2016 IEEE International Conference on Robotics and Automation (ICRA)
Stockholm, Sweden, May 16-21, 2016

A Sparse Snapshot-based Navigation Strategy for UAS Guidance in Natural Environments

Aymeric Denuelle[1,2] and Mandyam V. Srinivasan[1,3]

Abstract— **This paper presents a novel visual navigation strategy applied to real-time guidance and control of autonomous rotorcraft in outdoor environments. The proposed approach acquires a sequence of snapshots during an initial exploratory journey and computes the rotorcraft's 3D position and velocity relative to the locations of these memorised snaphots during its long-range homing return. This mapless algorithm provides an efficient strategy that enables route learning and drift-free navigation in unknown and unstructured 3D environments, based on a simple and sparse visual route description. The performance of our approach is evaluated in closed-loop flight experiments onboard a small-size rotorcraft. Dynamic path optimisation is observed when navigating back and forth between two locations of interest.**

I. Introduction

Nowadays, unmanned aerial systems (UAS), and especially rotorcraft UAS, are becoming popular solutions to a wide and growing range of civilian applications. UAS can replace human pilots in repetitive, possibly hazardous and extended flying operations in applications such as precision agriculture, environmental surveying, or search and rescue missions.

Vision has become a preferred sense in low-altitude flying operations where traditional satellite-based autopilots can become unreliable due to signal reflections or obstructions by nearby objects. Moreover, multi-purpose imaging sensors are an asset for applications onboard lightweight UAS with limited payload capacity where they can replace bulkier sensors (e.g. laser range-finders) and enable fully autonomous navigation in addition to conducting specific visual tasks (collision avoidance, object tracking, scene understanding, etc). However, vision-based solutions often have high computational power requirements, or rely on high-resolution imagery, to offer efficient, practical and real-time solutions for small-size UAS navigation.

Since insects, such as bees or ants, are remarkable navigators despite their low-resolution eyes and limited brain capacities, biological observations have been a source of inspiration for overcoming some of the limitations of vision-based algorithms for robot navigation. For example, bees memorise and pinpoint locations by using visual snapshots of the surrounding panorama [1]. This behaviour of retrieving a pre-visited location by using visual cues is called visual homing. In this paper, we address the problem of 'long-range homing', which consists of navigating back to a location that is outside the currently observable visual scene – as opposed to 'local homing', which involves moving toward a goal location that is currently visible. Biological experiments provide evidence that bees are able to store multiple routes, and do not need to retain their spatial relationships [2]. As insects do not seem to rely on maps, this work is motivated by a challenge to perform visual navigation using a mapless approach that relies on a sparse route description, and applicable to the real-time guidance of small-size UAS in outdoor environments. Low-altitude and long-term missions that require refueling or cargo refill, such as search and rescue deployments or spraying operations in agriculture, would benefit from this capability. Additionally, it would provide an alternative to satellite-based autopilots during short-term signal outages, to take over control and ensure safe and reliable navigation.

The main contributions of this paper are the acquisition of a minimal snapshot-based route that describes an exploratory journey between two locations of interest, and the development of a navigation strategy that enables long-range homing by fully estimating the UAS' state (3D position and velocity) relative to the locations of the memorised snapshots taken along the learned route. Importantly, the study also provides a validation of view-based theories of insect navigation [3] by demonstrating that they can be used successfully to guide a rotorcraft along three-dimensional routes.

The remainder of this paper is organised as follows: Section II introduces the relevant literature and Section III presents our long-range homing navigation strategy, while Section IV describes the rotorcraft platform. Results from outdoor closed-loop flight tests are presented in Section V, the performance of our approach is discussed in Section VI, and finally, Section VII presents the conclusions of the paper.

II. Related Work

Visual navigation techniques can be broadly separated into map-based approaches that focus on building and updating a map of the environment, and mapless approaches that rather rely on visual odometry or on a previously learned route. Visual simultaneous localisation and mapping (VSLAM) methods [4], [5] traditionally perform the costly extraction and matching of hundreds of features, or require accurate image alignment [6], to estimate the pose of a rotorcraft. These operations require sufficient image resolution and

Authors are with the Queensland Brain Institute[1] and School of Information Technology and Electrical Engineering[3], The University of Queensland, St Lucia, Australia, and the Autonomous Systems Program[2], CSIRO, Pullenvale, Australia. Corresponding author: a.denuelle@uq.edu.au

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).

computational power to be executed in real-time onboard small-size UAS.

For the purpose of visual homing, a map (in the sense of, for example, [7]) is not obligatory nor is the extraction of features. Navigation can rely on visual odometry, with optic flow (OF) being a widely used technique for the stabilisation and guidance of UAS. Indeed, the computation of OF does not require the identification or tracking of explicit landmarks within a visual scene. OF-based navigation techniques usually rely on intensity gradient-based methods [8], [9], block matching techniques [10], or simpler, non-iterative algorithms [11], [12]. However, such methods often require the fusion of the visual information with additional sensor data, e.g. inertial measurements [9], [13], to overcome the drift in their position estimates resulting from the integration of cumulative odometric errors. Alternatively, computing OF with regard to a particular snapshot enables drift-free pose stabilisation [14], [15] or navigation [16] for rotorcraft UAS.

In principle, rather than navigating purely by dead-reckoning, one can rely on a previously learned visual route (during an exploratory journey) to retrieve a pre-visited location. There is evidence that insects, such as bees, make use of a route memory to retrieve a place in complex environments [17]. [18] proposes a holistic route description where a classifier is trained to distinguish between positive forward-facing views belonging to the travelled route and negative ones (sideways-facing views). Homing is then achieved by visually scanning the environment and moving in the direction along which the views are positively classified (as belonging to the route). However, a visual route is traditionally composed of a sequence of discrete waypoints, representing visual locales and defined by either local or global descriptors. In this case, navigation consists in successively reaching each visual region by using a local homing method. Thus, route-based navigation is often implemented by concatenating a sequence of local homing behaviours (see [19] for a review of bio-inspired techniques).

