Snapshot-based Navigation for the Guidance of UAS

Aymeric Denuelle[†‡] and Mandyam V. Srinivasan[†]
[†]The University of Queensland, St Lucia, Australia
[‡]Autonomous Systems Program, CSIRO, Pullenvale, Australia
a.denuelle@uq.edu.au, m.srinivasan@uq.edu.au

Abstract

This paper presents a bio-inspired homing strategy applied to the guidance of rotorcraft unmanned aerial systems (UAS). In this approach, a sequence of panoramic snapshots is stored to build a visual route between the home and the goal locations. Navigation back to the initial location consists in reaching each of the memorised snapshot positions, by performing successive local homing steps. For that purpose, a snapshot-based method is used to estimate the rotorcraft's 3D position and velocity in real-time. This algorithm provides a simple, yet efficient means for describing a visual route, and enabling 3D navigation of a small-size quadrotor in unknown outdoor environments. Results from closed-loop flight experiments are presented to evaluate the performance of our approach.

1 Introduction

In the past decades, field service robotics has extended its range of application from mobile terrestrial robots to autonomous underwater vehicles and unmanned aerial systems (UAS). With a growing interest in civilian applications, UAS have been deployed in applications ranging from military and surveillance operations to environmental surveying, precision agriculture, search and rescue operations, structure inspection, etc. In the field of autonomous aerial navigation, a large number of applications require the ability of flying at low altitudes, near structures and more generally in cluttered environments. For these reasons, rotorcraft have gained in popularity as they enable vertical take-off and landing, hovering, and also perform maneuvers with greater accuracy than fixed-wing aircraft.

Although UAS autopilots continue to rely primarily on satellite navigation systems, vision has drawn the interest of the research community to further increase the autonomy of UAS but also their capabilities. Indeed, in addition to providing means of estimating egomotion and enabling navigation without relying on external signals, vision sensors are multi-purpose systems that can be simultaneously used to perform other specific tasks. Imaging devices augment navigational systems with the capability of collision avoidance, detection and tracking, without requesting additional sensors (laser rangefinders for instance) which are often bulky or power demanding. Furthermore, the same system can be used for additional vision-based tasks such as surveillance, inspection, situational awareness and scene understanding. Consequently, vision-based navigation enables the downsizing of the onboard avionics, thus providing an increased payload capacity, but it also offers, in small-scale operations, greater accuracy compared to GPS-based autopilots.

However, one remaining drawback of vision-based algorithms is their high computational burden or their reliance on high resolution imagery. To address this issue, insights into biological behaviours have contributed to the development of simpler yet efficient navigation techniques. Indeed, foraging insects, such as bees, wasps and ants, demonstrate remarkable navigation strategies, and that is despite their limited brain size and low-resolution eyes. For instance, bees and wasps make use of visual snapshots to memorise locations and pinpoint the hive entrance [Cartwright and Collett, 1983].

Visual homing is the behaviour in which an animal, or a robotic agent, retrieves a pre-visited location by relating the currently experienced visual information with that stored in memory. In this paper, we address the problem of 'long-range homing', i.e. retrieving a home position which is located outside the navigator's currently observable visual scene – as opposed to 'local homing' which involves moving towards a goal location that is already visible in the current scene. Inspired by biological observations, this work is motivated by the problem of visually navigating in natural environments, without prior knowledge of the surroundings, using a simple but

efficient method suitable for the real-time guidance of small-size, computationally constrained UAS. Such capability is needed in GPS-denied environments, as it would benefit low-altitude search and rescue operations as well as long-term UAS operations which require refueling, or cargo refill (in the case of precision spraying in agriculture for instance). Another application is that of providing a backup solution for GPS-based autopilots, taking over control during brief signal outages, and hence enabling safe and robust navigation.

The main contributions of this paper are the demonstration of the feasibility of a snapshot-based route description that enables long-range visual homing in 3D unstructured environments, and the development of a bio-inspired strategy that estimates the 3D position and velocity of a rotorcraft UAS through snapshot matching and enables its navigation in natural scenes. Concurrently, this paper also enables testing and validating view-based theories of insect navigation along three-dimensional routes [Collett and Collett, 2002] by applying these strategies to UAS guidance in outdoor environments.

The remainder of this paper introduces the related work in Section 2, the proposed bio-inspired homing strategy for long-range navigation is described in Section 3, and the experimental setup and rotorcraft platform are presented in Section 4. The performance of our approach is evaluated in Section 5, based on several outdoor closed-loop flight tests, and discussed in Section 6. Finally, Section 7 draws the conclusions of this paper.

2 Related Work

In the field of autonomous navigation, one can distinguish between map-based approaches, such as simultaneous localisation and mapping methods (SLAM), and mapless strategies that can be either based on following a route, reliant on visual odometry, or both.

Visual SLAM techniques [Weiss *et al.*, 2013; Engel *et al.*, 2012] rely on feature extraction to estimate the pose of a UAS. Although achieving really good accuracy in outdoor environments (better than 20cm in average for instance in [Engel *et al.*, 2012]), traditional SLAM methods perform the costly extraction and matching of hundreds of features, which generally requires sufficient image resolution and computational power to be executed in real-time onboard small UAS.

Although robust descriptors have been developed to detect natural landmarks and applied to UAS guidance [Cesetti *et al.*, 2010], the extraction of these features remains a challenging task in unstructured natural environments that do not present prominent landmarks. Moreover, while insects are remarkable navigators, [Dyer, 1996] provides evidence that they do not seem to rely on maps for navigation and rather store multiple routes, without retaining their topological relationships.

