THE UNIVERSITY *of* EDINBURGH

Using a quadcopter to model the visual navigation behaviours of flying insects

Jan Stankiewicz

Doctor of Philosophy

Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh

2021

Lay Summary

Many animals are capable of reliably navigating between sites of importance in their local habitats. Certain species of bee and wasp are particularly adept at this task; they have mastered the art of exploring large territories and transporting resources efficiently back to their nests. The precise details of how this is achieved are not yet fully understood, but vision is known to play a significant role. Roboticists are also interested in navigation by vision, but contemporary algorithms consume a lot of computational resource, and do not scale well to large environments, especially those in the natural world. How then does a bee achieve its navigational goals? A particularly inspirational accomplishment when its size (less than 2 grams), visual acuity ($1/60^{\text{th}}$ of our own), and brain volume (less than 1 mm^3) are considered.

A classical behavioural experiment is to displace a honey bee from a feeding site to a new location. When this procedure is conducted the bee will invariably travel in the direction that its nest would be in if the bee was still at the feeder. However, if the bee is displaced to a previously visited location within approximately 500m of its nest, it is usually able to fly directly to this nest, regardless of its position prior to the displacement. These experiments indicate that there are at least two separate navigation mechanisms at play. A long-range, relative position tracking system, and a shorter-range place recognition based visual homing system. In this thesis, existing models of terrestrial insect navigation behaviours are adapted and deployed on an aerial agent.

Model building is an established means of testing a hypothesis that relates to an animal's behaviour. If a model correctly predicts a given animal's behaviour when that animal is confronted with a particular stimulus, the model remains a viable hypothesis. If the model fails to achieve this, it should be either rejected or modified. Biorobots (robotic agents that are used to embody animal behavioural models), are regarded as an exemplary form of modelling, because they operate in the same arena as the animal; the physical world.

In Chapter 3, the use of a quadcopter as a biorobot for investigating insect navigation behaviours is proposed. The ability to move a camera freely around in three-dimensional space is seen as the primary requirement to model vision-based navigation behaviours in flying insects. This requirement can be met with a quadcopter, despite its larger size and dissimilar propulsion system. A platform is developed with all the necessary resources to deploy behavioural models, and the instrumentation to measure

its true position. Hardware plans and the software required to recreate the platform are shared.

One navigation strategy that is used by roboticists and observed in animals is path integration. In this process, the agent maintains an estimate of the distance and direction to a reference location by tracking its self-motion since that site was previously visited. A continual estimate of both speed and direction is generally required to achieve path integration. In Chapter 4, biologically relevant means of estimating speed are investigated, forming a companion with existing work concerning the methods by which insects track their heading (a digital compass is used for heading information in this study). As an animal moves through space, the scene passes over their retina. It is known that the rate of this motion can be estimated in neural circuitry. However, this rate is dependent on the distance to the surface (if you move your head from side to side, nearby objects will move across your visual field at a greater rate than distant objects). In this chapter, a path integration circuit is trialled with and without compensation of scene depth. The depth is measured using an aircraft height sensor. When height information is incorporated into the agent's speed estimate, it is able to return to within a few meters of the starting location. When the height information is not included, the agent's homing performance is dramatically reduced, indicating that flying insects could benefit from measuring this value. The path integration model is then tested in different environments and with different view configurations. The model is found to work best in scenes with low variation in terrain height, and where there is a lot of variety in the visual information (as is common in nature). In most environments, the model works best when the camera is looking towards the ground because this provides a relatively constant depth and has more visual variety than other regions (e.g. the sky).

A problem with path integration is that small errors in motion estimation tend to add up over the duration of a journey and can become large in the context of foraging behaviour. Visual navigation is a process that is known to be useful in counteracting this and has been well studied in desert ants. Ants returning to their nest after a foraging journey are thought to memorise views along their return path. On subsequent journeys along the same route, the ability to compare their currently experienced view with memorised views is thought to provide desert ants with knowledge of whether they have previously passed through a particular location. This process has been successfully modelled with ground-based biorobots. Furthermore, it has been shown that this information can be used to navigate along previously traversed routes.

In Chapter 5, the principles of the desert ant visual navigation models were applied

to the quadcopter biorobot developed in Chapter 3. A continuous line of view memories along the ground form a route that the aerial agent can follow. Think of Hansel and Gretel's breadcrumb trail. A process of zig-zagging over this trail so that contact is maintained with the previously traversed route is shown to be a reliable navigation method. This is even the case in terrain with apparently few distinctive visual landmarks such as an empty field.

A desirable property of each view memory is that it can be recognised by the biorobot, even when the biorobot is displaced from the actual site where the memory was formed. A biologically plausible method of processing raw images is shown to double the "catchment area" of each reference image, and thereby increase the reliability of the visual navigation procedure for a given number of reference images. This processing method involves the detection of strong edges in a view, and the orientation of each edge. The constellation of these features provides a reliable signature of the scene, which persists in different lighting conditions. This model works particularly well with low-resolution images.

Together the findings in Chapter 4 and Chapter 5 provide an updated hypothesis on how flying insects navigate to their nest with visual navigation. The principles could also be adopted by roboticists interested in building an aerial agent that can explore an environment or navigate from point to point with limited computational resources. Unlike many contemporary solutions this method operates robustly in natural scenes.

Abstract

Micro aerial vehicles (MAVs) have become increasingly prominent in the last decade, with several sectors now routinely using this technology for applications such as filming, surveying and maintenance. A significant barrier towards further MAV technology adoption is the absence of reliable, lightweight autonomous navigation systems that can robustly operate in areas where global navigation satellite systems (GNSS) signals are not reliable.

Flying insects are an order of magnitude smaller than MAVs and they can navigate between several sites of interest in large local neighbourhoods that span several kilometres. Fed by low resolution eyes and using neural processing circuits, the power consumption of an insect's brain is several orders of magnitude lower than state-of-the-art robotic visual navigation systems. This formidable capability has inspired ethologists, neuroscientists and engineers to engage in a process of reverse engineering the key mechanisms involved in local insect navigation behaviours, with the ultimate goal of describing the complete underlying neural circuitry.

In this thesis, recent advances in MAV technology are exploited as a means of evaluating candidate behavioural models that have only been deployed in simulation environments or on terrestrial robotic platforms. The hardware and software development of an aerial biorobot that is configured to test insect navigation models is described. This system features a quadcopter airframe, Pixhawk flight controller and selected interfacing ancillary avionics. The resultant platform has sufficient onboard processing power to flexibly deploy path integration and visual homing behavioural models. The biorobot also features an active mechanical view stabilisation system.

The biorobot is first used to embody a recently proposed anatomically constrained path integration circuit. To this end, a biologically plausible matched filter visual odometry pipeline is implemented. The viewing direction, resolution and field of view of the visual input to this circuit is systematically altered and tested in a variety of natural scenes. This process enables the prescription of an optimal visual sensor configuration on the basis of empirical evidence. When the biorobot is deployed in a relatively flat environment with the optimal view configuration, a homing error drift rate of 1.5m per 100m is estimated.

The biorobot subsequently supports an investigation into whether flying insects could use visual route following to overcome the drift issues associated with path integration. A robust procedure is developed and evaluated. It is found to be effective across distances of at least 30m, even in seemingly featureless environments such as

empty arable fields. It is known that orientated bandpass filters exist in the early stages of the human vision system. Using a complex wavelet structural similarity algorithm, the orientated bandpass filter approach is adapted to a visual homing framework. This configuration is shown to double the catchment area and increase the discriminability of the snapshot model for view matching in natural scenes when it is compared to existing view matching techniques that operate in the spatial domain.

Acknowledgements

First and foremost I thank Barbara Webb for her supervision and guidance which extends to all areas of this manuscript. I thank you for having the vision to establish the *Insect Robotics Group* and for allowing me to be part of this journey. Together with Yvan Petilot, you have helped me navigate the trickier phases of the PhD process and helped me to negotiate the inevitable patches of turbulence along the way.

I would like to thank Colin Wilson, Douglas Howie and David Hamilton of the informatics workshop for your guidance on designing the aluminium and 3D printed parts that are detailed in chapter 3. Moreover, I am grateful for your skill at manufacturing these parts. This effort was a prerequisite to the majority of results presented in this thesis.

Thank you to the biologists and neuroscientists who have taken the time to reply to my emails over the years. Particular acknowledgements to: Andrea Adden and Stanley Heinze for your perspectives from the sharp end of the insect brain; Mike Mangan and Jacqueline Degen for your insights from the field; Wolfgang Stürzl and Jochen Zeil for sharing your learning flight dataset and ensuing discussions relating to this.

I am indebted to the communities behind the multiple open-source software packages on which I have relied upon. In particular a massive thank you to the PX4 and ROS communities. Both of these projects have saved me decades of development time and the outcome of this thesis would not have been possible without the resources that you provide.

Thank you to friends and family for your support in the last four years. In particular, thank you to Steven Anderson and Denis Skelly for your invaluable, eagle-eyed proofreading efforts. Finally, a special thank you to Amelia, for your tremendous patience in recent years and for allowing space for postgraduate research in our relationship.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Jan Stankiewicz*)

Contents

Acronyms

AOTU	Anterior optical tubercle
API	Application programming interface
CAD	Computer aided design
COTS	Customisable off-the-shelf
CPU	Central processing unit
CWSSIM	Complex wavelet structural similarity
DC	Direct current
DOF	Degree of freedom
EMC	Electromagnetic compatibility
EMD	Elementary motion detector
ESC	Electronic speed controller
FC	Familiarity count
FCU	Flight control unit
GNSS	Global navigation satellite systems
IMU	Inertial measurement unit
LED	Light emitting diode
LPTC	Lobula plate tangential cells
LTPC	Local tangent plane coordinate
MAV	Micro aerial vehicle

MOD	Median outlier detection
MSE	Mean squared error
NRMSE	Normalised root mean squared error
RC	Remote control
RR	Recognition rate
RTK	Real-time kinematic
SDF	Simulation description format
SLAM	Simultaneous localisation and mapping
SSIM	Structural similarity
TORF	Transverse oscillating route following
UAV	Unmanned aerial vehicle
UDP	User datagram protocol
VTOL	Vertical takeoff and landing

Chapter 1

Introduction

1.1 Motivation

1.1.1 Navigation in autonomous micro aerial vehicles (MAVs)

In the last decade, commercial unmanned aerial vehicle (UAV) technology has seen a rapid increase in financial investment and product capability. Improvements in battery energy density, brushless motor control, low power computing technology and flight stability algorithms have facilitated the miniaturisation of UAVs with useful payload and flight time attributes. These smaller aircraft, micro aerial vehicles (MAVs)[1], being smaller, cheaper and safer than larger UAV systems have already found substantial application in filming, mapping and surveying sectors where the MAV can be human operated or supervised.