A visual locale can be represented by a global descriptor such as the whole image [16], [20]–[24], or a multi-dimensional image histogram based on colour, edge density, intensity gradient and texture [25]. Local descriptors, extracted from input images, can be used in lieu of global parameters to describe a visual locale: they can be feature points (image corners [26], [27], SIFT points [28]), average landmark vectors [29], intensity patterns and their image coordinates [30], or geometric features (lines) [31], [32]. In addition to reducing the memory required to store a route, local descriptors can enable simple, qualitative, navigation schemes [30]. However, one remaining limitation is the life-cycle of these local descriptors, which need to be extracted in sufficient number but also successfully tracked throughout the visual locales [26], [30]. This advocates the use of panoramic images [26], also used in general in image-based approaches to decrease the sensitivity to occlusions by dynamic objects, and to enable route following from a different direction.

In this paper, we propose a mapless visual navigation strategy that is applied to real-time guidance and control of a small-size rotorcraft, without requiring any additional data fusion with other onboard sensors. Our approach makes use of omnidirectional images to build a sparse and minimal visual route description, without requiring feature extraction. Drift-free long-range homing is achieved in outdoor environments by using a snapshot matching technique to fully estimate the rotorcraft's state (3D position and velocity) relative to the locations of the memorised panoramic images.

III. Route Learning and Autonomous Navigation

This section describes our navigation strategy, i.e. the route learning process that occurs between the home and goal locations (exploratory journey), and explains how the acquired route description can enable long-range homing back to the initial location. At the core of both modules is a snapshot-based method that estimates the UAS' pose relative to the location of a particular memorised view. During route learning, these egomotion estimates are used to select key frames that constitute the visual route. During long-range homing, the egomotion estimates are used to control the rotorcraft's position and velocity so that it follows the previously acquired route by reaching each of the learned snapshot locations (i.e. by performing sucessive local homing steps).

A. Snapshot-based Estimation of Egomotion

In our navigation strategy, we use a hierarchical, block-matching algorithm that computes sparse optic flow (400 points), with subpixel precision, between the current camera frame and a particular memorised snapshot, similarly to [15], [16]. This snapshot-based computation of optic flow prevents the cumulative odometric errors that would occur when using traditional frame-to-frame integration of OF measurements [15].

The onboard vision system (described in Section IV-A) delivers 360x180 pixel panoramic images that are used to compute the optic flow vectors (an example is displayed in Fig. 4(b)). An optimised iterative process, described in [33], is applied to these OF measurements (400 points) to determine the best combination of 3D translation and rotation that minimises the reprojection error, assuming a planar ground. 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.

These egomotion estimates are then scaled with stereo measurements of height (also provided by the vision system) to obtain a metric 3D position signal that is fed into the control loop (see Section IV-B) along with the velocity signal. The rotorcraft's velocities are directly computed by differentiating and low-pass filtering the position estimates delivered by the snapshot-matching OF-based technique (OF-SM) described above.

B. Route Description

In this section we use the OF-SM method to build a sparse route description, composed of a series of duples which

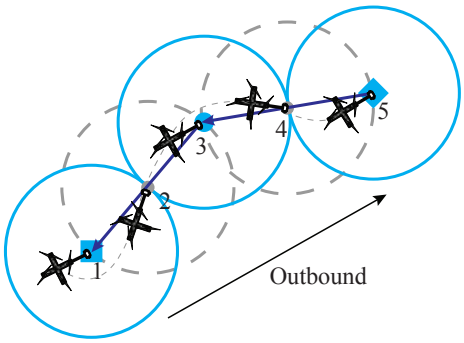


Fig. 1. An example of route learning between the home (square) and goal (diamond) locations. In addition to storing the snapshots at the home and goal positions, the rotorcraft memorises a snapshot waypoint (indicated by a blue dot) along with an associated vector (displayed in dark blue) pointing towards the center of the previous CA (i.e. home in this case). CAs are represented by blue circles, while the temporarily memorised snapshot locations and corresponding CAs are displayed by grey dots and dashed circles, respectively.

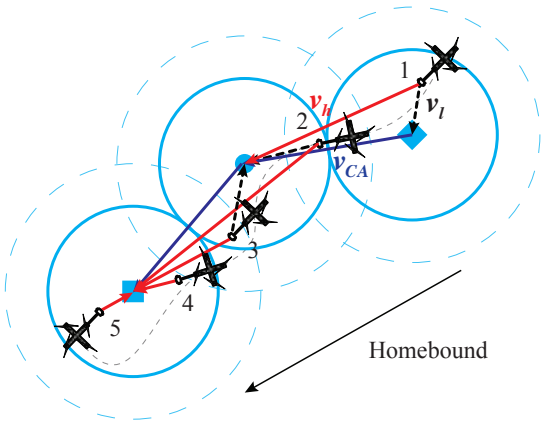


Fig. 2. An example of long-range homing from the goal (diamond) to the home (square) location, with the successive positions of the rotorcraft (location of the frontal vision system) indicated by increasing numbers. Tangential CAs are represented by blue solid circles of radius $r < R_{CA}$ (conservative value) whereas the dashed, overlapping circles have a radius equal to R_{CA}, thus enabling the robust crossing from one CA to another. Within a CA, the dashed vectors are the relative position estimates to the snapshot waypoint location, as provided by the OF-SM method. Memorised inter-CA vectors are displayed in dark blue. Homing vectors followed by the rotorcraft are shown in red.

associate a snapshot (representing a visual locale) with a vector pointing at the center of the previous visual locale.