Optic flow (OF) has been widely used in visual odometry for the stabilisation and guidance of unmanned rotorcraft because its computation does not require the identification or tracking of explicit landmarks within a visual scene. Indeed, OF-based navigation can rely on intensity gradient [Romero *et al.*, 2009; Kendoul *et al.*, 2009], block matching techniques [Strydom *et al.*, 2014], or simpler, non-iterative algorithms [Zufferey and Floreano, 2006; Li *et al.*, 2013]. Due to the integration of cumulative odometric errors, such methods usually experience a long-term drift in their pose estimates. However, this effect can be reduced to some extent by fusing the visual information with inertial measurements [Kendoul *et al.*, 2009; Ahrens *et al.*, 2009]. Another way to prevent this long-term drift is to perform image matching with a reference snapshot, and to compute optic flow with regard to this snapshot, rather than integrating odometric estimates [Li *et al.*, 2015].

Instead of relying on dead reckoning, navigating back to a familiar location can rely on learning a visual route and then make use of this memorised route to perform long-range homing. A visual route can be represented holistically, without waypoints, by training a classifier to distinguish between positive forward-facing views belonging to the travelled route and negative ones (left and right-facing views) [Baddeley *et al.*, 2011]. Route following is then achieved by scanning the environment and moving in the direction which seems to belong to the route, as indicated by the classifier.

Traditionally, a visual route is represented as an ordered series of discrete waypoints that define visual locales. In that case, navigation involves successively reaching each of these waypoints until retrieving the initial location. Consequently, route-based navigation is often performed by augmenting local homing schemes (see [Denuelle and Srinivasan, 2015] for a review). The route waypoints can be represented by global descriptors (such as the entire image [Matsumoto *et al.*, 1996; Jones *et al.*, 1997; Vardy, 2006; Smith *et al.*, 2007; Labrosse, 2007], or multi-dimensional image histograms based on colour, edge density, intensity gradient and texture [Zhou *et al.*, 2003]), but also by extracted local descriptors such as feature points (image corners [Argyros *et al.*, 2005], Harris corners [Courbon *et al.*, 2009], SIFT points [Fu and Hsiang, 2012]), average landmark vectors [Smith *et al.*, 2006], intensity patterns and their image coordinates [Chen and Birchfield, 2006], or geometric features (lines) [Ohno *et al.*, 1996; Sagüés and Guerrero, 2005].

Even if they enable simple, qualitative, navigation schemes [Chen and Birchfield, 2006], a sufficient number of these local descriptors needs to be successfully

extracted and tracked throughout the visual locales [Argyros *et al.*, 2005; Chen and Birchfield, 2006], advocating the use of panoramic images [Argyros *et al.*, 2005]. Image-based approaches, using global descriptors, also use omnidirectional vision to make view-based navigation less sensitive to occlusions by dynamic objects and enable the following of the learned route from a different direction. Panoramic vision, which is employed by insects, also has the advantage of simplifying the computation of egomotion by making translation easily distinguishable from rotation [Nelson and Aloimonos, 1988].

In this paper, we present a bio-inspired, mapless approach to the visual navigation of a rotorcraft UAS. Compared to the existing literature, our approach provides several advantages: it does not require any absolute pose estimation or feature extraction, and it is therefore suitable for navigation in unstructured natural environments. Although based on optic flow, the UAS pose is estimated without requiring path integration, and it does not accumulate odometric errors. Additionally, a simple route description is used where the memorised route is defined only by a sequence of panoramic snapshots. Thus, we propose here an efficient, route-based, navigation strategy that is applicable to real-time visual guidance and control of UAS.

3 Route Learning and Long-range Homing

This section presents our navigation strategy that comprises two modules: route learning and long-range homing. Both modules make use of a snapshot-based method which computes optic flow to estimate the rotorcraft's egomotion relative to a memorised panoramic snapshot. This egomotion information is then used to either select whether the current camera frame should be memorised as a new snapshot waypoint (route learning) or to estimate the 3D state (position and velocity) of the UAS relative to the position corresponding to a snapshot acquired previously along the navigated route.

3.1 Snapshot-based Estimation of Egomotion

A pyramidal, 7x7 pixel block-matching algorithm is employed to compute sparse optic flow with subpixel precision, similar to [Thurrowgood *et al.*, 2014; Strydom *et al.*, 2014], with the difference that optic flow is always computed between the current camera frame and a particular snapshot, and not between successive frames. Consequently, this method has the advantage of providing drift-free egomotion estimates when compared to the traditional frame-to-frame integration of optic flow measurements that accumulates odometric errors [Denuelle *et al.*, 2015a]. Optic flow vectors are computed on 360x180 pixel panoramic images, as delivered by the onboard vision system (see Section 4.1). An optimised iterative process [Powell, 2009] is applied to these OF measurements (400 points) to determine the best egomotion (combination of 3D translation and rotation) that minimises the reprojection error, assuming ground planarity, as described in [Thurrowgood *et al.*, 2014]. Attitude information (roll and pitch angles) from the onboard attitude and heading reference system (AHRS) is used to determine the UAS orientation (normal vector) relative to the ground plane.