The autonomous navigation systems deployed on modern commercial MAVs are reliant on global navigation satellite systems (GNSS). As outlined in (Loianno et al., 2018), there are many potential applications of MAVs in low altitude, cluttered environments where GNSS signals are not reliable. Recent research efforts to overcome this problem have been dominated by visual navigation and in particular simultaneous localisation and mapping (SLAM) algorithms. One issue with this approach is that considerable computational resource is required to map the world or even localise against an existing visual map. Therefore additional onboard computing power is required which in turn leads to an increase in the platform weight and power consumption, thus counteracting the power system gains that have driven the MAV revolution. Another

[1]The definition of an MAV is somewhat ambiguous in the literature, a working definition here is any rotorcraft UAV weighing less than 5Kg.

issue is that SLAM is inherently a probabilistic solution, with many possible failure modes (Alsayed et al., 2017; Cadena et al., 2016; Williams et al., 2007). It is therefore unlikely that a navigation system relying exclusively on a SLAM navigation system would be certified for use in safety critical applications such as an autonomous MAV flight in inhabited spaces, without redundant backup systems in place. Therefore, there is a significant research opportunity for the development of parsimonious navigation systems for autonomous MAVs (Chahl, 2015).

1.1.2 The capabilities of habitually flying central place foraging insects

As a counterpoint to the state of affairs in MAV technology, nature has crafted a wide array of flying animals that can navigate several kilometers between key locations. Certain species within the class, Insect, have evolved to routinely navigate between places of interest with highly constrained information processing resources. Bumble bees for example, are known to remember the location of multiple feeding sites and can flexibly navigate between them on a single foraging trip (Woodgate et al., 2017). Honey bees that are displaced from their nest are able to navigate home from distances of up to 13km (Pahl et al., 2011). Solitary ground nesting wasps are able to return to their inconspicuous burrows following hunting bouts (Tinbergen, 1972). Animals that display this behaviour of returning to a fixed nesting point are known as *central place foragers*[2]. Given the difficulties faced by aerospace engineers, a natural question to ask is; *how do flying insects reliably achieve point-to-point navigation?*.

Visual cues are known to play a key role in the navigational abilities of foraging insects that navigate on the wing. Bees and wasps are known to conduct exploratory flights on the first departure of their nests. During this process they are thought to learn some kind of visual representation that relates to their respective habitats and acts as a reference on future journeys (Stürzl et al., 2016). In contrast with its experienced sisters, an animal that is denied this experience will fail to return to its nest when it is artificially displaced from it (Degen et al., 2016). Thus there is strong evidence that some form of *visual homing* mechanism exists. It has also been apparent for some time that these insects maintain an online estimate of the distance and direction back to their nests by path integration. Quite literally, some insects can add up or integrate their

[2]Throughout this thesis, the word 'forage' will refer to central place foraging behaviour unless otherwise specified

path, or motion over time. Vision is also thought to play a key role in the process of egomotion estimation in flying insects.

1.1.3 Biorobots as a neurethology modelling tool

In order to benefit from the strategies employed by flying insects, knowledge about the mechanisms that produce this behaviour is essential. *Neuroethology* is a field that seeks to completely describe the sensorimotor systems that govern specific animal behaviours, down to the neural level. This quest is often proceeded by a description of behaviours at a functional or algorithmic level. One approach towards this endeavour is to develop models that account for an animal's behaviour when it is placed in a certain scenario. An appropriately defined model can be systematically evaluated in order to reject incorrect hypotheses and generate new predictions.

Robots built with analogous sensorimotor systems to an animal are a category of *biorobot*[3]. This kind of biorobot has proved to be a useful model / model testing platform on several occasions (Webb, 2002b). An advantage of this modelling approach is that being of the same world, a biorobot is subject to the same physical constraints and noisy environmental signals as its analogue. By contrast, simulations have a tendency to ignore or grossly simplify the physical properties of the natural world. However, a biorobot is not automatically a good candidate for an ethological model, careful consideration must be invested when translating a theoretical process into a physical model. Note that the goal is usually to compare the behaviour of the robot with that of the animal, rather than to maximise the robot's proficiency at conducting the task. A framework that can be used to guide the attributes of a biorobot and associated test paradigms is presented in (Webb, 2002a).

1.1.4 Biorobots and the mechanisms of navigation in central place foraging flying insects.

Biorobots have already played a role in the study of insect navigational behaviours. Developing robots that can fly through tunnels like those used by ethologists studying bee flight control has resulted in regular bi-directional knowledge transfer between roboticists and ethologists (Serres and Ruffier, 2017). Terrestrial robots such as Sahabot (Lambrinos et al., 1999) have played a role in modelling and testing closed loop path

[3] In this thesis the term *biorobot* refers to a robot that has been built in order to test an ethological hypothesis unless otherwise specified.

integration and visual homing behaviours in the environment of the desert ant. However, in the last decade, some significant modelling advances have been made in this field that were not evaluated on Sahabot:

1. Routes can be learned using a chain of visual memories (*visual route following*) (Baddeley et al., 2012). This strategy has been deployed on a biologically plausible neural network model that is inspired by the structure of insect brain *mushroom bodies* (Ardin et al., 2016).

2. A biologically plausible neural circuit for path integration has been developed (Stone et al., 2017).

To date, few publications have described the evaluation of these concepts with robotic platforms and/or in the natural world. The path integration circuit in (Stone et al., 2017) was informed by the neurophysiology of a flying insect but it has only been tested in simulation or in a laboratory on a ground based robot. By contrast, flying insects rely on this system to navigate large distances over uneven terrain. Visual homing procedures have also been tested on the same robotic platform (Kodzhabashev and Mangan, 2015); while this robot was tested in a natural outdoor environment, trials were limited to 8m routes consistent with ant behaviour experiments. Furthermore, all visual route following implementations have suffered from reliability issues associated with route divergence due to local minima. More recently a terrestrial robot was tested over distances of 10m (Knight et al., 2019).

1.2 Outlook

1.2.1 Research questions

The work in this thesis lies at the confluence of the areas discussed in section 1.1. The objective is to gain greater understanding of the mechanisms governing visual navigation in flying insects with the hope that engineers will ultimately be able to make use of this knowledge in order to develop more efficient visual navigation configurations. It is also hoped that the MAV platform developed herein will give those with similar ambitions a useful starting point. In order to guide this work towards biological relevance, the following questions were asked at the outset:

1. To what extent can flying insects rely on a path integration strategy when navigating in real habitats?

2. How do flying insects bridge the gap between the accuracy limits of their path integration system and the need to locate their potentially inconspicuous nests?

1.2.2 Contributions

The main contributions from each chapter are summarised in the following section:

1.2.3 Chapter 3: An aerial platform for testing insect inspired navigation algorithms

- The suitability of a quadcopter MAV as a biorobotic platform for visual navigation model embodiment is reviewed.
- A robust platform for deploying insect behaviours on an autonomous MAV is developed.

1.2.4 Chapter 4: Deploying the central complex path integration circuit on an MAV

- A view adaptable, biologically plausible, matched filter optic flow pipeline for sensing egomotion is developed.
- The performance of the central complex model introduced in (Stone et al., 2017) is tested for the first time on a holonomic platform over large scale distances in the natural world.
- A mathematical model for the anticipated error of such systems in environments with differing topographical construction is developed. This offers a prediction for the required range of visual navigation systems during foraging behaviours.
- The impact of camera view direction and resolution on odometry performance are quantified.
- The benefits of using a mechanically stabilised view to mitigate optic flow induced by quadcopter body rotations are investigated.
- The benefits of using height compensation in the odometry circuit are investigated and shown to be important for an optic flow based odometer which operates at variable height/speed ratios.

1.2.5 Chapter 5: Complex wavelet structural similarity for view based image matching

- Complex wavelets and structural similarity are introduced as a biologically plausible shift invariant enhancement to the image difference models currently used in the biorobotics community.
- Ground facing views are shown to be more salient than a horizontal views for the task of locating a goal in the XY plane.
- A new, robust, biologically plausible route following procedure is presented and evaluated on the robotic platform presented in Chapter 3. This makes use of the fact that flying insects can perform view comparison gradient descent by translating sideways.
- The assumption that flat mown fields are "featureless" is challenged, instead we find that unique descriptors can be found with biologically plausible resolutions in this type of environment. The complex wavelet descriptors generated from multiple views captured in a local area are unique and have large hamming distances between each other.

Chapter 2

Background

In this chapter, background concepts relating to the thesis are presented. A general review of the capabilities and the relevant anatomy of a subset of flying insects is presented in sections 2.1 to 2.3. The sensory systems employed by these insects to detect their ego-motion and global heading state are reviewed in section 2.4. This permits a detailed review in section 2.5 of the behavioural evidence for local navigation behaviours, namely path integration and visual navigation. The contemporary view on the mechanisms governing these behaviours is also reviewed. Finally, flying insect biorobots which have a focus on navigation behaviours are reviewed in section 2.6. Finally, relevant state-of-the-art biorobots are reviewed in section 2.6

2.1 An overview of flying insect local navigation

Navigation, can be defined as the process of an agent accurately finding its way to a target location in space. *Hymenoptera* is an order of insect containing many species that are credited with the ability to navigate in a local[1] neighbourhood. Over 100 million years ago, the ancestors of these species evolved a *central place foraging* lifestyle which is characterised by a behavioural pattern of locating food sources in their habitat and returning to fixed nesting sites with food for their offspring (Peters et al., 2017). There are many extant species of bee and wasp that navigate distances of several kilometers on the wing for this purpose. This demonstrates a capacity to navigate well beyond the range of reflexive *taxis* mechanisms such as odour plume following (Olberg, 1983) or visual beaconing (Collett, 2010).