A panoramic snapshot can be used to describe a particular location and defines, in this case, a 3D visual locale (or 'catchment area') within which local homing, for a given method, is successful. We represent here the catchment area (CA) associated with the OF-SM method by a sphere of radius R_{CA} equal to about 0.3 height units [34]. Consequently, a visual route can be defined by memorising snapshots taken at regular intervals separated by a maximal distance $d = R_{CA}$. If the exploratory (outbound) journey is conducted at a constant height, then this route description results in catchment areas that overlap by an amount of 50%.

However, in order to decrease the memory usage, we aim at building a minimal route where the catchment areas do not overlap (or are minimally overlapping if a conservative value, smaller than 0.3 height units, is chosen for R_{CA}). This absence of overlap (or minimum overlap) occurs when the snapshot waypoints are separated by a distance $d = 2 \times R_{CA}$, which results in tangential CAs. This case of touching CAs is the most sparse route description that can be obtained without relying on path integration, while ensuring that the rotocraft is always within one CA at any time during long-range homing. This enables the current view to be matched with a snapshot waypoint at each time step, thus providing continuous estimates of the rotorcraft's state.

In our approach, once an initial snapshot is memorised at the home location, a new snapshot is taken every time the rotorcraft leaves the current CA. This information is provided by the OF-SM method, which outputs the 3D relative position to the last memorised snapshot waypoint location. As soon as the rotorcraft is located at a distance $d \geq R_{CA}$ from the last route waypoint (hence exiting the current CA), both the input panoramic image and the vector to the previous waypoint location are temporarily stored in memory as a duple. This snapshot is set as the new reference view in the OF-SM method (it defines a new CA, overlapping by 50% with the previous one) and is used to decide when the next snapshot should be taken, following the same approach. However, only every second snapshot is eventually retained as part of the memorised route, along with a resultant vector that points towards the previously learned snapshot (belonging to the route). This resultant vector is obtained by adding the current inter-CA vector to the previous one (temporarily stored in a duple). Only the memorising of a new CA entails the addition of two vectors, both computed using the OF-SM method. Consequently, a sparse route description with tangential CAs is obtained, without resorting to path integration. An illustrative example of the route learning process is displayed in Fig. 1.

C. Route-based Navigation

Based on the above route description, the navigation algorithm presented here relies on the assumption that, when the rotorcraft leaves the current CA, it has entered the CA of the next snapshot waypoint belonging to the learned route.

The homeward return is initiated within the CA of the snapshot taken at the goal location. Then the UAS navigates in the direction that points towards the center of the next CA. This homing vector v_h is obtained by summing the inter-CA vector v_{CA} (associated with the current snapshot waypoint and recalled from memory) with the local vector v_l (obtained through the OF-SM method), which points towards the center of the current CA. Therefore, from any position within the current CA, the homing vector to the next snapshot location is defined by $v_h = v_l + v_{CA}$, as illustrated in Fig. 2.

Once the rotorcraft has left the current CA (indicated by $v_l \geq R_{CA}$), the OF-SM switches to the next snapshot waypoint (in the reverse order of the learned route) to estimate the UAS' state within the new CA. The inter-CA vector associated with the new snapshot is then added to the local vector v_l to obtain a new homing vector pointing toward the

(a) (b)

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

next snapshot location. This process of successive switching and short-cutting is repeated each time the rotorcraft leaves a CA, until the home location is reached.

The above description of route following pertains to the case of tangential CAs, which is the expected nature of the learned route. However, some largely overlapping CAs can also be part of the route description. Such a situation could occur, for example, when the rotorcraft faces a strong wind gust, or performs a sharp turn that sends the UAS back to a previously learned, and subsequently exited, CA. Overlapping CAs can also be memorised when the UAS performs a sharp turn, as seen in Fig. 6.

During the inbound journey back to the home location, the route learning module is run concurrently with the route following module. This enables the learning of a new, shorter route description between the goal and home locations that can be used to return to the goal during a subsequent journey (see Section V).

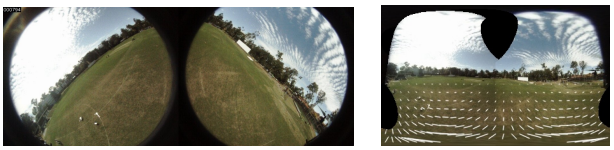
IV. Flight Platform and Sensor Suite

This section describes the experimental platform, its onboard sensors, and gives an overview of the control structure that enables closed-loop navigation.

Our flight platform is a small-size, custom-built, quadrotor which embeds a dual camera system located at the front, as pictured in Fig. 3(a), an Intel NUC computer (2.6 GHz dual-core processor), a MicroStrain 3DM-GX3-25 AHRS, and a Piksi GPS receiver [35] (for ground truthing purposes only, see Section V-A).

A. Bio-inspired Vision System

Our rotorcraft is equipped with a biologically-inspired visual system [33] that comprises two miniature fish-eye cameras, each possessing a 190° field of view. They are mounted back-to-back and tilted towards each other by 10°, as pictured in Fig 3(b), thus providing a near-panoramic vision as well as a frontal region of stereo vision. The panoramic vision, as employed by insects, simplifies the computation of egomotion by making translation easily distinguishable from rotation [36]. The stereo overlap (central strip of the panoramic images) provides a means of estimating the height in flight and therefore enables conversion of the OF measurements (obtained in the OF-SM method for pose estimation) into the metric values that are required by the control loop described in Section IV-B. The left and right images from the cameras (as shown in Fig. 4(a)) are software



(a) (b)

Fig. 4. Left and right images as captured by the onboard vision system (a), and the stitched, single viewpoint, panoramic image (before its greyscale conversion) used in the OF-SM method, with measured optic flow vectors superimposed (b).