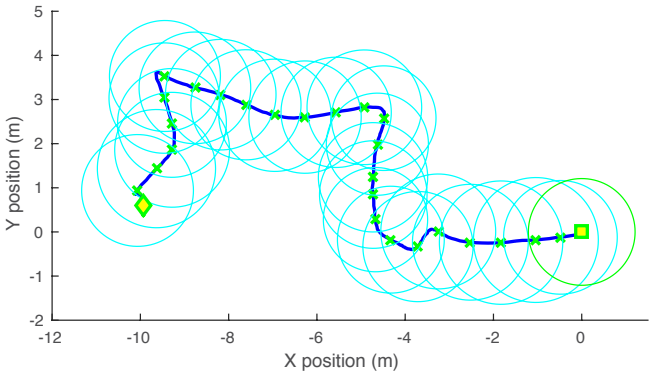

Figure 1: Route learning during a closed-loop flight test: a succession of snapshots are taken along the journey from a home position (square) to a goal location (diamond), as indicated by ground truth data. The snapshot waypoint locations are marked by green crosses and the overlapping catchment areas are shown in cyan.

Because the obtained 3D translation and rotation are expressed in terms of height units, these egomotion estimates are then augmented with height measurements, provided by the vision system (see Section 4.1), and the resultant 3D position signal is fed into the control structure (described in Section 4.2) that requires metric inputs. The UAS' 3D velocity is obtained by differentiating and low-pass filtering the position estimates, and is also used in the control loop.

3.2 Visual Route Description

In this section, we describe the algorithm used to select the key frames that need to be stored to build a minimal visual route between two locations. We aim at building a minimalistic route that is composed only of a series of snapshots, without memorising any other information to describe the waypoint locations along a travelled journey. Additionally, this route description must enable the UAS to navigate back to its initial location without relying on path integration at any moment.

A panoramic snapshot describes a particular location, but also defines, for a given local homing method, a 3D region, or catchment area [Cartwright and Collett, 1987], within which an agent can successfully return to

that place of interest. As demonstrated in [Denuelle *et al.*, 2015b], the catchment area associated with our snapshot-based OF method can be represented by a sphere of radius R_{CA} equal to about 0.3 height units. We assume, in the route description presented here, that the UAS navigates in a non-cluttered environment where this 0.3 height unit catchment area assumption remains valid for each camera frame.

If an agent has the ability to reach each snapshot with perfect accuracy, then the entire route can be defined by a series of snapshots taken at regular distances equal to the radius of the catchment area (CA), which corresponds to a minimal overlap of 50% between consecutive catchment areas. However, as this route description is applied to UAS navigation, we consider that reaching a particular snapshot is defined by getting back to its vicinity (SV), represented by a sphere of radius r_{SV} at the snapshot location. Consequently, the minimal overlap between successive CAs is directly linked to the accuracy with which the UAS is tasked, or able, to reach a snapshot waypoint. We denote the snapshot vicinity by SV, its radius by r_{SV}, and the radius of the catchment area by R_{CA}. A minimalistic route description is then obtained by storing a new snapshot at every location separated from the previous memorised snapshot by a 3D distance $d = R_{CA} - r_{SV}$.

Route learning is initiated by taking a snapshot at the home location and terminated when a location of interest is reached. A new snapshot is added to the visual route every time that the estimated 3D travelled distance between the current frame and the last memorised snapshot becomes greater than the distance d. The travelled distance is estimated by our snapshot-based OF method, without relying on path integration. An example of the route learning process that occurred during a closed-loop flight test is displayed in Figure 1.

3.3 Bio-inspired Long-range Homing

For both the route learning and long-range homing stages of our navigation strategy, the UAS' egomotion needs to be estimated relative to a particular, memorised snapshot. That snapshot is either the last stored key frame during route learning, or the current snapshot waypoint that is used as intermediary homing setpoint during route following.

It is assumed here that a visual route, defined by a succession of snapshots and representing a journey between a home and goal locations, has been stored in memory, as previously described in Section 3.2. Long-range homing back to the home location is performed by successively reaching, in reverse order, each of the intermediary waypoint locations belonging to the learnt route.

The snapshot-based OF method described in Section 3.1 is now used as a local homing method and enables

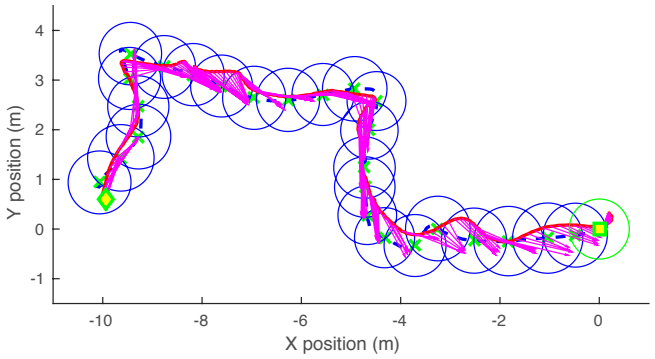

Figure 2: Long-range homing during a closed-loop flight test: the followed path back to the home location is shown in red and consists in reaching the vicinity (indicated by blue circles) of each route waypoint memorised during the outward journey (dashed blue line) to the goal location. Home and goal positions are marked by a square and a diamond, respectively. Home vectors to each route waypoint are shown in magenta.

estimation of the UAS' state relative to the current route waypoint that needs to be reached. A snapshot waypoint is considered to be reached when the UAS is within the vicinity of the snapshot location (SV), represented by a sphere of radius r_{SV}, according to the route description presented in Section 3.2. Then, the preceding snapshot waypoint is set as the new intermediary goal and the UAS' state is estimated relative to this newly selected waypoint. This process is repeated until the home location is reached, as pictured in Figure 2.