[1]The term *"local"* is used to distinguish between the ability to return to a location of importance (with journeys measured in minutes or hours) and long range migratory behaviour (with journeys measured in

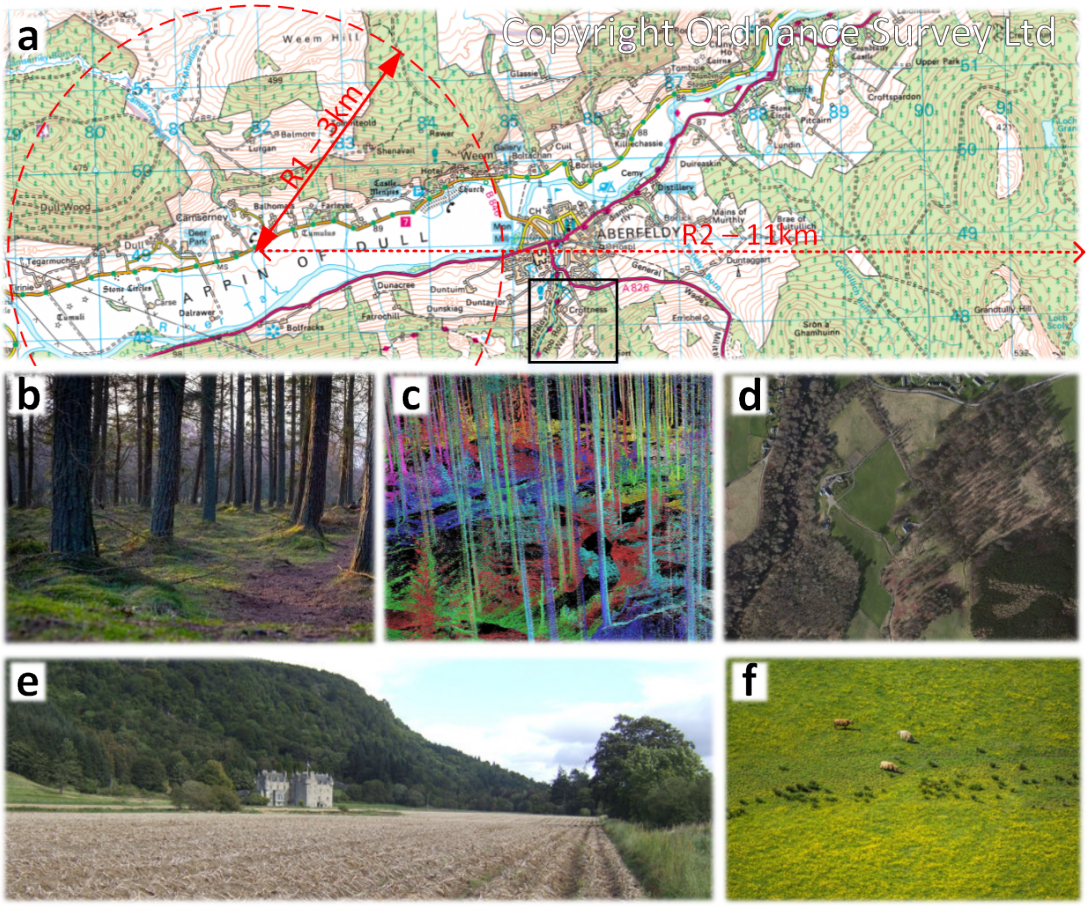


Figure 2.1: Navigating a landscape. **a**) Topographical map indicating the typical boundaries of the foraging range of honey bees (radius r1, making red dashed circle at 3km) and maximum trained return trip of 11km (r2), Contains OS data © Crown copyright and database rights 2020. **b**) View of forest section just above ground level **c**) Point cloud of a Boreal forest generated in (Kukko et al., 2017) licensed under CC BY 4.0**d**) Satellite image of the area denoted by a black rectangle in panel **a**, generated with Bing™ maps, reprinted with permission from Microsoft Corporation. **e**) Terrestrial view that contains a prominent landmark, castle Menzies. **f**) Aerial view of a meadow with few reliable visual features.

The ordinance and survey map presented in fig. 2.1a illustrates the scale of the navigation task space confronting flying central place forager insects, with individuals routinely flying up to 3km from their nest over varied terrain. The concept of a *map*, which offers a simplified representation of the world, is integral to many robotic navigation systems. When a map is involved, the navigation task can be divided into at least two sub-functions; localisation and path planning. *Localisation* refers to the task of self orientation with respect to the map or some other origin (*where am I?*) and *path planning* involves generating a schedule to arrive at the goal (*how do I get to where I want to be?*).

Modern robot visual navigation systems often make use of dense self-generated maps that detail the 3D structure of their environment. fig. 2.1c depicts a point cloud of a small area of forest featuring hundreds of thousands of data points. When a robot is in this location it can use its current sensor state to infer its most likely location in the map. While this mode of sensing can provide robust localisation in the real world, the amount of data required to describe even a single 3km foraging route is not something that can practicably be deployed on modern MAV platforms.

The mechanisms governing local insect navigation are not yet fully specified but given the size of insect brains and the behavioural evidence to date, it is reasonable to assume that dense 3D maps do not feature in their method. A recent review postulates that three mechanisms could account for the bulk of behavioural observations relating to these animals (Webb, 2019). The three mechanisms are; *path integration*, *vector memory* and *view memory*. Path integration (known as *"dead reckoning"* in robotics) is the process of integrating all motion travelled since departing from a known location. It can be used to maintain a constant estimate of which way and how far to go to get back to a reference location. Furthermore, this capability raises the possibility of a vector memory, whereby the distance and direction of places of interest relative to an origin can be flexibly used to generate novel routes between these sites. Note that this polar coordinate system is consistent with recent neural models which are based on neurophysiologicl data Stone et al. (2017); Webb (2019). View memory can be used to associate familiar views with actions to get to the desired location. Here *visual navigation* will be used as an umbrella term for view memory based behaviours.

For both robotic and insect visual navigation methods, the image sensor's view direction and view properties can have a considerable bearing on the task complexity. Compare the terrestrial view in fig. 2.1b with the aerial view fig. 2.1d which are both

days or weeks).

captured in the same geographic coordinates (latitude and longitude). The ground view is cluttered; at the level of an image pixel, small motions in the scene cause large deviations in view content. As a consequence, view matching in this kind of environment requires many views to be stored. By contrast, in the aerial perspective, the structure of the scene remains stable, even in the presence of large motions. There are also features in the landscape that can act as route guide, the river for example. In other scenarios, a terrestrial view in open ground may feature prominent *landmarks*, such as the castle in fig. 2.1e, which can be used to navigate over a long range. This type of landmark is especially salient when it is proximal to the destination, or when it is along the route such that it can act as a *waypoint*. Depending on altitude and topography, aerial views can be lacking in useful navigation. To a human eye, the most prominent landmarks in fig. 2.1f are the cattle which are clearly not a stable landmark. Therefore, an agent's environment and perspective of the environment influence the complexity of the navigation task.

2.2 Species-of-interest

In this section the species-of-interest are defined. Each of these model animals (shown in fig. 2.2) has a central place foraging lifestyle and has been extensively studied for its local navigation ability.

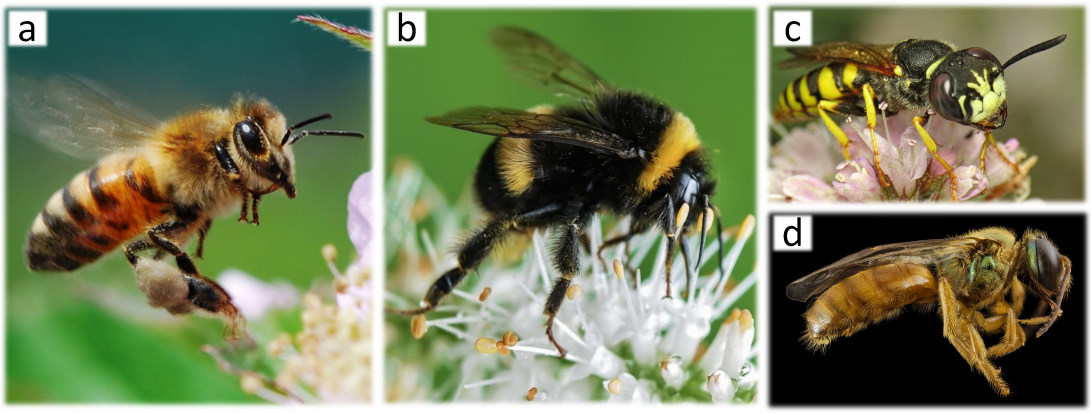


Figure 2.2: Montage of the species-of-interest: **a**) honey bee; John Baker, CC BY 2.0, **b**) bumble bee **c**) Philanthus triangulum. Martin Cooper, CC BY 2.0. **d**) Megalopta genalis

2.2.1 Honey bees

Honey bees are arguably the most intelligent central place foraging flying insect and are known to have the longest potential foraging range of this group, with animals being trained to feeders with a line of sight distance of up to 11km (von Frish K., 1967) and returning from displacements of up to 13km (Pahl et al., 2011). Honey bees live in diverse habitats including open "featureless" arable fields and cluttered dense forests (Hung et al., 2018). Foraging behaviour is weakly divided into different roles; scouts, which have a tendency to seek out new food supplies, and foragers, which tend to visit the same location throughout their lives (Degen et al., 2016). Honey bees can remember multiple feeder sites, the time at when these sites yield the greatest reward (Lindauer, 1960), and the most efficient way to link several sites in one forage (Buatois and Lihoreau, 2016). Scouts are able to mark the location of food sources with odour (Reinhard and Srinivasan, 2009), and even communicate the location of feeder sites to their colony through the parameters of a *waggle dance* (see section 2.2.1.1). Recent advances in the knowledge of honey bee navigation stem from *harmonic radar* tracking. This technology involves the attachment of a miniaturised transponder onto an animal such that it's location can be tracked by a ground based radar station. While these systems are not very accurate ($\pm 10m$), the 500m range of harmonic radar has been instrumental in tracking flying insect behaviours in their natural environments.

2.2.1.1 The waggle dance

When a foraging honey bee returns from a lucrative feeding site she may perform a stereotypical *dance* in her nest, the parameters of which have been shown to encode the bearing and distance to the food source (von Frish K., 1967) (see fig. 2.3). Her sisters are able to interpret this dance and thereby also visit this location in a direct fashion (von Frish K., 1967). Von Frisch concludes in (von Frish K., 1967) that honey bees do not encode metric distance of their flights because the assumed range dance parameter is not equal for feeders placed at the same distance but in different compass directions from the hive. This notion has since been corroborated by flight tunnel experiments (Srinivasan et al., 2000).

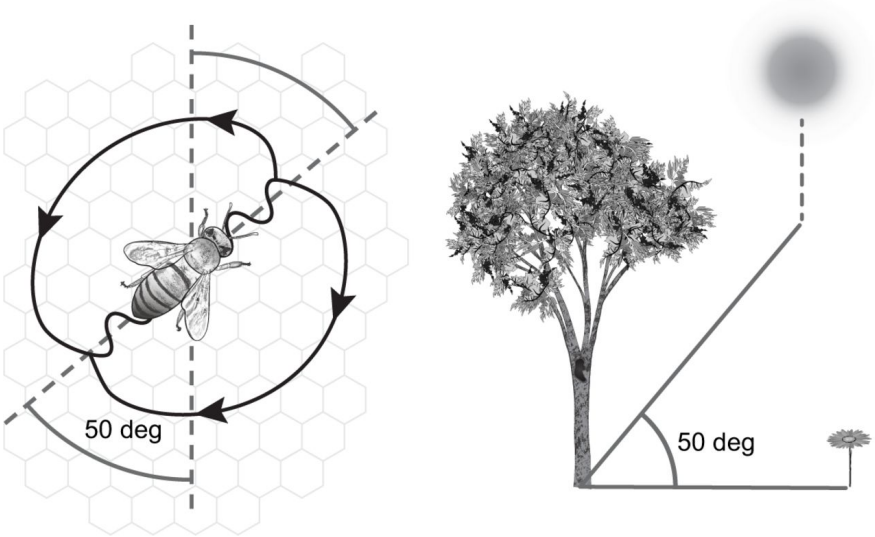


Figure 2.3: Illustration of the waggle dance, copied with permission from (Barron and Plath, 2017). Left; Schematic of the waggle dance which comprises of a straight sector in which the abdomen of the bee oscillates, followed by a looping back round to the starting point. The direction of the looping section typically alternates between clockwise and anti-clockwise motion. The direction of the food source relative to the solar azimuth is encoded by the angle of the straight sector with respect to gravity and the waggle quantity or sector length encodes the distance to the food source.