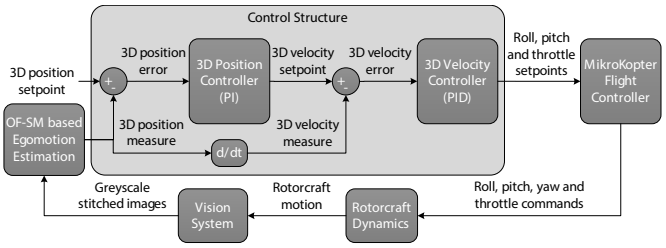


Fig. 5. Overview of the control structure used to regulate the rotorcraft's state to reach the successive snapshots' locations along a previously learned route.

synchronised, converted to grayscale, and stitched together to form a 360x180 pixel, single viewpoint, panoramic image, available at a 25Hz frame rate for the computation of the rotorcraft's pose estimates (see Fig. 4(b)).

B. Onboard Control Loop

During the closed-loop flight tests (see Section V), only information provided by our vision system (panoramic images) and the onboard AHRS (Euler angles) is used in the control loop to regulate the quadrotor's 3D state (position and velocity). Fig. 5 shows the control structure, where 3 cascaded position-velocity (PI-PID) controllers are used in conjunction with a MikroKopter flight controller [37] to regulate the rotorcraft's state and attitude in flight, respectively. This control loop runs at the camera frame rate (25Hz) and provides attitude and throttle commands for the guidance of the rotorcraft. The snapshot-based method described in Section III provides the position controller with position estimates relative to the snapshots acquired during route learning, hence we use a constant 3D position setpoint set to $(0, 0, 0)$. The position estimates are compared to this setpoint and a 3D velocity setpoint is output by the position controller. In the velocity controller, the quadrotor's velocities (derived from the visual position estimates) are compared to the 3D velocity setpoint (output by the position controller) to determine the roll, pitch and thrust setpoints that are then fed into the MikroKopter attitude stabilisation loop. In addition to providing information that is used to deliver the roll, pitch and thrust commands, the AHRS measurements are also used in the MikroKopter flight controller to deliver a yaw command.

V. Closed-loop Flight Experiments

This section presents the results from a number of outdoor closed-loop flight tests where the navigation strategy de-

scribed in Section III is used to perform long-range homing, back to the home location, but also repetitive navigation between the home and goal locations.

A. Flight Parameters and Experimental Setup

In the flight experiments, we used a conservative value of 0.2 height units for the radius of the CAs. As seen in Fig. 2, this enables robust crossing of tangential CAs since they overlap when using a value of 0.3 height units for R_{CA}.

Initially, the rotorcraft is commanded to follow a pre-set square-wave trajectory (i.e. two segments of length 6m, laterally separated by two segments of length 4m, resulting in a total length of 20m), as pictured in Fig. 6, at a height above ground of about 5m. This exploratory journey, starting at the home location, defines the goal location at its termination. It is performed using a frame-to-frame OF-based method, similarly to [10]. Autonomous take-off and landing are also based on this scheme (before and after the navigation tests). After this initial journey, navigation fully relies on the OF-SM method described in Section III-A.

Three distinct scenarios are tested. First, the quadrotor is simply tasked to perfom long-range homing, from the goal location, and retrieve the home position. Secondly, the rotorcraft's mission is defined as retrieving the goal location, following its return to the home location from the goal. In a last scenario, the UAS is required to travel multiple times between the home and goal locations. During repetitive navigation, the quadrotor is tasked to hover at each of the locations of interest for a duration of 5s, before resuming the homing task (towards home or towards the goal).

All flight experiments were conducted in relatively low-wind conditions, with wind gusts up to 6 knots (measured at ground level), and monitored by a differential GPS (DGPS) setup to evaluate the performance of our navigation strategy. Ground truth positioning is provided by two Piksi GPS receivers [35], with real-time kinematics functionality, that deliver differential GPS measurements at a frequency of 5Hz. Although logged in real-time onboard the rotorcraft, this ground truth data is never used for guidance or control purposes. 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.

Results from the three scenarios are expressed in terms of the length of the travelled homing trajectories (as monitored by DGPS) between the home and goal locations, the size of the learned visual route (number of memorised snapshots) that describes the homing path, and the accuracy with which the rotorcraft reaches either home or the goal. Successive returns to the home location H and the goal G are labeled by H_i and G_i, respectively, with i denoting the number of times a location has been reached using our long-range homing strategy. H_0 defines the beginning of the exploratory journey (location of the home snapshot and CA) and G_0 indicates the end of this commanded trajectory as the goal location.

B. Flight Scenario 1: Home Retrieval

During the commanded exploratory journey, the route learning module runs continuously, memorising the outward route to the goal, thus enabling long-range homing back to the home location once the goal has been defined. An example of the route learning and long-range homing processes that occur during a closed-loop flight test is presented in Fig. 6. The CAs shown in the figure are not perfectly tangential for two reasons: the figure uses ground truth data, which is different from the estimated (by the OF-SM method) travelled positions during route learning, and secondly, a new snapshot is taken each time the estimated distance from the last memorised snapshot is greater than the R_{CA} value, hence resulting in only almost tangential CAs.

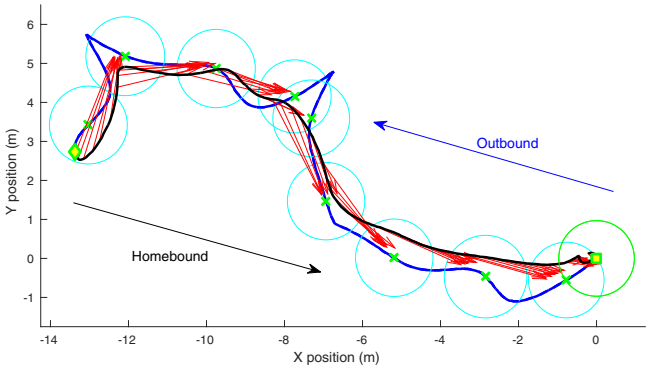


Fig. 6. Route learning and long-range homing in a closed-loop flight test: the home and goal locations are represented by a square and a diamond, respectively. The exploratory journey is shown in dark blue, with green crosses indicating the snapshot waypoint locations and the corresponding CAs, shown in cyan. The homing path back to the home location is represented in black, with the red vectors being the homing vectors that point towards the center of each memorised CA in reverse order.