During long-range homing, we need to know when a snapshot vicinity is reached, so that the next route waypoint can be set as the new intermediary goal. This information depends on the ability to scale the OF measurements relative to a memorised snapshot by the height at the snapshot location. Although the height h_S of the snapshot waypoints is not stored during route learning, this can be deduced from the current height estimate h by $h_S = \frac{h}{(1+\delta y)}$ where δy is the estimated translation along the vertical axis between the current frame and a particular snapshot, using the snapshot-based OF method described in Section 3.1.

4 Experimental Platform and Setup

This section provides an overview of the rotorcraft, the onboard sensor suite and the control structure used during the closed-loop flight tests. It also presents the flight scenario and the parameters that were used for the navigation strategy.

4.1 Rotorcraft Platform

Our experimental platform is a small-size, custom-built, quadrotor incorporating a dual camera system located at

(a) (b)

Figure 3: The rotorcraft platform used during the closed-loop experiments (a), and a close-up of the biologically-inspired vision system (b).

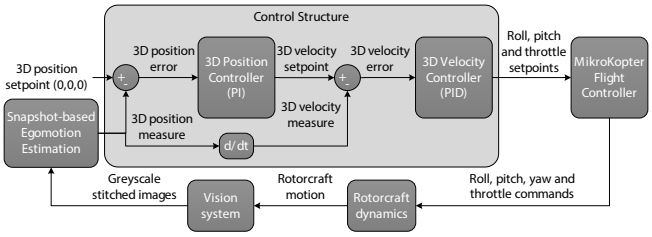

Figure 4: Onboard control structure used to control the rotorcraft's velocity to attain the 3D position setpoint corresponding to the location of the next snapshot.

the front, as pictured in Figure 3(a). This biologically-inspired visual system, described in [Thurrowgood *et al.*, 2014], combines two miniature cameras, mounted back-to-back and equipped with fish-eye lenses, as seen in 3(b), so that it delivers an almost spherical field of view. Images from both cameras are software synchronised at a 25Hz frame rate, converted to grayscale, and stitched together to form the 360x180 pixel panoramic images used by our snapshot-based pose estimator. Additionally, the two cameras are slightly tilted towards each other to enable a frontal stereo overlap. Hence, a stereo-based estimate of height is directly available by using the central strip of the panoramic images. A MicroStrain 3DM-GX3-25 AHRS and a Piksi GPS receiver[1] (as an element of our ground truthing system, see Section 4.3) are also a part of the onboard sensor suite.

Real-time image processing, guidance and control of the rotorcraft are performed by the onboard Intel NUC computer (2.6GHz dual-core processor) while low-level attitude stabilisation is handled by a MikroKopter flight controller[2].

4.2 Onboard Control Structure

The onboard control loop runs at the cameras' frame rate (25Hz). It is composed of 3 sets of two cascaded proportional-integral-derivative (PI-PID) controllers, as shown in Figure 4, that regulate the quadrotor's 3D position and velocity. Our snapshot-based OF method, described in Section 3.1, provides the position controller with an estimated 3D position measure relative to the current snapshot waypoint. Independently of the current waypoint, the 3D setpoint is always set to be $(0, 0, 0)$. The position controller compares the 3D measure and setpoint, and outputs a 3D velocity setpoint that is used as an input into the velocity controller, along with an estimated 3D velocity measure (obtained through our snapshot-based OF method). The velocity controller then delivers appropriate roll, pitch and throttle commands to the MikroKopter flight controller. This MikroKopter inner-loop for attitude stabilisation then updates the roll, pitch and throttle commands. Additionally, it also uses AHRS heading measurements to output a yaw command.

4.3 Ground Truthing System

In Section 5, we evaluate the performance of our snapshot-based navigation strategy, using ground truth positioning provided by two Piksi GPS receivers, with real-time kinematics functionality, that deliver differential GPS measurements at a frequency of 5Hz. Although logged in real-time onboard the quadrotor, this ground truth data is not used during the closed-loop flight tests, only the visual estimates (see Section 3.3) and attitude information (Euler angles) from the AHRS are employed to control the rotorcraft.

4.4 Flight Parameters and Experiment Description

In the flight experiments presented in Section 5, we chose a conservative value for the size of the catchment area R_{CA}, equal to 0.25 height units, and successive snapshots were taken with a 75% overlap. Consequently, during route following, the snapshot vicinity can be defined as a sphere of half the radius of the catchment area, i.e. $r_{SV} = 0.5 \times R_{CA}$.

Our flight tests consist of two stages: an exploratory ('learning') journey between a home and a goal location, and autonomous navigation back home using the vision-based strategy described in Section 3. First, the rotorcraft follows a pre-set square wave trajectory of length 20m to a goal location, by navigating using frame-to-frame optic flow measurements, similarly to [Strydom *et al.*, 2014] (the same method is also used to enable autonomous take-off and landing during the tests). During the exploratory journey, only the route learning module is in operation. Subsequently, the route following module is used to navigate back to the home location (see results in Section 5.1). The route learning stage is running concurrently to memorise the inward journey and enable route following back to the goal if desired. Route

[1] `http://docs.swiftnav.com/pdfs/piksi_datasheet_v2.3.1.pdf`

[2] `http://wiki.mikrokopter.de/en/FlightCtrl_ME_2_1`

learning is achieved during several outward and inward journeys in the results presented in Section 5.2.