2.2.2 Bumble bees

Like their honey bee cousins, the foraging trips of bumble bees have also been studied with harmonic radar providing useful comparative datasets. They are less social than honey bees, living in colonies numbering several hundred; they are not known to be able to communicate the location of feeder sites to their sisters. Bumble bees were recorded foraging distances of up to 2.5km from their nest in (Hagen et al., 2011) which is likely to be an underestimate of their maximum foraging range. Bumble bees begin foraging after fewer learning flights than honey bees, sometimes on the first trip but more typically after 2-3 outings (Woodgate et al., 2017). It has been shown that bumble bees are likely to attend to motion parallax generated depth cues on their learning flights (Dittmar et al., 2010).

2.2.3 Solitary ground digger wasps

The utility of odour cues for navigation in social bees is poorly understood and operates over an unspecified range (Reinhard and Srinivasan, 2009). However, studies

performed by Tinbergen demonstrate that the solitary ground nesting wasp, *Philanthus triangulum* is capable of locating its ground nest without any odour cues (Tinbergen, 1972). Philanthus triangulum nests in a small hole in the ground, often surrounded by the nests of other wasps making the navigation task especially challenging. These animals are also known to have a foraging range of at least 1km (Evans, 1966).

2.2.3.1 Sweat bees

While behavioural evidence has been derived from the other species on this list, the sweat bee, *Megalopta genalis*, was the animal that provided the basis for the first anatomically plausible central complex circuit for path integration (Stone et al., 2017). Megalopta genalis is a crepuscular central place forager which lives in either solitary or weakly social nests in the canopy of tropical forests. This animal is of particular interest because: 1) the low light conditions in which it operates suggest that the polarisation patterns of the sky are the only available compass cue; 2) the forest canopy provides an overtly three-dimensional environment with a varied depth profile. These factors place stringent demands on a path integration system.

2.3 Key anatomy for visual navigation

2.3.1 Compound eyes

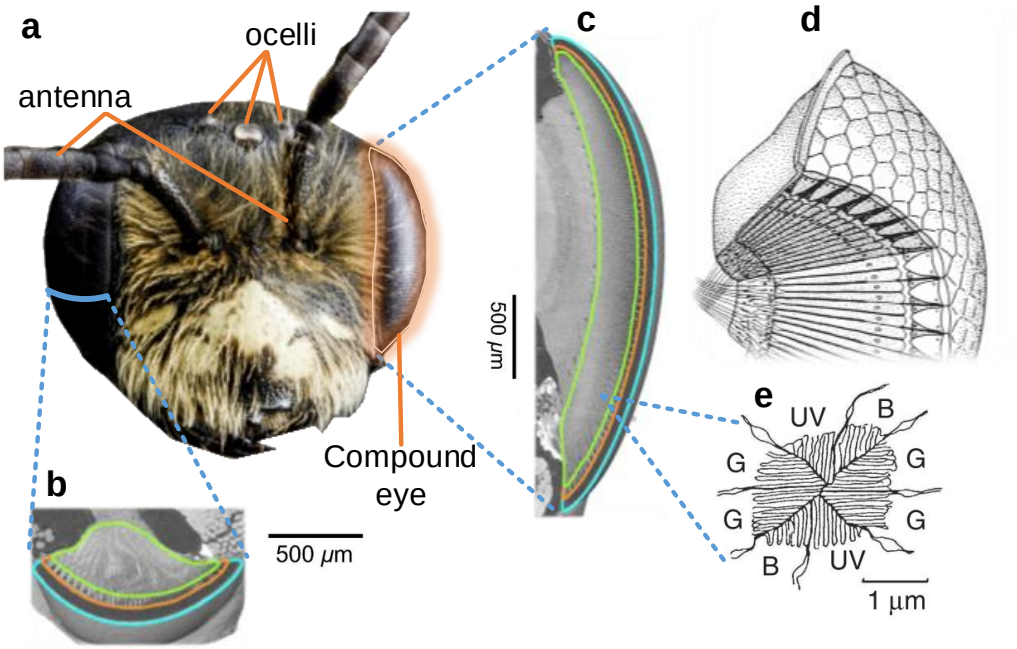


Figure 2.4: Honey bee compound eye. **a**) Annotated photograph of a honey bee head from anterior view, photograph by Andrena Cressonii. Transverse (**b**) and vertical (**c**) histological slices of a honey bee's compound eye, extracted from (Taylor et al., 2019), licensed under CC BY 4.0. **d**) Sketch of an apposition compound eye, showing the eye surface and cutaway showing the parallel structure of the eye facets. **e**) sketch showing a transverse slice through a honey bee ommatidium, there are eight photoreceptors marked according to their respective wavelength sensitivities (G; green, B; blue, UV; ultraviolet). **d**&**e** modified with permission from (Land et al., 2002).

Flying insects visualise their environment predominantly through compound eyes (shown in fig. 2.4a) which are collections of basic light detecting units known as *ommatidia*. Each *ommatidium* contains multiple photoreceptive cells that are sensitive to different light wavelengths. Wasps and bees tend to be trichromatic (Briscoe and Chittka, 2001), with motion vision pathways derived mostly from four green (535nm) rhabdoms (as marked in fig. 2.4e).

The species-of-interest all have *apposition* compound eyes which feature ommatidia that are encased with light blocking pigment. The acuity of apposition compound eyes

in a particular region can therefore be quantified by measuring the difference in viewing angle of contiguous ommatidia, usually referred to as the *interommatidial angle*. The maximum resolution of a compound eye is considerably lower than the *simple eyes* of a human. Another key difference between simple and compound eyes is that the latter usually have a panoramic aspect, sparsely sampling in virtually every direction that is not occluded by the animal's body (Taylor et al., 2019). The acuity of a compound eye is typically varied across the surface, parsimoniously matching the ecological needs of the animal. The flicker fusion frequency of flying insects in flight is known to reach 300Hz (Howard et al., 2005), indicating that their visual system is approximately six times faster than that of humans.

The species-of-interest also have three small simple eyes on their head known as *ocelli* (annotated in fig. 2.4) which offer a second means of measuring light. In comparison to signals passing through the optic lobes, signals from the ocelli have a faster, more direct pathway to motor circuits and have been implicated in body attitude control (Parsons et al., 2010), a configuration that has inspired the stability control system of a miniaturised flapping winged robot (Fuller et al., 2014).

2.3.2 Head stabilisation anatomy

The thorax and abdomen of flying insects dynamically rotate through large angles with respect to gravity during flight (Taylor et al., 2013). Compound eyes are statically linked to the head such that their viewing angle is completely dependent on the head orientation of the associated animal. Changes in head orientation would directly induce optic flow (explored later in section 2.4.1.3), and have an impact on an insect's ability to navigate using the snapshot model (section 2.4). In order to minimise disruption to these functions, flying insects are known to keep their heads stable with respect to gravity (Taylor and Krapp, 2007).

The mechanisms of head stabilisation have been particularly well documented in the wasp species, Polistes humilis. Wasps attached to a rotating platform actively rotate their heads to counteract the apparatus induced motion (Viollet and Zeil, 2013). The control response to perturbations in head roll driven by chirp signals and step inputs are related to panorama texture and lighting conditions, with the best performance observed in bright outdoor environments. In this condition, errors of up to 10° are observed in the test rig but this is not observable in video analysis of insects in free flight. It is therefore likely that a feed-forward efference copy control system is an important mechanism

for this behaviour (Viollet and Zeil, 2013). Behavioural and cellular evidence for an efference copy relating to body rotations has been observed in dragon flies (Mischiati et al., 2014) and fruit flies (Kim et al., 2015) respectively indicating that it is a widely adopted strategy.

2.3.3 Mechanoreceptive organs

Flying insects are also known to sense disturbances in their body attitude via mechanoreceptive organs. The hindwings of flies have morphed into gyroscopic sensing club-like organs known as *halteres*. It has also been demonstrated that Johnston's organs on the antenna of hawk moths can sense gyroscopic forces and are required for stable flight (Sane et al., 2007). It is not yet known if bees, which do not have halteres, can sense angular motion through mechanoreceptors. Perhaps more relevant to this work, it is known that honey bees can sense vibrations consistent with airflow (Ai et al., 2009). The antenna positioning reflex, which encodes a hawk moth's speed in free flight, is observable in honey bees at the onset of airflow (Taylor et al., 2013) but is not observable in the presence of artificially presented optical flow cues (Roy Khurana and Sane, 2016). This implies that bees can measure their *airspeed*. A model for the combination of air speed and optic flow inputs is proposed in (Taylor et al., 2013).

2.3.4 Insect brain

As shown in the schematic in fig. 2.5, insect brains are laterally symmetric about a midline. Each neuropil in the organ has a contralateral pairing, except for the central complex which is located on the midline. The basic brain structure is conserved across all species, with neuropils often enlarged when the associated function is particularly important to the survival of a species. For example, the visual lobes are large in flying insects.

Working from the lateral extremity, optic lobes process visual information in three distinct neuropils (lamina, medulla and lobula) with information flowing through in that order. These neuropils have a columnar structure, with one column per ommatidium, so information propagates through in a retinotopic fashion. The mushroom bodies, which are large in Hymenoptera, are associated with memory and learning (Webb and Wystrach, 2016), properties which are thought to be crucial in the task of visual navigation. The central complex is thought to be involved in the integration of several navigational cues (Collett, 2019; Honkanen et al., 2019) and generation of steering

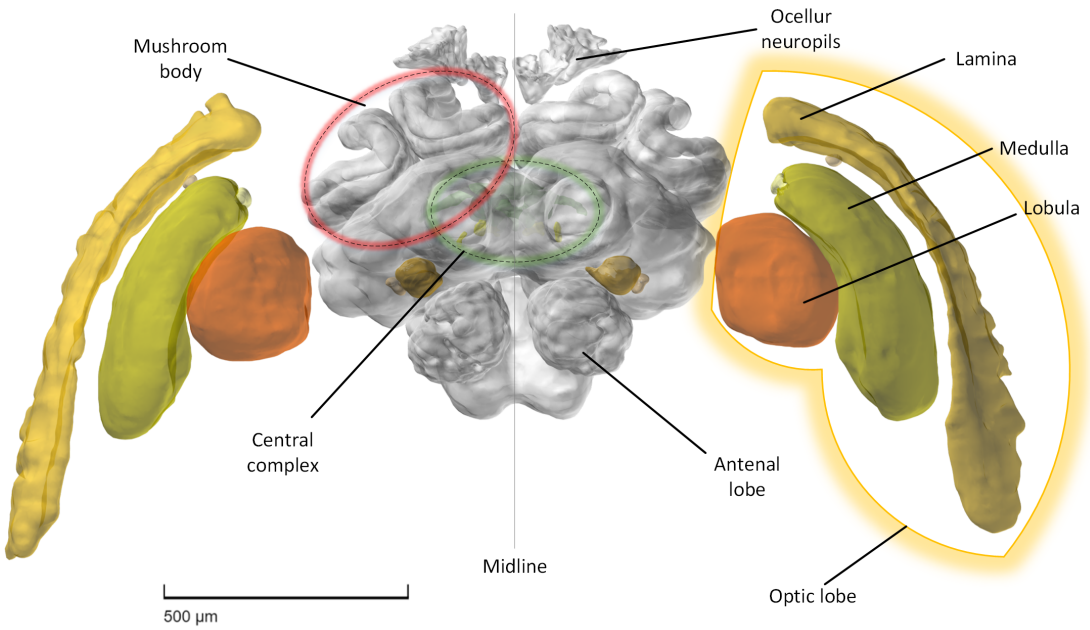


Figure 2.5: Schematic of the brain of a female Megalopta genalis generated with the insect brain database. Key neuropils that are relevant to navigation are labelled.

commands. Retinotopic visual information is known to propagate from the medulla and lobula into the anterior optical tubercle (AOTU) in honey bees (Mota et al., 2011), which also has connections to the mushroom bodies and central complex. The AOTU may thus be a crucial processing area and an information conduit in the task of visual navigation. This neuropil is also known to play a central role in the processing of polarisation pattern information for generating a compass heading.