Table I presents the results from 3 single long-range homing runs back to the home location, over 3 flight tests. An average 16.45% path reduction is observed between the exploratory journey and the path travelled during the homing run. On the way to the goal, a mean of 9.33 snapshots is needed to describe an averaged travelled path of 22.22m (journey between the border of the home CA to the border of the goal CA). It corresponds to a travelled distance of 2.38m between two snapshots, i.e. a mean CA radius of 1.19m, which is close to the expected value of 1m (R_{CA} = 0.2 height units, and the exploratory journey was set at an altitude of about 5m). In these tests, the home location was declared reached when the visual estimates indicate that the quadrotor is within a sphere of 0.5m at the home snapshot location. Table I demonstrates that home is successfully retrieved with arrival indicated at a mean distance of 0.48m from the center of the CA, as indicated by DGPS measurements.

C. Flight Scenario 2: Goal Retrieval

This scenario requires the rotorcraft to retrieve the goal location once it has travelled back to the home position, and thus tests whether both outward and inward journeys are successfully learned and described by a visual route.

The statistics of the homing paths (towards both home and the goal) are presented in Table II, where an overall path reduction of 16.09% is observed between the outward exploratory journey and the final homing run toward the

TABLE II

SCENARIO 2: STATISTICS OF HOMING TRAJECTORIES AND HOMING ACCURACY OVER 6 CLOSED-LOOP FLIGHT TESTS

	Path length (m)			Path reduction (%)			Route size (snapshots)			Route size reduction (%)			Home position error (m)		Goal position error (m)	
Flight test	$H_0 \rightarrow G_0$	$G_0 \rightarrow H_1$	$H_1 \rightarrow G_1$	$H_0 \rightarrow H_1$	$G_0 \rightarrow G_1$	$H_0 \rightarrow G_1$	$H_0 \rightarrow G_0$	$G_0 \rightarrow H_1$	$H_1 \rightarrow G_1$	$H_0 \rightarrow H_1$	$G_0 \rightarrow G_1$	$H_0 \rightarrow G_1$	Mean	Std dev.	Mean	Std dev.
4	20.95	19.06	19.72	9.03	-3.48	5.86	9	8	7	11.11	12.5	22.22	0.34	0.094	0.0027	0.26
5	22.66	17.86	19.50	21.17	-9.16	13.94	9	8	7	11.11	12.5	22.22	0.28	0.14	-0.47	0.15
6	22.81	17.88	19.00	21.62	-6.27	16.70	9	8	7	11.11	12.5	22.22	0.34	0.18	-0.55	0.24
7	24.54	18.95	20.53	22.76	-8.33	16.33	10	7	7	30	0	30	0.35	0.10	-0.31	0.17
8	19.62	16.39	14.45	16.42	11.84	26.32	8	6	7	25	-16.67	12.5	0.30	0.11	-0.58	0.49
9	21.66	20.90	17.90	3.51	14.37	17.37	9	8	8	11.11	0	11.11	0.34	0.070	-0.28	0.088
AVG	**22.04**	**18.51**	**18.52**	**15.75**	**-0.17**	**16.09**	**9**	**7.50**	**7.17**	**16.57**	**3.47**	**20.05**	**0.32**	**0.12**	**-0.36**	**0.23**

TABLE I

SCENARIO 1: STATISTICS OF HOMING TRAJECTORIES AND HOMING ACCURACY OVER 3 CLOSED-LOOP FLIGHT TESTS

Flight test	Outward path length (m)	Inward path length (m)	Path reduction (%)	Route size (snapshots)	Mean distance between snapshots (m)	Home position error (m)
1	24.25	19.08	21.30	10	2.42	0.48
2	23.16	17.47	24.59	9	2.57	0.48
3	19.26	18.60	3.45	9	2.14	0.49
AVG	**22.22**	**18.38**	**16.45**	**9.33**	**2.38**	0.48

goal. The number of memorised snapshots constituting the visual route decreases in average by 20.05% over the two homing runs, from 9 to 7.17 snapshots stored in memory. At the home and goal locations, hover is commanded once the visual estimates indicate that the quadrotor is within the snapshot vicinity, i.e. within a sphere of radius 0.5m centred at the snapshot location. The hover statistics demonstrate that both locations are successfully retrieved, with mean position errors of 0.32m and -0.36m, respectively (as indicated by DGPS data).

D. Flight Scenario 3: Repetitive Navigation

In this scenario, the UAS is tasked to navigate multiple times between the home and goal locations after its initial exploratory journey. Each newly learned route replaces the previously learned route (outbound or homebound). Table III presents the statistics of an average flight experiment (values computed from the results of 3 separate flight tests) where the rotorcraft is commanded, in each flight experiment, to reach the home and goal locations 3 times (from the end of the exploratory journey $H_0 - G_0$) before eventually returning home (H_4). This average flight has a total travelled length of about 130m along which successive journeys are reduced by 5.41%, resulting in a total path reduction of 33.57% between the exploratory and last travelled journeys. Overall, the size of the learned route decreases by 26.92% with an average reduction of 3.85% between successive journeys. Additionally, the home and goal position errors remain within the defined accuracy (i.e., half the radius of the CA, which is about 0.5m for a flight conducted at a height above ground of 5m), as measured by DGPS data and shown in Table III. This demonstrates that our navigation scheme enables drift-free navigation between two locations.

