf. on Robotics and Automation* pp 3455–62
[10] Dieter Fox W B and Thrun S 1997 The dynamic window approach to collision avoidance *IEEE Trans. Robot. Autom.* **4** 1
[11] Fiala M 2005 Artag, a fiducial marker system using digital techniques *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* vol 2 (IEEE) pp 590–6
[12] Glendinning P 2004 The mathematics of motion camouflage *Proc. R. Soc.* B **271** 477–81
[13] Gonzalez-Bellido P T, Fabian S T and Nordström K 2016 Target detection in insects: optical, neural and behavioral optimizations *Curr. Opin. Neurobiol.* **41** 122–8
[14] Howe D 1991 Introduction to the basic technology of stealth aircraft: part 1—basic considerations and aircraft self-emitted signals (passive considerations) *J. Eng. Gas Turbines Power* **113** 75–9
[15] Justh E W and Krishnaprasad P 2006 Steering laws for motion camouflage *Proc. R. Soc.* A **462** 3629–43
[16] Milford M J, Wyeth G F and Prasser D 2004 RatSLAM: a hippocampal model for simultaneous localization and mapping *IEEE Int. Conf. on Robotics and Automation Proc.* vol 1 (IEEE) pp 403–8
[17] Mizutani A, Chahl J S and Srinivasan M V 2003 Insect behaviour: Motion camouflage in dragonflies *Nature* **423** 604–4
[18] Olberg R M 2012 Visual control of prey-capture flight in dragonflies *Curr. Opin. Neurobiol.* **22** 267–71
[19] Ranó I and Iglesias R 2016 Application of systems identification to the implementation of motion camouflage in mobile robots *Auton. Robots* **40** 229–44
[20] Reddy P, Justh E and Krishnaprasad P 2007 Motion camouflage with sensorimotor delay *46th IEEE Conf. on Decision and Control* (IEEE) pp 1660–5
[21] Sheikh Y, Javed O and Kanade T 2009 Background subtraction for freely moving cameras *IEEE 12th Int. Conf. on Computer Vision* (IEEE) pp 1219–25
[22] Srinivasan M V and Davey M 1995 Strategies for active camouflage of motion *Proc. R. Soc. Lond.* B **259** 19–25
[23] Strydom R, Denuelle A and Srinivasan M V 2016 Bio-inspired principles applied to the guidance, navigation and control of UAS *Aerospace* **3** 21
[24] Strydom R, Singh S P and Srinivasan M V 2015 Biologically inspired interception: a comparison of pursuit and constant bearing strategies in the presence of sensorimotor delay *IEEE Int. Conf. on Robotics and Biomimetics* (IEEE) pp 2442–8

[25] Strydom R, Thurrowgood S, Denuelle A and Srinivasan M V 2015 *UAV Guidance: a Stereo-Based Technique for Interception of Stationary or Moving Targets* (Cham: Springer) pp 258–69
[26] Strydom R, Thurrowgood S, Denuelle A and Srinivasan M V 2016 TCM: a vision-based algorithm for distinguishing between stationary and moving objects irrespective of depth contrast from a UAS *Int. J. Adv. Robot. Syst.*
[27] Strydom R, Thurrowgood S and Srinivasan M V 2014 Visual odometry: autonomous UAV navigation using optic flow and stereo *Proc. Australasian Conf. on Robotics and Automation (Melbourne, Australia)*
[28] Thurrowgood S, Moore R J, Soccol D, Knight M and Srinivasan M V 2014 A biologically inspired, vision-based guidance system for automatic landing of a fixed-wing aircraft *J. Field Robot.* **31** 699–727
[29] Wiederman S D and O'Carroll D C 2011 Discrimination of features in natural scenes by a dragonfly neuron *J. Neurosci.* **31** 7141–4
[30] Yalcin H, Hebert M, Collins R and Black M J 2005 A flow-based approach to vehicle detection and background mosaicking in airborne video *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* vol 2 (IEEE) p 1202