2.4 The visual mechanisms of state estimation in flying insects

A prerequisite for the path integration behaviour reviewed in section 2.5, is the ability of flying insects to sense both their egomotion in body coordinates (section 2.4.1) and their heading in global coordinates (section 2.4.2). These signals are thought to be inferred primarily via visual sensorimotor systems.

2.4.1 Sensing egomotion with vision in flying insects

Optic flow is a visual cue that many animals exploit in order to sense their egomotion (J., 1950). Because this concept is also widely studied from a computer vision perspective, this section begins with a general definition of optic flow and information about its properties before switching to a review of the neuroethological perspective of optic flow and its role in estimating speed and height in flying insects.

2.4.1.1 A general definition of optic flow

As a camera or an eye (visual signal sink) moves around in an environment, the pattern of light intensity observed at the signal sink shifts according to 1) its egomotion and 2) in-scene motion. The apparent motion of the measured brightness pattern between temporally separated samples is referred to as *optic flow* (Horn, 1986). Where *"apparent"* alludes to the fact that detected levels of brightness can also vary due to; 1) lighting changes 2) discontinuities, e.g. caused by in-scene occlusions or signal sink boundaries 3) geometric distortions at the source (e.g. non-rigid body motion) and sink (e.g. lens properties) 4) photometric noise at the sink. Objects can also be moving without causing any change at the signal sink (e.g. an untextured spinning sphere). The actual motion, or ground truth, of points in a 3D world as witnessed in from a 2D imaging surface, is known as a *motion field*.

2.4.1.2 The properties of egomotion induced optical flow

When a camera moves around a three-dimensional environment with six degrees of freedom, the resultant optic flow field is typically non-uniform as illustrated in the frames of fig. 2.6a. Motion type (rotational or translational), motion magnitude and motion direction are properties that have a bearing on all types of egomotion induced

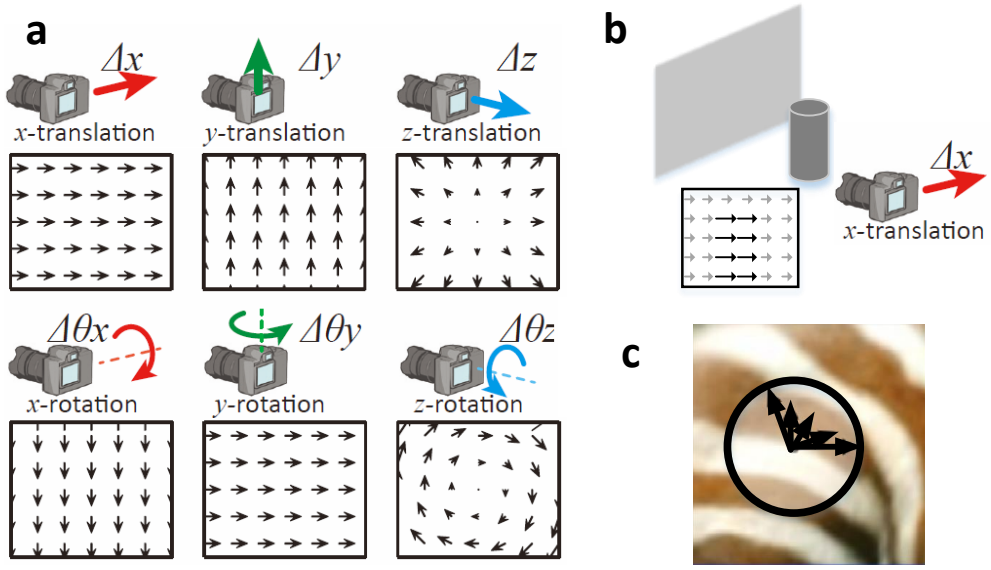


Figure 2.6: Egomotion optic flow field properties. **a**) The motion indicated by the coloured arrows associated with each camera icon causes the optic flow fields shown in the frame beneath each camera when the camera is moving perpendicular to a flat plane. Adapted from (Jo et al., 2015) © 2015 IEEE. **b**) For translatory motion, flow vectors are inversely proportional to depth, nearby objects therefore induce greater flow. **c**) The aperture problem, when a scene is viewed through an aperture (denoted by the circle), the direction of motion of solid bars is ambiguous as indicated by the arrows.

optic flow. As a camera moves about or along an axis of motion the flow field magnitude increases as a function of the motion magnitude. However, no flow is induced at the expansion point, where the view and motion directions coincide. As the angle between the view and motion direction increases so does the induced flow, with the maximum flow generated when the view direction is parallel to the flow inducing surface. This is why the flow vectors at the centre of the z-translation and z-rotation frames in fig. 2.6a have lower magnitudes than those at the border. For translational motion, scene depth is inversely proportional to the magnitude of induced flow (as illustrated in fig. 2.6b).

2.4.1.3 Sensing motion with optical flow

Low latency attitude estimation is required for stable flight control and flying insects are reliant on vision for this task. More than half of the central nervous system of a fly is dedicated to vision processing (Silies et al., 2014), with a large portion of

this involved in motion perception. An extensive body of work with the objective of mapping the visual motion processing circuit of fruit flies and other dipteran species exists (Borst et al., 2019; Silies et al., 2014), with recent gains supported by advances in neurophysiological and genomic techniques. At the processing level, motion detection circuits in flying insects are thought to be widely conserved across species (Douglass and Strausfeld, 2001). As a comparison, the bumble bee's motion detection system is reviewed in (Paulk et al., 2009). Equivalent neurons are found in both families although there are some structural differences at the lobula.

Motion detection in insect vision is served by an array of fundamental motion detecting units, widely referred to as elementary motion detectors (EMDs). The basic EMD configuration was first posited by Hassenstein and Reichardt (Hassenstein and Reichardt, 1956), who hypothesised that the temporal duration of the motion of an intensity shift across neighbouring photoreceptors can be measured according to the scheme outlined in fig. 2.7a-b. While the neural circuit is now known to be somewhat more complicated than this schematic (these nuances are discussed in (Borst et al., 2019)), the basic principles outlined in fig. 2.7b remain an accurate functional description of the basic unit of motion perception in insect vision to this day.

While the EMD cannot discriminate between intensity shifts caused by egomotion and exomotion, the downstream wide-field lobula plate tangential cells (LPTC) are heavily implicated in the detection of egomotion. As their name would suggest, these neurons run tangentially and are post-synaptic to the nerve endings of putative EMD circuit output neurons (Borst et al., 2019), and thus are an integration site of motion information from a large solid angle of the retina. In contrast to the constituent parts of the EMD, neuroscientists can record from the LPTCs with relative ease. The receptive fields of several LPTCs has been mapped in a laboratory setting, examples of which are presented in fig. 2.7c-d. This data indicates that each LPTC acts as a *matched filter* to the animal's rotation around a particular axis. That is, the LPTC firing rate has an increased output when the detected optic flow field is consistent with the anticipated flow field generated by motion about the axis of a preferred direction. The response of a subset of LPTCs to an increasing stimulus of rotation flow around the preferred direction is linear (within bounds) (Franz et al., 2000) providing strong evidence that this is their function. LPTCs polarise or depolarise depending on the direction of motion which means that each neuron can infer and encode clockwise or anticlockwise egomotion (Hausen, 1984).

The majority of described LPTCs relate to rotational optic flow matched filters

(Kohn et al., 2018), 50 out of approximately 60 LPTCs in the dipteran brain have now been characterised as such (Longden et al., 2017). While rotational detection benefits from large neurons that subtend a wide field of view (Koenderink and van Doom, 1987), this is not necessarily true of translational neurons, which could explain the comparative dearth of described neurons with this characteristic. In 2017, two neuron types (TN1 and TN2) were recorded in sweat bees and were found to have properties commensurate with translational optical flow matched filters (Stone et al., 2017). These neurons are located in the central brain, downstream from the optic lobes. They have preferred directions at $\pm 45°$ and $\pm 135°$ in azimuth with respect to the animal's anterior. TN neurons appear to be sensitive to lateral motion in the insect's field of view which is likely to be particularly salient for detecting egomotion in the forested habitat of sweat bees. TN1 neurons have also been recorded in the bumble bee and are likely to be conserved across at least other species of Hymenoptera.

VT1 is a second neuron with a receptive field that implies sensitivity towards translational motion. It was recently identified in the lobula plate of blow flies (Longden et al., 2017). Described by the authors as a sideslip motion detector, VT1 has a similar peak azimuth sensitivity as the TN neurons. The variation of the receptive field of VT1 in elevation was also measured and shown to have a peak sensitivity at 35° beneath the horizon. As shown in fig. 2.7d, VT1 is sensitive to motion right down to 70° below the horizon. This neuron is also reported to produce spike bursts when a simulated elevated object is presented to the test animal on a screen. This characteristic is described by the authors of (Longden et al., 2017) as a potential mechanism for the detection of motion parallax caused by prominent objects in the ventral field of view.