TABLE III

SCENARIO 3: STATISTICS OF CONSECUTIVE HOMING TRAJECTORIES AVERAGED OVER 3 CLOSED-LOOP FLIGHT TESTS (FLIGHTS #10-12)

Path direction	Path length (m)	Path reduction (%)	Route size (snapshots)	Route size reduction (%)	Mean home position error (m)	Mean goal position error (m)
$H_0 \rightarrow G_0$	21.36	/	8.67	/	/	/
$G_0 \rightarrow H_1$	19.04	10.83	7.00	19.23	0.44 ± 0.11	/
$H_1 \rightarrow G_1$	17.32	9.06	6.67	4.76	/	-0.076 ± 0.20
$G_1 \rightarrow H_2$	14.58	15.82	6.33	5.00	0.37 ± 0.16	/
$H_2 \rightarrow G_2$	15.52	-6.46	6.00	5.26	/	0.11 ± 0.31
$G_2 \rightarrow H_3$	14.35	7.52	5.67	5.56	0.50 ± 0.16	/
$H_3 \rightarrow G_3$	14.02	2.30	5.33	5.88	/	0.18 ± 0.12
$G_3 \rightarrow H_4$	14.19	-1.18	6.33	-18.75	0.60 ± 0.19	/
AVG	**16.30**	**5.41**	**6.5**	**3.85**	$\mathbf{0.48 \pm 0.16}$	$\mathbf{0.072 \pm 0.21}$
$\mathbf{H_0 \rightarrow H_4}$	**130.39**	**33.57**	/	**26.92**	/	/

Results from a longer experiment are presented in Table IV and the corresponding trajectories of the rotorcraft are displayed in Fig. 7. In this case, the quadrotor is tasked to reach each location of interest 9 times (from G_0 to H_{10}) before returning home. However, due to a low-battery alert, the statistics of the 10th journey towards home are not included as the UAS interrupted its homing journey and entered a safe landing procedure. Overall, the UAS travelled a distance greater than 300m during this experiment, with successive paths decreasing in length by an average of 5.52%, leading to a total path reduction of 41.90%. An average reduction of 2.33% between successive runs (home to goal or goal to home) is observed in the size of the learned route. Indeed, the number of memorised snapshots required to describe the route between the home and the goal locations reduces from 9 to 5 over the entire experiment, representing an overall reduction of 44.44%.

VI. DISCUSSION OF RESULTS AND LIMITATIONS

The results presented in Section V demonstrate that our proposed vision-based strategy enables drift-free navigation, with the home and goal locations accurately retrieved after short homing runs as well as after longer, repetitive homing runs. For example, in the case of the long homing run presented in Table IV, if the rotorcraft was navigating using a frame-to-frame OF-based method similar to [10], there would be a 3D position error of 9.2m in reaching the goal for the 9th time (this is the position error from G_0 to G_9, computed offline from the onboard video footage recorded during the

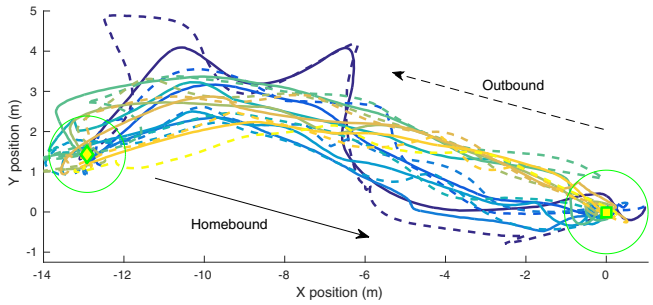


Fig. 7. Repetitive navigation between the home (square) and goal (diamond) locations: the outward journeys are represented by dashed lines and the inward ones by solid lines (DGPS data), with lightening colours representing the chronology (from dark blue to yellow). The initial exploratory journey (dashed dark blue) excepted, all journeys (towards both the home and goal locations) are obtained by route learning and route-based navigation, as described in Section III.

TABLE IV

SCENARIO 3: STATISTICS OF CONSECUTIVE HOMING TRAJECTORIES DURING A CLOSED-LOOP FLIGHT TEST (FLIGHT #13)

Path direction	Path length (m)	Path reduction (%)	Route size (snapshots)	Route size reduction (%)	Mean home position error (m)	Mean goal position error (m)
$H_0 \rightarrow G_0$	24.27	/	9	/	/	/
$G_0 \rightarrow H_1$	18.97	21.84	7	22.22	0.92 ± 0.22	/
$H_1 \rightarrow G_1$	16.17	14.72	6	14.29	/	-0.27 ± 0.068
$G_1 \rightarrow H_2$	14.72	8.99	6	0	0.69 ± 0.084	/
$H_2 \rightarrow G_2$	14.85	-0.85	7	-16.67	/	0.095 ± 0.11
$G_2 \rightarrow H_3$	14.36	3.32	7	0	0.98 ± 0.049	/
$H_3 \rightarrow G_3$	14.73	-2.64	6	14.29	/	0.83 ± 0.15
$G_3 \rightarrow H_4$	15.92	-8.02	7	-16.67	0.83 ± 0.056	/
$H_4 \rightarrow G_4$	14.09	11.50	6	14.29	/	0.61 ± 0.20
$G_4 \rightarrow H_5$	/	/	7	-16.67	/	/
$H_5 \rightarrow G_5$	21.56	/	8	-14.29	/	0.49 ± 0.47
$G_5 \rightarrow H_6$	17.47	18.98	7	12.5	0.98 ± 0.12	/
$H_6 \rightarrow G_6$	15.87	9.11	7	0	/	0.58 ± 0.67
$G_6 \rightarrow H_7$	15.71	1.01	6	14.29	0.74 ± 0.073	/
$H_7 \rightarrow G_7$	15.42	1.84	6	0	/	0.47 ± 0.13
$G_7 \rightarrow H_8$	15.37	0.34	7	-16.67	0.87 ± 0.040	/
$H_8 \rightarrow G_8$	15.41	-0.26	6	14.29	/	0.13 ± 0.090
$G_8 \rightarrow H_9$	13.89	9.87	5	16.67	1.005 ± 0.037	/
$H_9 \rightarrow G_9$	14.10	-1.54	5	0	/	0.13 ± 0.12
AVG	**16.27**	**5.52**	**6.58**	**2.33**	$\mathbf{0.88 \pm 0.085}$	$\mathbf{0.34 \pm 0.22}$
$\mathbf{H_0 \rightarrow G_9}$	**>292.90**	**41.90**	**/**	**44.44**	**/**	**/**

repetitive flight test #13 by performing path integration from frame-to-frame OF measurements, and without the use of any snapshots).