5 Closed-loop Flight Results

In this section, we present the results of closed-loop flight tests where our navigation strategy (described in Section 3) is used to perform long-range homing (back to the home location, after the initial exploratory journey), and repetitive navigation between the home and goal locations. These two scenarios are respectively presented in Sections 5.1 and 5.2, and were tested in low-wind conditions, with wind gusts ranging from 3 to 6 knots (measured at ground level). All tests were conducted at a height of about 5m above ground.

Results are presented in terms of the length of the homing trajectories, the size of the visual route (number of memorised snapshots), and the homing accuracy. In the following results, X, Y and Z axes designate the geodesic North, East and Down directions, respectively, as used by our differential GPS system.

5.1 Long-range Homing

An example of route learning and long-range homing (route following) recorded during the closed-loop flight tests is shown in Figures 1 and 2.

Although 20 flight experiments were conducted, and all of them were successful homings, a smaller set (10) of flight tests, where the DGPS data does not present any temporary signal loss (resulting in positional inaccuracy), is used in Table 1 to compute the homing statistics. Outward and inward paths (respectively from and to the home location) were measured in length by integrating the DGPS position data. On average, we observe a path reduction of about 4.5% between the exploratory journey and the trajectory followed during long-range homing. This can be explained by the fact that, in our navigation strategy, route following is defined as only reaching each snapshot waypoint's vicinity rather than the exact snapshot locations. During route learning, the mean travelled distance between two acquired snapshots is 0.73m, which is slightly greater than the theoretical inter-snapshot distance $d = R_{CA} - r_{SV} = 0.625$m (for a flight conducted at an average height of 5m with R_{CA} and r_{SV} chosen as described in Section 4.4). This makes sense as two successive snapshots are taken at locations that are spatially separated by a distance d, and this is done independently of the actual distance travelled by the quadrotor (for example, it prevents multiple snapshots to be taken if the rotorcraft temporarily hovers at a location, despite the integrated ground truth travelled distance increasing).

In our experiments, home is reached when the rotorcraft enters its catchment area (of radius $R_{CA} = 1.25$m) and the UAS is tasked to hover for 5s. The mean home position error, as indicated by DGPS data (see Table 1), is 0.50 ± 0.25m, which demonstrates that the rotorcraft successfully reached the home catchment area and even the home vicinity (sphere of radius 0.625m).

Flight test	Outward path length (m)	Inward path length (m)	Path reduction (%)	Route size (snapshots)	Mean distance inter-snapshots (m)	Home position error (mean ± std dev., m)
1	19.18	17.30	9.78	27	0.71	0.52 ± 0.24
2	17.47	16.07	8.00	24	0.73	0.66 ± 0.30
3	17.54	16.88	3.80	25	0.70	0.45 ± 0.30
4	18.17	17.63	2.98	25	0.73	0.58 ± 0.30
5	17.72	17.15	3.18	25	0.71	0.43 ± 0.24
6	17.87	16.25	9.03	26	0.69	0.68 ± 0.16
7	19.14	21.47	-12.16	26	0.74	0.38 ± 0.15
8	17.45	15.57	10.79	24	0.73	0.58 ± 0.22
9	17.96	19.32	-7.58	24	0.75	0.35 ± 0.25
10	18.75	15.51	17.29	24	0.78	0.37 ± 0.28
AVG	**18.12**	**17.31**	**4.51**	**25**	**0.73**	$\mathbf{0.50 \pm 0.25}$

Table 1: Statistics of homing trajectories and homing accuracy over 10 closed-loop flight tests

5.2 Repetitive Navigation between Home and Goal Locations

Figure 5 displays homing trajectories followed by the rotorcraft when navigating back and forth several times between the home and goal locations, during a closed-loop flight test. Because the route learning module is run concurrently with the route following method during the first return to the home location, a new route that describes the inward journey followed by the rotorcraft is learned. This is achieved by memorising a sequence of snapshots separated by a distance d, using the snapshot taken at the goal location as initial reference and according to the algorithm described in Section 3.2. The new learned route is independent of the previous learned route, and only describes the path that the rotorcraft undertakes to get to the home position. Consequently, this new route is now used to head towards the goal (the route was learnt on the inward journey but is used by the route following module to navigate on the outward journey towards the goal location). This process is then repeated multiple times, in a similar way, to navigate between the home and goal locations.

Figure 5 shows 7 successive paths taken by the rotorcraft to navigate between home and the goal, after an initial exploratory journey that defines the goal location. Statistics of the consecutive homing trajectories are presented in Table 2. The mean home and goal position errors are computed from ground truth data. They demonstrate accurate homing to both locations (errors are less than the radius of the catchment area). On av-

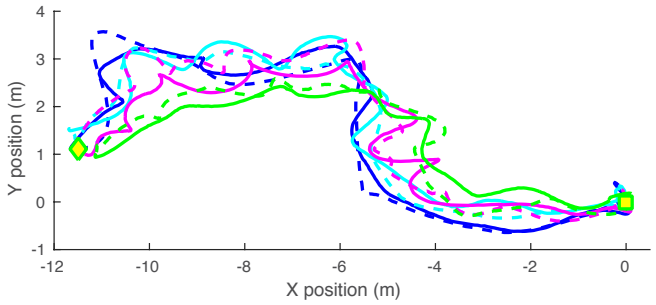

Figure 5: Navigation back and forth between the home and goal locations (pictured respectively as a square and a diamond): the outward journeys are represented by dashed lines and the inward ones by solid lines (ground truth data). The exploratory journey (dashed blue) excepted, all journeys (towards the home location and towards the goal position) are conducted by performing long-range homing using the route learning and navigation strategy described in Section 3. The successive runs are depicted in blue, cyan, magenta and eventually green.