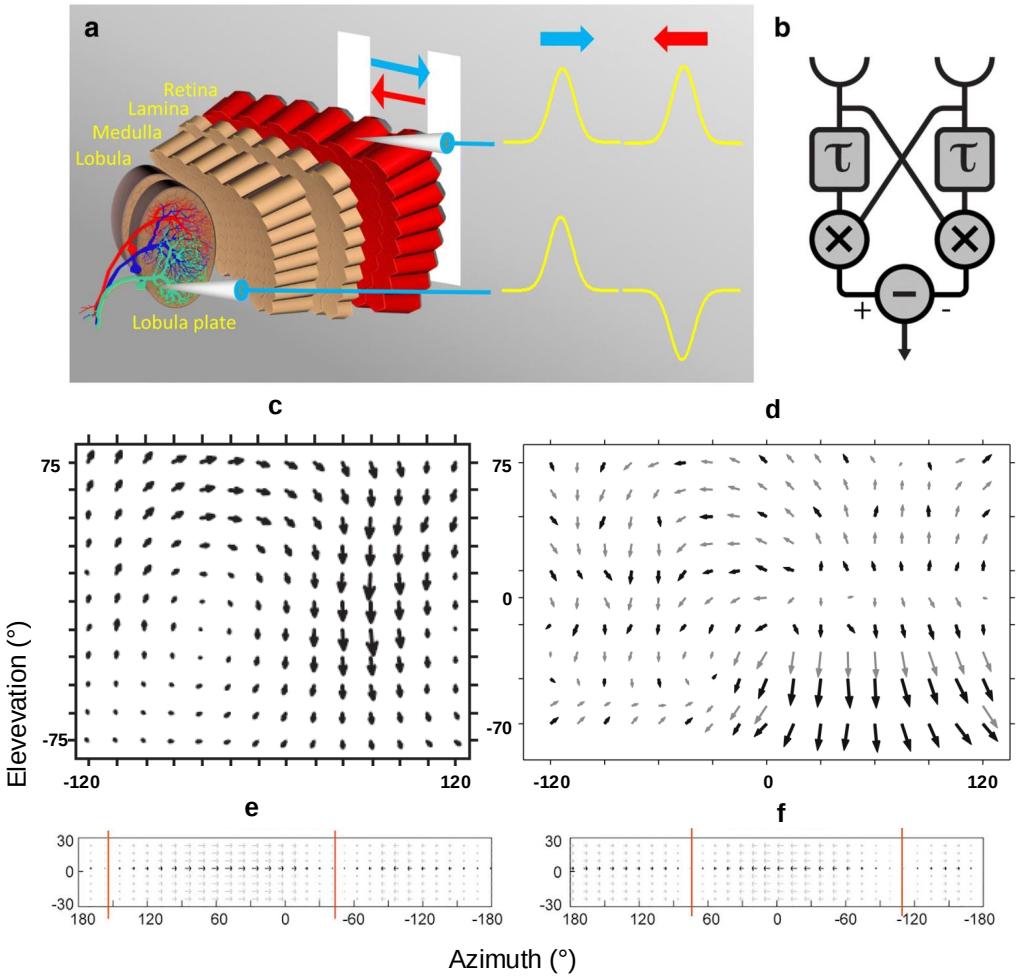


Figure 2.7: Motion detection in insect vision. **a**&**b** from (Borst et al., 2019) licensed under CC BY 4.0. **a**) 3D Schematic of the visual optical neuropils and traces of the lobula plate tangential cells (LPTC)s in the lobula plate. A bright bar moving across the visual field leads to a depolarisation of the photoreceptors as measured by the upper electrode. The lower electrode demonstrates that the LPTC polarises or depolarises depending on the direction of the bar's motion. **b**) The Hassenstein–Reichardt EMD model. τ=low-pass filter. If a delayed signal from one of the photoreceptors arrives at the relevant multiplication node at or near the same time as the other photoreceptors undelayed signal, the output is non-zero. **c**) Receptive field of the VS8 (rotationally sensitive) neuron in the fly lobula plate. modified with permission from (Krapp and Hengstenberg, 1996) **d**) receptive field of the VT1 neuron from (Longden et al., 2017) with permission from Elsevier. **e**&**f**) Receptive fields of the TN1 neuron corresponding to the sweat bee left and right optic lobes respectively modified from (Stone et al., 2017) with permission from Elsevier.

2.4.1.4 Flight control with optic flow in tunnels

Honey bees that are trained to fly through a narrow tunnel with textured walls conduct their flight path along the midline of the tunnel (Srinivasan et al., 1991) (fig. 2.8a). Interestingly, when one of the walls is motorised and driven in the opposite direction to that of the animal's flight, the animal shifts its lateral position, towards the moving wall (fig. 2.8b), thus serving to counteract the artificially induced flow. This procedure supports the notion proposed in (Coombs and Roberts, 1992) that flying insects control their lateral position, at least in confined spaces, by maintaining an equilibrium of lateral flow detected by the animal's left and right eyes. Furthermore, the forward speed of the animal is related to the width of the tunnel, prompting the idea that a honey bee controls its speed by maintaining a fixed optical flow rate (Srinivasan et al., 1996). These mechanisms are known as optic flow *regulation* systems. The centering response behaviour occurs in tunnels where the left and right walls have vertical sine gratings of different spatial frequencies or intensities (Baird et al., 2005) (fig. 2.8d-e), suggesting that the optical flow calculation (LPTC output) is independent of spatial information and contrast. A plausible neural mechanism for this has been proposed in (Cope et al., 2016).

Honey bees and bumble bees flying through a tunnel have also been shown to predictably mediate their height position in response to vertical tunnel constrictions (Portelli et al., 2011a) and flow induced by a moving tunnel floor (Portelli et al., 2010). That is, vertically narrower tunnels cause the animal to slow down, and floors that move in the same direction as the animal cause it to reduce its height. These experiments demonstrate that the putative optical flow regulator must flexibly serve at least two different control setpoints (lateral and ventral), complicating the path integration odometry task considered in section 2.5.1. A model for dual channel optic flow control was proposed in (Portelli et al., 2011b), with model simulations yielding similar trajectories as the experiments in (Portelli et al., 2011a).

Bumble bees flying into a region of increased clutter slow down ahead of the transition, whereas when they depart from an area of high clutter they do not alter their speed until they have cleared this region (Linander et al., 2015b) (fig. 2.8h-i). According to the optical flow regulator paradigm, this suggests that the area of maximum optical flow is attended to for speed regulation. This idea is further evidenced in (Lecoeur et al., 2019), where 3D obstacles were placed in a tunnel causing the animals to slow down to a rate proportional to the distance to the nearest surface. The author also predicts that

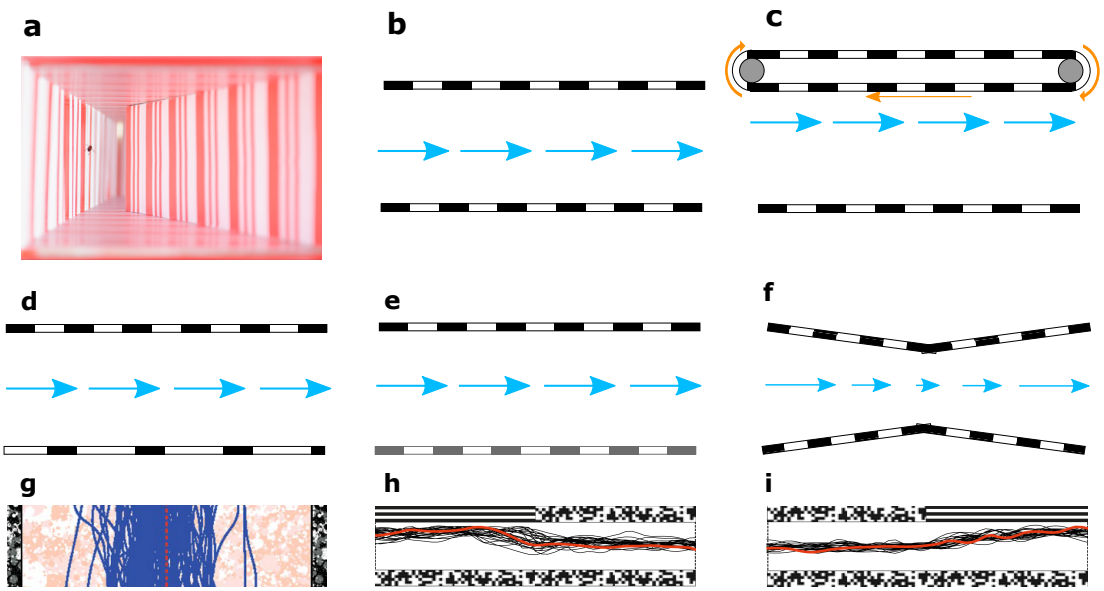


Figure 2.8: Tunnel experiment paradigms **a**) Photograph of a bee flying through a constricted tunnel with vertical stripes - extracted from (Serres and Ruffier, 2017), licensed under CC BY 4.0. **b-f**) overhead schematics of experimental paradigms. The blue arrows shows the flight path with their length representing the animal speed at that location. Black and white gratings show the texture that is on the flight tunnel walls and floor. **g**) real trajectories (blue lines) of bumble bees flying through a wide tunnel with natural texture on the floor and walls **h-i**) real trajectories (black lines) of bumble bees flying through a narrow tunnel with walls containing a transition from low to high optic flow inducing surfaces and vice versa. The axial striped surfaces indicate surfaces that generate low amounts of optic flow commensurate with translational motion. Panel **g** is extracted from (Linander et al., 2015a) with permission from *Springer Nature* and **h-i** is adapted with permission from (Linander et al., 2015b)

lateral position, speed and height are all controlled with different strategies.

When bumble bees fly in tunnels of progressively increasing widths (arguably tending towards more naturalistic conditions), the position control strategy can be seen to shift from one of predominantly lateral control to ventral control (Linander et al., 2017). Animals with access to ventral optic flow texture in wide tunnels are able to fly straight along the length of the tunnel but they do not centre their flight on the tunnel midline(fig. 2.8g). When the floor texture is removed, the same animals exhibit a comparatively tortuous flight path.

Finally; edge tracking, wide-field stabilisation, and ventral expansion signals have been shown to shape fruit fly height control (Straw et al., 2010). These modalities are

also likely to exist in the species-of-interest, further increasing the number of possible signals involved in a biologically plausible odometry circuit.

In summary, flight tunnels have provided useful insights into the flight control of bees. This line of enquiry has facilitated the development of control systems that accurately mimic bee flight control in certain conditions. However, flight tunnels have planar surfaces with unnatural depth statistics, and the applicability of these models has not yet been evaluated in more natural environments.

2.4.1.5 Speed and height estimation and control in open space

For navigation behaviours like path integration (section 2.5.1), the displacement or *ground speed* of the animal is a more obviously useful metric than raw optic flow for gauging journey length. However, to convert optic flow into speed, the distance and viewing angle between an eye and the flow inducing surface is also required. Since Hymenoptera are not thought to be able to perceive a useful range of depths directly (e.g. by stereopsis Werner et al. 2016), it is widely stated that flight speed and height are simultaneously mediated by optic flow (Serres and Ruffier, 2017). The use of airspeed to mediate a constant balance of speed and height was proposed in (Lecoeur et al., 2019). Presumably the auxiliary visual height control mechanisms observed in flies (Straw et al., 2010) could also play a role in this task. Neither of these mechanisms has yet been formulated or tested on a robotic platform (Lecoeur et al., 2019), but this would provide a means of testing existing formulations in natural scenes.