In Tables III and IV, we observe that dynamic path optimisation occurs when travelling multiple times back and forth between the home and goal locations, at an average rate of about 2% to 4% between successive runs (outward or inward). This leads to a final route description that is composed of 5 to 6.33 snapshots during the last travelled journey, depending on the total length of the repetitive navigation test. In our experiments, the home and goal locations are separated by a 3D distance of about 12m. Because the flight tests were conducted at a height above ground of about 5m, the CAs have a radius equal to approximately 1m. Thus home and goal CAs are separated by about 10m (border to border). The minimal expected size of the visual route, describing a homing run and constituted of tangential CAs (with a 2m diameter), is about 5 snapshots. This minimal route description is indeed observed in the longest flight experiment, as seen in Table IV, indicating that the final route is almost a straight line.

In our approach, this learned visual route is used to fully estimate the quadrotor's state by computing its 3D position and velocity relative to the locations of the memorised snapshots. Although it successfully demonstrates route-based control and guidance of a rotorcraft UAS in outdoor environments, navigation remains constrained to the 3D space that is described by the learned visual route and the associated CAs. This spatial constraint makes our strategy sensitive to strong wind gusts that can carry the UAS away from its operating range. In that case, the visual estimates obtained through the OF-SM method rapidly become noisy, compromising the computation of velicity from the position estimates.

One corrective behaviour would be to temporarily stop the following of the last memorised route, and to rather estimate the rotorcraft's state by refering to the snapshot waypoints currently being learnt (as route learning runs concurrently with route following). In this case, once the perturbation stops, the rotorcraft can return to the CA where it exited its initial route (by following, in reverse order, the recently memorised snapshots), and from there it can resume its long-range homing process. An alternative solution would be to use frame-to-frame OF to estimate the UAS' state while it is outside the current CA, and to bring it back to the CA. However, both methods require the ability to detect the moment when the rotorcraft leaves the current CA. This detection can be easily achieved, when estimating the quadrotor's egomotion with the OF-SM method by monitoring the reprojection error, which would rapidly increase in this case. Indeed, this residual 'fitting' error in finding the best 3D translation and rotation combination that matches the OF measurements grows rapidly when the rotorcraft leaves the CA.

VII. CONCLUSION

We present in this paper a novel, mapless navigation strategy based on a snapshot matching technique, which is applied to the real-time control and guidance of rotorcraft UAS in unknown outdoor 3D environments. This approach relies on learning a minimal visual route, constituted of a sequence of sparse snapshot-vector duples, which does not rely on path integration. The rotorcraft's state (3D position and velocity) is fully estimated relative to the locations of the snapshot waypoints, without requiring any additional sensor data fusion, and navigation back to a previsited location is achieved by successively reaching each of the route waypoints. In addition to using a sparse route description, our navigation strategy demonstrates dynamic path optimisation, producing significantly reduced homing paths when travelling back and forth between two familiar locations.

The view-based strategies that we have used for the learning, following and optimisation of navigational routes are very similar to those that have been postulated for insect navigation [3]. While our study cannot prove that insects use

these strategies for navigation, it demonstrates that they are certainly viable.

Future work will aim at making our navigation scheme more robust by implementing a corrective solution for situations when the rotorcraft exits the operating range (catchment areas) of the learned route. Further flight experiments will also enable the testing of our strategy in cases where the size of the catchment areas varies significantly, due to altitude variations.

REFERENCES

[1] B. A. Cartwright and T. S. Collett, "Landmark Learning in Bees," *Journal of Comparative Physiology*, vol. 151, no. 4, pp. 521–543, 1983.

[2] F. Dyer, "Spatial memory and navigation by honeybees on the scale of the foraging range," *The Journal of Experimental Biology*, vol. 199, no. 1, pp. 147–154, 1996.

[3] M. Collett, D. Harland, and T. S. Collett, "The use of landmarks and panoramic context in the performance of local vectors by navigating honeybees," *Journal of Experimental Biology*, vol. 205, no. 6, pp. 807–814, 2002.

[4] S. Weiss, M. W. Achtelik, S. Lynen, M. C. Achtelik, L. Kneip, M. Chli, and R. Siegwart, "Monocular Vision for Long-term Micro Aerial Vehicle State Estimation: A Compendium," *Journal of Field Robotics*, vol. 30, no. 5, pp. 803–831, Sep. 2013.

[5] J. Engel, J. Sturm, and D. Cremers, "Accurate Figure Flying with a Quadrocopter Using Onboard Visual and Inertial Sensing," *IMU*, vol. 320, p. 240, 2012.

[6] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast Semi-Direct Monocular Visual Odometry," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 15–22.

[7] R. Menzel, U. Greggers, A. Smith, S. Berger, R. Brandt, S. Brunke, G. Bundrock, S. Hülse, T. Plümpe, F. Schaupp, and others, "Honey bees navigate according to a map-like spatial memory," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 8, pp. 3040–3045, 2005.

[8] H. Romero, S. Salazar, and R. Lozano, "Real-Time Stabilization of an Eight-Rotor UAV Using Optical Flow," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 809–817, Aug. 2009.