Path direction (Home to Goal or Goal to Home)	Path length (m)	Path reduction (%)	Route size (snapshots)	Route size reduction (%)	Home position error (mean ± std dev., m)	Goal position error (mean ± std dev., m)
$H_0 \rightarrow G_0$	18.54	/	25	/	/	/
$G_0 \rightarrow H_1$	20.01	-7.94	25	0	0.58 ± 0.34	/
$H_1 \rightarrow G_1$	17.50	12.56	23	8.00	/	-0.095 ± 0.22
$G_1 \rightarrow H_2$	19.62	-12.09	24	-4.35	0.45 ± 0.34	/
$H_2 \rightarrow G_2$	20.94	-6.72	24	0	/	-0.10 ± 0.17
$G_2 \rightarrow H_3$	20.18	3.63	23	4.17	0.48 ± 0.36	/
$H_3 \rightarrow G_3$	18.14	10.10	20	13.04	/	-0.54 ± 0.19
$G_3 \rightarrow H_4$	15.23	16.03	19	5.00	0.51 ± 0.32	/
AVG	**18.77**	**2.22**	**22.88**	**3.69**	$\mathbf{0.50 \pm 0.34}$	$\mathbf{-0.25 \pm 0.19}$
$\mathbf{H_0 \rightarrow H_4}$	**150.16**	**17.86**	/	**24**	/	/

Table 2: Statistics of consecutive homing trajectories during a closed-loop flight test

erage, we observe a path reduction of 2.22% between two consecutive trajectories, with the route size (number of memorised snapshots) decreasing by 3.69%. This relatively slow path optimisation results from occasional sinuous trajectories, as seen in Figure 5, that were caused by wind gusts. However, overall, we observe a 17.86% path reduction after 7 runs, with the number of memorised snapshots decreasing by 24%.

6 Discussion of Results and Limitations

The results presented in Table 1 show a small path reduction when the rotorcraft is tasked to follow the learned route. This is due to our approximation of a snapshot location being reached when the quadrotor enters its vicinity (half the size of the catchment area). In a cluttered environment where route following needs to be accurate, the size of the snapshot vicinity needs to be decreased to the accuracy with which the UAS platform can reach a waypoint so that it enables safe navigation back to the home location.

Conversely, in the case of navigating in a more open environment, the path reduction that can be observed in Figure 5 and Table 2 when travelling back and forth between two locations of interest becomes a strength, as it enables optimisation of the journey (i.e. obtaining a straighter trajectory between the home and goal locations) and consequently a reduction of the size of the route description (i.e. the number of snapshots required to define the route). In our test, after 7 homing runs, the required visual memory had decreased by 24%. On the one hand, a large snapshot vicinity is likely to enable faster homing and better route optimisation, but this requires strongly overlapping catchment areas during route learning. On the other hand, if the SV is reduced, long-range homing requires each snapshot waypoint to be approached closely, and this is unlikely to result in significant route optimisation.

The homing accuracy results (see Table 1) demonstrate that our navigation strategy is drift-free. This is also the case during repetitive navigation where both home and goal locations are retrieved, with home being reached accurately after a 150m long travelled journey (see Table 2). As a comparison, if the same journey was travelled using a frame-to-frame OF odometer (similarly to [Strydom *et al.*, 2014]), the navigation would result in a 3D position error of 3.1m in reaching the home location (computed offline from the video footage of the repetitive navigation flight test).

Contrary to [Matsumoto *et al.*, 1996; Vardy, 2006], our navigation approach does not require the storage of an action or direction vector associated with each snapshot waypoint to perform homing, hence it enables a simple route structure. Moreover, route learning does not rely on path integration, when compared to the parsimonious augmented ALV (average landmark vector) scheme presented in [Smith *et al.*, 2006] that requires initialisation of the homing algorithm.

In our approach, the rotorcraft's 3D state is fully estimated relative to the current route waypoint through our snapshot-based OF method. Although demonstrating that both 3D position and velocity can be determined by using the memorised snapshots and used for autonomous guidance, our navigation strategy remains limited by the size of the catchment areas. Indeed, outside the CAs, the position estimates become rapidly noisy, making it difficult to estimate the rotorcraft's 3D velocity. In the case of strong wind gusts, the UAS is likely to move outside the CA without being able to retrieve it, thus resulting in homing failures. One solution would be to use

frame-to-frame OF to estimate the UAS' state while the platform is outside the CA, and to bring it back to the CA before resuming the use of our snapshot-based OF method. However, this corrective solution requires the ability to detect that the UAS is leaving the CA. This can be done by monitoring the reprojection error (when estimating the UAS' egomotion from the OF measurements) that would increase rapidly in such a situation.