Because of their small size, it is difficult to record the trajectories of flying insects while they are foraging. Harmonic radar (Capaldi et al., 2000) has provided invaluable information about the routes of foraging honey bees (Riley et al., 2003) and bumble bees (Riley et al., 1999). The ability of these insects to keep a straight course in strong side-wind was ascertained through this method. However, harmonic radar only reports a 2D horizontal location, at low spatial accuracy ($\pm$10m), low temporal resolution (0.33Hz) and it can only be used in open ground. Due to these limitations, it has not yet been possible to evaluate the models developed from flight tunnel paradigms with natural datasets.

Anecdotal evidence suggests that in strong head wind conditions, bees slow down and move closer to the ground (Riley et al., 2003; von Frish K., 1967). But whether this is to maintain an optic flow setpoint, or to avoid the wind is unknown. Honey bees are also said to avoid flying over dynamic or low textured surfaces such as water bodies, and if induced to do this, they have a tendency to lower their altitude to the extent that

they often crash into the water surface (von Frish K., 1967). The typical cruising height of bees commuting in open space has not been formally measured but the authors in (Degen et al., 2015) state this to be approximately 8m (personal communication).

2.4.2 Global heading estimation

Many species of insect are known to infer their global (*allothetic*) heading state using the location of the sun (Homberg et al., 2011; von Frish K., 1967). This is even time-compensated so that it can be used at any time of day (Dyer and Dickinson, 1994). Some animals perform tasks with an *idiothetic* heading estimate (Kim and Dickinson, 2017), whereby the animal integrates rotational rate information over time. In fig. 2.9, a comparison of the relative performances of robotic systems using equivalent idiothetic and allothetic heading information is presented. A path integration (section 2.5.1) system with an allothetic source of heading information provides a better spatial location estimate than an idiothetic system because the errors are not accumulated (Heinze et al., 2018) (i.e. the current heading state estimate is not dependent on the previous estimate).

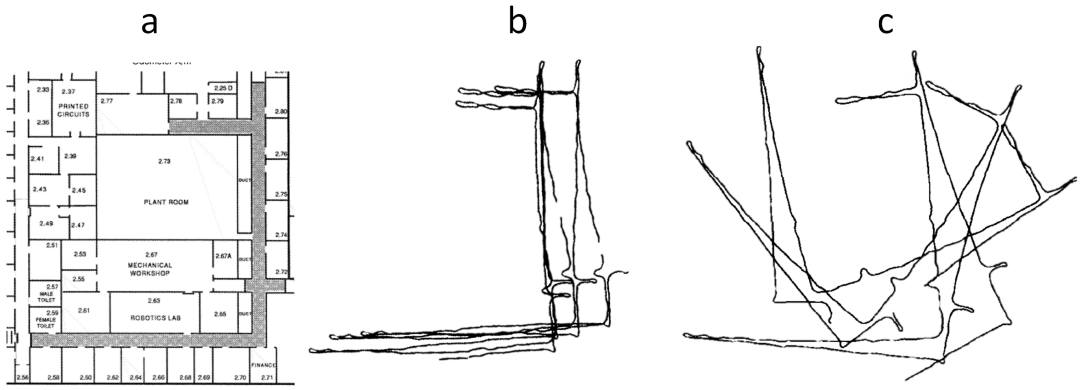


Figure 2.9: Benefits of allothetic heading in path integration, adapted from (Duckett and Nehmzow, 1998) with permission from Elsevier. A wheeled robot traverses the corridor shaded in grey in the building plan in **a**. **b**) Path integration trajectory when compass (heading) sensor is used (allothetic). **c**) Robot path integration trajectory for the same robot route as **b**, but using integrated angular rate gyro for heading data instead of compass (idiothetic).

Insects are known to be able to infer the sun's location from celestial polarisation patterns. The angle of polarisation increases with distance from sun in the sky (fig. 2.10)

and the skyward facing photoreceptors of insect compound eyes have a tendency to be polarisation sensitive. In cue conflict experiments (using a mirror to shift the perceived location of the sun) flying (von Frish K., 1967) and terrestrial (Wehner and Müller, 2006) insects both preferentially navigate with the polarisation pattern. The advantage of polarisation patterns is that the sun's location can be inferred under cloud cover (Pomozi et al., 2001) and in scenarios where only small patches of sky are visible (e.g. under a forest canopy).

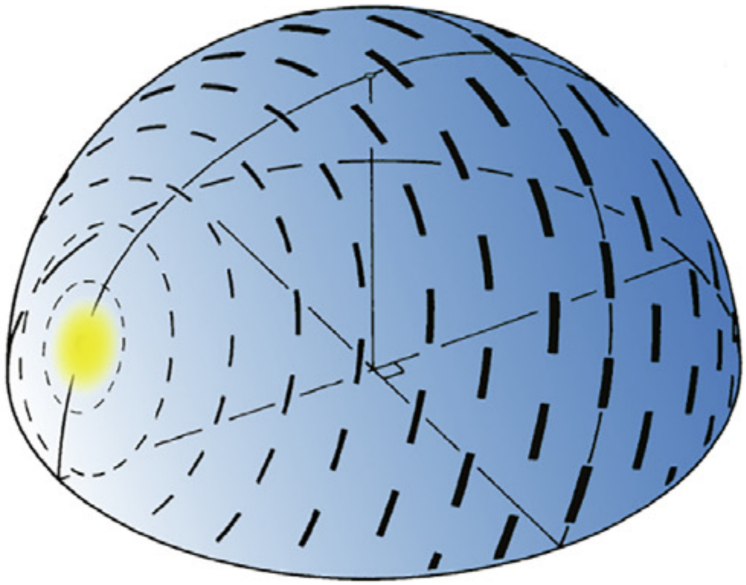


Figure 2.10: Schematic of polarisation patterns in the sky modified with permission from (Honkanen et al., 2019). Direct sun is not polarised, the widths of the concentric dashed line circles centered on the sun represent the degree of polarisation which gradually increases with subtended angle from the sun.

The neural pathways of circuits from the retina to the protocerebral bridge (see fig. 2.12) in the central complex have been mapped (Heinze and Homberg, 2009) and the columns of this structure have been demonstrated to encode the animal's heading (Heinze and Homberg, 2007; Homberg et al., 2011). Neural recordings suggest that cells in the protocerebral bridge act as a ring attractor, with the heading encoded as a bump of activity in an ensemble of eight neurons (TB1 neurons). While other heading cues such as integrated optic flow (Kim and Dickinson, 2017) and landmark location have also been shown to propagate to the same brain area, a view of the sky appears to be crucial for path integration in honey bees as evidenced in (Dacke and Srinivasan, 2008). Here, honey bees trained to fly through a tunnel that is partially covered by opaque panels appear not to accumulate motion conducted in the covered sectors.

2.5 Local navigation in flying insects

In this section the local navigation capabilities of the species-of-interest are reviewed. These are path integration (section 2.5.1 and visual navigation section 2.5.2)

2.5.1 Path integration

Path integration, a cue that is central to the navigational capabilities of foraging insects, has recently been reviewed in a number of journal articles (Collett, 2019; Heinze et al., 2018; Webb, 2019). It can be defined as the ability of an animal to estimate its current location, relative to a place of significance, by integrating its motion over time. The ability to use this estimate to guide a direct route towards the goal location is another key aspect of path integration. Path integration requires a continuous estimation of motion (see section 2.4.1.5), and for the species-of-interest, an allothetic heading estimate (see section 2.4.2). Note that unless these estimates are error free, an error will accumulate over time. This phenomena is known as *drift*.

2.5.1.1 Behavioural evidence

Displacement experiments constitute strong behavioural evidence that flying insects possess a path integration system. Bees that are displaced in an opaque container from a learned feeder site to a novel location consistently travel in the direction and distance that their nest would be if they were at the feeder site. In (Riley et al., 2003), such a procedure was conducted (the resultant trajectories are plotted in fig. 2.11a) and the presence of novel visual features typically did not impact this trend. However, a geographic boundary between fields was sometimes detected and enabled a subset of animals to navigate home via the visual navigation mechanisms outlined in section 2.5.2.