[9] F. Kendoul, I. Fantoni, and K. Nonami, "Optic flow-based vision system for autonomous 3D localization and control of small aerial vehicles," *Robotics and Autonomous Systems*, vol. 57, no. 6-7, pp. 591–602, Jun. 2009.

[10] R. Strydom, S. Thurrowgood, and M. Srinivasan, "Visual Odometry: Autonomous UAV Navigation using Optic Flow and Stereo," in *Australasian Conf. on Robotics and Automation (ACRA)(Melbourne)*, 2014.

[11] J.-C. Zufferey and D. Floreano, "Fly-Inspired Visual Steering of an Ultralight Indoor Aircraft," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 137–146, Feb. 2006.

[12] P. Li, M. Garratt, A. Lambert, M. Pickering, and J. Mitchell, "Onboard Hover Control of a Quadrotor using Template Matching and Optic Flow," in *World Congress in Image Processing*, 2013.

[13] S. Ahrens, D. Levine, G. Andrews, and J. 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*. IEEE, 2009, pp. 2643–2648.

[14] P. Li, M. Garratt, and A. 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*, Mar. 2015.

[15] A. Denuelle, R. Strydom, and M. V. Srinivasan, "Snapshot-based Control of UAS Hover in Outdoor Environments," in *Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on*. IEEE, Dec. 2015, pp. 1278–1284.

[16] A. Denuelle and M. V. Srinivasan, "Snapshot-based Navigation for the Guidance of UAS," in *Australasian Conf. on Robotics and Automation (ACRA)(Canberra)*, 2015.

[17] L. Dittmar, W. Stürzl, S. Jetzschke, M. Mertes, and N. Boeddeker, "Out of the box: how bees orient in an ambiguous environment," *Animal Behaviour*, vol. 89, pp. 13–21, Mar. 2014.

[18] B. Baddeley, P. Graham, A. Philippides, and P. Husbands, "Holistic visual encoding of ant-like routes: Navigation without waypoints," *Adaptive Behavior*, vol. 19, no. 1, pp. 3–15, Jan. 2011.

[19] A. Denuelle and M. V. Srinivasan, "Bio-inspired Visual Guidance: from Insect Homing to UAS Navigation," in *Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on*. IEEE, Dec. 2015, pp. 326–332.

[20] Y. Matsumoto, M. Inaba, and H. Inoue, "Visual Navigation using View-Sequenced Route Representation," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 1. IEEE, 1996, pp. 83–88.

[21] S. D. Jones, C. Andresen, and J. 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*, vol. 2. IEEE, 1997, pp. 551–557.

[22] A. Vardy, "Long-Range Visual Homing," in *IEEE International Conference on Robotics and Biomimetics, 2006. ROBIO'06*, Dec. 2006, pp. 220–226.

[23] L. Smith, A. Philippides, P. Graham, B. Baddeley, and P. Husbands, "Linked Local Navigation for Visual Route Guidance," *Adaptive Behavior*, vol. 15, no. 3, pp. 257–271, Jan. 2007.

[24] F. Labrosse, "Short and long-range visual navigation using warped panoramic images," *Robotics and Autonomous Systems*, vol. 55, no. 9, pp. 675–684, Sep. 2007.

[25] C. Zhou, Y. Wei, and T. Tan, "Mobile Robot Self-Localization Based on Global Visual Appearance Features," in *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 1. IEEE, 2003, pp. 1271–1276.

[26] A. A. Argyros, K. E. Bekris, S. C. Orphanoudakis, and L. E. Kavraki, "Robot Homing by Exploiting Panoramic Vision," *Autonomous Robots*, vol. 19, no. 1, pp. 7–25, 2005.

[27] 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*, Oct. 2009, pp. 5315–5320.

[28] Y. Fu and T.-R. Hsiang, "A fast robot homing approach using sparse image waypoints," *Image and Vision Computing*, vol. 30, no. 2, pp. 109–121, Feb. 2012.

[29] L. Smith, A. Philippides, and P. Husbands, "Navigation in Large-Scale Environments Using an Augmented Model of Visual Homing," in *From Animals to Animats 9*, ser. Lecture Notes in Computer Science, S. Nolfi, G. Baldassarre, R. Calabretta, J. C. T. Hallam, D. Marocco, J.-A. Meyer, O. Miglino, and D. Parisi, Eds. Springer Berlin Heidelberg, Jan. 2006, no. 4095, pp. 251–262.

[30] Z. Chen and S. T. Birchfield, "Qualitative Vision-Based Mobile Robot Navigation," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*. IEEE, 2006, pp. 2686–2692.

[31] T. Ohno, A. Ohya, and S. I. 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*, vol. 2. IEEE, 1996, pp. 672–679.

[32] C. Sagüés and J. Guerrero, "Visual correction for mobile robot homing," *Robotics and Autonomous Systems*, vol. 50, no. 1, pp. 41–49, Jan. 2005.

[33] S. Thurrowgood, R. J. D. Moore, D. Soccol, M. Knight, and M. 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*, vol. 31, no. 4, pp. 699–727, Jul. 2014.

[34] A. Denuelle, S. Thurrowgood, F. Kendoul, and M. V. Srinivasan, "A view-based method for local homing of unmanned rotorcraft," in *Automation, Robotics and Applications (ICARA), 2015 6th International Conference on*. IEEE, 2015, pp. 443–449.

[35] Swift Navigation, "Piksi GPS Receiver." [Online]. Available: http://docs.swiftnav.com/pdfs/piksi_datasheet_v2.3.1.pdf

[36] R. C. Nelson and J. Aloimonos, "Finding Motion Parameters from Spherical Motion Fields (Or the Advantages of Having Eyes in the Back of Your Head)," *Biological Cybernetics*, vol. 58, no. 4, pp. 261–273, 1988.

[37] MikroKopter, "FlightCtrl V2.1." [Online]. Available: http://wiki.mikrokopter.de/en/FlightCtrl_ME_2_1