7 Conclusion

In this paper we present a bio-inspired, mapless approach to visual navigation, based on route learning and following, applied to real-time control and guidance of rotorcraft UAS in natural environments. Our approach offers a lightweight navigation framework, based on snapshot matching, that creates a minimalistic route description (where only panoramic images are stored) and enables drift-free long-range homing by successively reaching a sequence of learned route waypoints. The rotorcraft's 3D position and velocity are estimated relative to the snapshot waypoints without relying on path integration or frame-to-frame optic flow computation. Moreover, our results demonstrate that this simple route description can still lead to route optimisation when navigating in an open environment. After successive homing runs back and forth between two locations of interest, the trajectories are reduced in length, as is the number of memorised snapshots. Additionally, by using view-based strategies for route learning and long-range homing similar to those postulated for insect navigation [Collett and Collett, 2002], we successfully demonstrated the viability of these biological hypotheses.

Further flight testing will aim to demonstrate that these homing paths converge towards a straight line between two locations, and minimal memory usage. Additionally, inspired by insect navigation that uses both path integration and snapshots to navigate to a goal [Wehner *et al.*, 2014], future work will investigate a snapshot-vector navigation strategy to provide a more sparse route representation and a faster route optimisation process when navigating a given journey repeatedly.

Acknowledgments

Sincere thanks to R. Strydom for his assistance during the flight experiments. The research described here was supported partly by the ARC Centre of Excellence in Vision Science (CE0561903), by Boeing Defence Australia Grant SMP-BRT-11-044, ARC Linkage Grant LP130100483, ARC Discovery Grant DP140100896, a Queensland Premier's Fellowship, and an ARC Distinguished Outstanding Researcher Award (DP140100914).

References

[Ahrens *et al.*, 2009] Spencer Ahrens, Daniel Levine, Gregory Andrews, and Jonathan P. How. Vision-Based Guidance and Control of a Hovering Vehicle in Unknown, GPS-denied Environments. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 2643–2648. IEEE, 2009.

[Argyros *et al.*, 2005] Antonis A. Argyros, Kostas E. Bekris, Stelios C. Orphanoudakis, and Lydia E. Kavraki. Robot Homing by Exploiting Panoramic Vision. *Autonomous Robots*, 19(1):7–25, 2005.

[Baddeley *et al.*, 2011] Bart Baddeley, Paul Graham, Andrew Philippides, and Philip Husbands. Holistic visual encoding of ant-like routes: Navigation without waypoints. *Adaptive Behavior*, 19(1):3–15, January 2011.

[Cartwright and Collett, 1983] B. A. Cartwright and Thomas S. Collett. Landmark Learning in Bees. *Journal of Comparative Physiology*, 151(4):521–543, 1983.

[Cartwright and Collett, 1987] B. A. Cartwright and T. S. Collett. Landmark Maps for Honeybees. *Biological Cybernetics*, 57(1-2):85–93, 1987.

[Cesetti *et al.*, 2010] A. Cesetti, E. Frontoni, A. Mancini, P. Zingaretti, and S. Longhi. A Vision-Based Guidance System for UAV Navigation and Safe Landing using Natural Landmarks. *Journal of Intelligent and Robotic Systems*, 57(1-4):233–257, January 2010.

[Chen and Birchfield, 2006] Zhichao Chen and Stanley T. Birchfield. Qualitative Vision-Based Mobile Robot Navigation. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2686–2692. IEEE, 2006.

[Collett and Collett, 2002] Thomas S. Collett and Matthew Collett. Memory use in insect visual navigation. *Nature Reviews Neuroscience*, 3(7):542–552, July 2002.

[Courbon *et al.*, 2009] J. Courbon, Y. Mezouar, N. Guenard, and P. Martinet. Visual navigation of a quadrotor aerial vehicle. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009*, pages 5315 –5320, October 2009.

[Denuelle and Srinivasan, 2015] Aymeric Denuelle and Mandyam V. Srinivasan. Bio-inspired Visual Guidance: from Insect Homing to UAS Navigation. In *Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on*. IEEE, 2015. (accepted).

[Denuelle *et al.*, 2015a] Aymeric Denuelle, Reuben Strydom, and Mandyam V. Srinivasan. Snapshot-based Control of UAS Hover in Outdoor Environments. In

Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on. IEEE, 2015. (accepted).

[Denuelle *et al.*, 2015b] Aymeric Denuelle, Saul Thurrowgood, Farid Kendoul, and Mandyam V. Srinivasan. A view-based method for local homing of unmanned rotorcraft. In *Automation, Robotics and Applications (ICARA), 2015 6th International Conference on*, pages 443–449. IEEE, 2015.

[Dyer, 1996] F. Dyer. Spatial memory and navigation by honeybees on the scale of the foraging range. *The Journal of Experimental Biology*, 199(1):147–154, 1996.

[Engel *et al.*, 2012] Jakob Engel, Jürgen Sturm, and Daniel Cremers. Accurate Figure Flying with a Quadrocopter Using Onboard Visual and Inertial Sensing. *IMU*, 320:240, 2012.

[Fu and Hsiang, 2012] Yu Fu and Tien-Ruey Hsiang. A fast robot homing approach using sparse image waypoints. *Image and Vision Computing*, 30(2):109–121, February 2012.

[Jones *et al.*, 1997] Stephen D. Jones, Claus Andresen, and James L. Crowley. Appearance Based Process for Visual Navigation. In *Intelligent Robots and Systems, 1997. IROS'97. Proceedings of the 1997 IEEE/RSJ International Conference on*, volume 2, pages 551–557. IEEE, 1997.

[Kendoul *et al.*, 2009] Farid Kendoul, Isabelle Fantoni, and Kenzo Nonami. Optic flow-based vision system for autonomous 3D localization and control of small aerial vehicles. *Robotics and Autonomous Systems*, 57(6-7):591–602, June 2009.

[Labrosse, 2007] Frédéric Labrosse. Short and long-range visual navigation using warped panoramic images. *Robotics and Autonomous Systems*, 55(9):675–684, September 2007.

[Li *et al.*, 2013] Ping Li, Matthew Garratt, Andrew Lambert, Mark Pickering, and James Mitchell. Onboard Hover Control of a Quadrotor using Template Matching and Optic Flow. In *World Congress in Image Processing*, 2013.

[Li *et al.*, 2015] Ping Li, Matthew Garratt, and Andrew Lambert. Monocular Snapshot-based Sensing and Control of Hover, Takeoff, and Landing for a Low-cost Quadrotor: Monocular Snapshot-based Sensing and Control. *Journal of Field Robotics*, March 2015.

[Matsumoto *et al.*, 1996] Yoshio Matsumoto, Masayuki Inaba, and Hirochika Inoue. Visual Navigation using View-Sequenced Route Representation. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 1, pages 83–88. IEEE, 1996.

[Nelson and Aloimonos, 1988] Randal C. Nelson and John Aloimonos. Finding Motion Parameters from Spherical Motion Fields (Or the Advantages of Having Eyes in the Back of Your Head). *Biological Cybernetics*, 58(4):261–273, 1988.

[Ohno *et al.*, 1996] Takayuki Ohno, Akihisa Ohya, and Shin Ichi Yuta. Autonomous Navigation for Mobile Robots Referring Pre-recorded Image Sequence. In *Intelligent Robots and Systems, 1996. IROS'96. Proceedings of the 1996 IEEE/RSJ International Conference on*, volume 2, pages 672–679. IEEE, 1996.

[Powell, 2009] Michael JD Powell. The BOBYQA algorithm for bound constrained optimization without derivatives. 2009.

[Romero *et al.*, 2009] H. Romero, S. Salazar, and R. Lozano. Real-Time Stabilization of an Eight-Rotor UAV Using Optical Flow. *IEEE Transactions on Robotics*, 25(4):809–817, August 2009.

[Sagüés and Guerrero, 2005] C. Sagüés and J.J. Guerrero. Visual correction for mobile robot homing. *Robotics and Autonomous Systems*, 50(1):41–49, January 2005.

[Smith *et al.*, 2006] Lincoln Smith, Andrew Philippides, and Phil Husbands. Navigation in Large-Scale Environments Using an Augmented Model of Visual Homing. In Stefano Nolfi, Gianluca Baldassarre, Raffaele Calabretta, John C. T. Hallam, Davide Marocco, Jean-Arcady Meyer, Orazio Miglino, and Domenico Parisi, editors, *From Animals to Animats 9*, number 4095 in Lecture Notes in Computer Science, pages 251–262. Springer Berlin Heidelberg, January 2006.

[Smith *et al.*, 2007] Lincoln Smith, Andrew Philippides, Paul Graham, Bart Baddeley, and Philip Husbands. Linked Local Navigation for Visual Route Guidance. *Adaptive Behavior*, 15(3):257–271, January 2007.

[Strydom *et al.*, 2014] Reuben Strydom, Saul Thurrowgood, and M. Srinivasan. Visual Odometry: Autonomous UAV Navigation using Optic Flow and Stereo. In *Australasian Conf. on Robotics and Automation (ACRA)(Melbourne)*, 2014.

[Thurrowgood *et al.*, 2014] Saul Thurrowgood, Richard J. D. Moore, Dean Soccol, Michael Knight, and Mandyam V. Srinivasan. A Biologically Inspired, Vision-based Guidance System for Automatic Landing of a Fixed-wing Aircraft: Automatic Landing of Fixed-wing Aircraft. *Journal of Field Robotics*, 31(4):699–727, July 2014.

[Vardy, 2006] A. Vardy. Long-Range Visual Homing. In *IEEE International Conference on Robotics and Biomimetics, 2006. ROBIO'06*, pages 220 –226, December 2006.

[Wehner *et al.*, 2014] Rüediger Wehner, Ken Cheng, Holk Cruse, and others. Visual navigation strategies in insects: lessons from desert ants. *New visual neurosciences. MIT Press, Cambridge, MA*, pages 1153–1163, 2014.

[Weiss *et al.*, 2013] Stephan Weiss, Markus W. Achtelik, Simon Lynen, Michael C. Achtelik, Laurent Kneip, Margarita Chli, and Roland Siegwart. Monocular Vision for Long-term Micro Aerial Vehicle State Estimation: A Compendium. *Journal of Field Robotics*, 30(5):803–831, September 2013.

[Zhou *et al.*, 2003] Chao Zhou, Yucheng Wei, and Tieniu Tan. Mobile Robot Self-Localization Based on Global Visual Appearance Features. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 1, pages 1271–1276. IEEE, 2003.

[Zufferey and Floreano, 2006] J.-C. Zufferey and D. Floreano. Fly-Inspired Visual Steering of an Ultralight Indoor Aircraft. *IEEE Transactions on Robotics*, 22(1):137–146, February 2006.