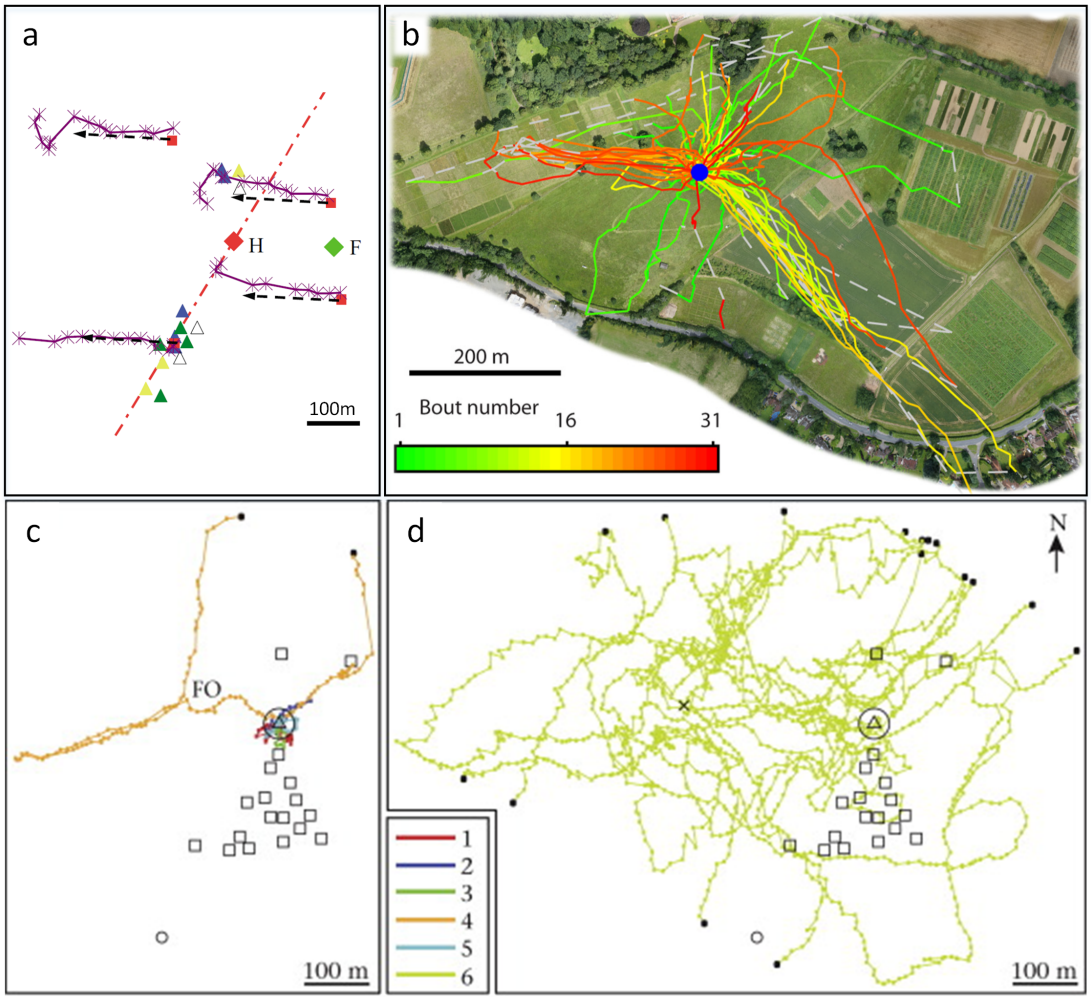


Figure 2.11: Path integration trajectories **a**) From (Riley et al., 2003), bees trained to fly between H and F are displaced from F to nearby but novel locations (red squares). The displaced bees travel the distance and direction towards their would be nest before commencing a search pattern. Purple lines show example trajectories, dashed black lines indicate the trajectory that would take the animal back to H under perfect path integration guidance. The triangles and red dashed line denote the locations of distinctive landmarks. **b** The entire ontogeny record of a bumble bee, figure from (Woodgate et al., 2016) CC BY 4.0. Several of the early bouts have asymmetric inbound/outbound routes and pass over terrain that would cause height discontinuities. **c**-**d**) trajectories recorded with harmonic data from (Degen et al., 2015) CC BY 4.0, the triangle indicates the hive location and the open circle denotes an estimate of the maximum distance that bees could see their nest from (30m). Open square symbols indicate the location of distinctive landmarks, filled black circles show where the animal went out of radar range. The first five flights of a scouting honey bee (**c**) are followed by a long convoluted outbound route marked by the green line in **d**.

Recent harmonic radar based ontogeny procedures (Degen et al., 2018, 2015, 2016; Woodgate et al., 2016) confirm the considerable path integration abilities of both honey bees and bumble bees. Both species display the ability to perform long convoluted outbound routes in novel terrain as shown by the trajectories in fig. 2.11b-d. It is also apparent from fig. 2.11b that the outbound and inbound routes of a particular foraging trip often do not overlap and pass over different ground level features. For example, in some instances the outbound route passes over a combination of meadow, forest and field boundaries, whereas the inbound route exclusively traverses open meadow land. Depending on how the insect negotiates these obstacles, considerable error could accumulate in a path integration circuit that relies on optic flow regulation section 2.4.1.4.

2.5.1.2 What is measured?

To date, interpreting the waggle dance (described in section 2.2.1.1) has been the de facto method of measuring honey bee path integration circuit state (Srinivasan, 2015). In the last 20 years there has been an emerging consensus that optical flow is the metric conveyed in the waggle dance, rather than absolute distance or energy expenditure. The principle evidence being: 1) recruiting bees foraging between high buildings or between the ground and a weather balloon, signal an under-estimate of the distance in their waggle dance, even though they travel further (Esch and Burns, 1996) 2) Animals that pass through a short (6m) but narrow textured tunnel close (6m) to their hive conduct waggle dances that overestimate their journey by a factor of 30 (Srinivasan et al., 2000).

There is some evidence to suggest that the distance encoded by the waggle dance is not the same entity as that used by the path integration circuit. An alternative means of estimating the range encoded by the path integration circuit state is to train an animal to forage in a certain location and then in a test scenario, remove the feeder and monitor where that animal's search pattern is centered. It has been shown that this method reports a different value from the waggle dance in experimental conditions (Dacke and Srinivasan, 2008). When feeder sites are shifted closer or further from the hive, the waggle dances of recruiters usually continue to reflect the original distance for several subsequent forage trips (Chatterjee et al., 2019). It is also not yet clear what foraging bees interpreting the waggle dance actually measure (Ai et al., 2019; Greggers et al., 2013), therefore it cannot be guaranteed that the full scope of the communication is understood (Menzel, 2019). Finally, the range and distance parameters measured by biologists have a high variance (Tanner and Visscher, 2008), with recruited bees some-

times searching in locations 200m away from the communicated feeder site (Preece and Beekman, 2014). While the reason for this variance is open to interpretation, this degree of error appears at odds with the relatively precise path integration capabilities of honey bees set out in section 2.5.1.1.

However, if a more pertinent metric (such as absolute distance) of a particular journey exists, why would this not be communicated by the waggle dance? In section 2.4.1.5, it was established that optic flow is flexibly measured in different portions of the eyes of bees. If optic flow is the raw unit for path integration systems, how are multiple optical flow values from different eye regions converted into a single odometry value? It is speculated in (Linander et al., 2017) that ventral optical flow would provide the most reliable cue in natural environments due to the omnipresence of the ground and its relatively consistent depth value.

2.5.1.3 Anatomy, neurophysiology and modelling

The central complex is a cluster of neuropils at the centre of the insect brain (see fig. 2.12a-b for orientation). Each of the constituent neuropils forms a continuation of a structure containing 16-18 columns, where the number of columns is dependent on the species and each column is similar in structure to its neighbours. Neurons from several functional regions of the brain synapse onto neurons in the central complex and horizontal neurons provide connectivity of these signals across the structure. The central complex is therefore thought to be a site of signal integration in the insect brain (Heinze et al., 2018; Honkanen et al., 2019).

There is now significant evidence that the central complex is the area where path integration occurs in insect brains (Heinze et al., 2018). A model inspired by, and faithful to the connections between, identified neurons in the central complex of Megalopta genalis is presented in (Stone et al., 2017). The model, depicted in fig. 2.12c, receives input from a population of neurons that are known to encode heading information (TB1) and neurons that were shown to fire with a frequency that is correlated with the rate of optical flow consistent with translatory motion (TN) neurons (functionality presented in section 2.4.1.3). As with the anatomical data, the model has a columnar structure with eight columns per central body hemisphere. It is thought that populations of putative memory cells (CPU4) in each column could accumulate a representation of how far the insect has travelled in the direction represented by that column (dictated by the velocity sensitive neurons). Steering cells (CPU1), are thought to be able to convert the memory state into an action (commanded yaw rate) that will serve to steer the animal in the

opposite direction (i.e. 180° offset) to the vector encoded by the memory neurons.

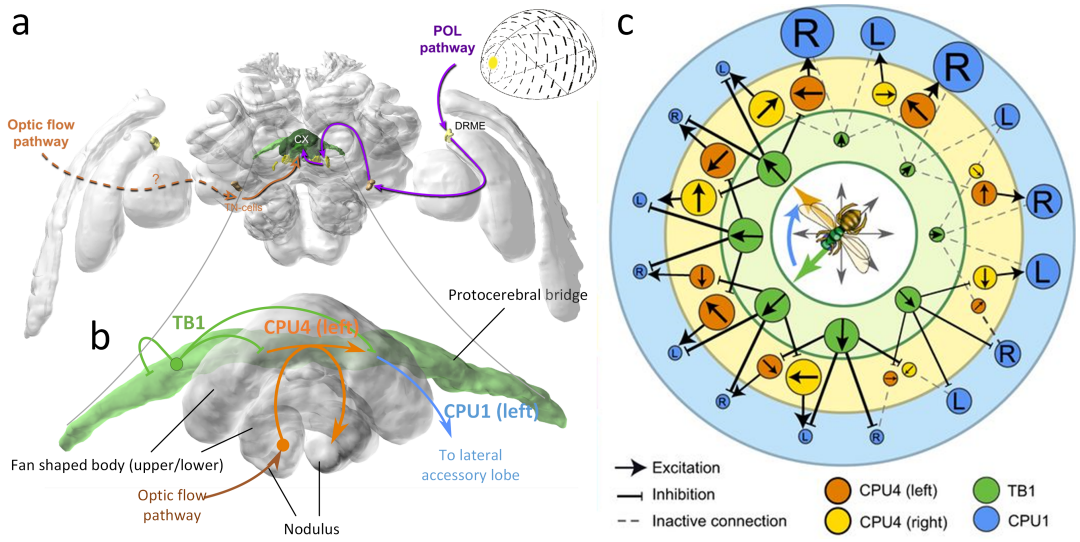


Figure 2.12: Anatomy of the central complex circuit. **a**) Orientation of central complex circuit and optic flow/POL compass pathways. **b**) Central complex path integration circuit schematic, generated using the insect brain database. **c**) Schematic of the connections in the central complex circuit. The large concentric circles denote brain areas. Each of the smaller circles represents a neuron (of type specified by the key below) and its radius denotes that neuron's current activity. In this snapshot, the green arrow shows that the insect's current heading is deviated from the peak of activity in the memory ring (orange cells). The right steering cells (Blue R (CPU1)) are offset clockwise by one column from the memory cells. In this case they produce more activity than the left cells, which are offset *anti-clockwise* from the memory cells. As a consequence, the circuit outputs a steer right signal. Panels a&c are modified with permission from (Honkanen et al., 2019)

2.5.2 Visual navigation

Visual navigation is a process by which the species-of-interest use information from their current view in order to infer what action to take to progress towards its goal. Usually this process depends on a prior exploration of its habitat and the storage of salient visual information. Visual navigation does not suffer from sensor drift but it can have a relatively short range. It therefore compliments path integration which has diametrically opposite properties. Visual navigation is an umbrella term for several putative cues that rely on the snaphsot based view comparison mechanisms that are

outlined in section 2.5.2.2. The terminology relating to these cues has evolved in recent years. Here, the two cues of interest for this work are defined as follows:

1. **visual homing** - a visual navigation mechanism associated with inferring a goal direction from a single view memory with potentially a large spatial displacement from a reference snapshot.

2. **visual route following** - a visual navigation mechanism associated with the following of a chain of overlapping view memories formed along a route, thus allowing visual navigation to occur well beyond the catchment area of a single view memory.

2.5.2.1 Behavioural evidence for visual navigation

It is known that honey bees displaced hundreds of meters away from their nest entrance (i.e. where their path integration circuit is "discharged" or in a *zero-vector* state) are able to relocate their nests (Menzel, 2019). It is thought that this distance is outside the range of odour cues (Degen et al., 2016), therefore this ability is widely attributed to visual navigation (Menzel et al., 2000).

Building on the work in (Menzel et al., 2000), particularly strong evidence that visual memories are important for homing in honey bees is presented in (Degen et al., 2016). Here, bees with differing levels of foraging experience are displaced from their nests and tracked with harmonic radar. In open farmland, 13/13 displaced honey bees that had not conducted any learning flights were not able to find their way home. 11/13 bees that had conducted only a short learning flight (extending less than 30m radially from the nest) managed to home successfully. Their flights were typified as convoluted (akin to a search pattern) but culminated in a direct homeward path. This direct homeward path typically originated in a location that they had experienced on their learning flight. 40/50 experienced foragers returned home after being displaced, their routes tended to be convoluted when they were in unfamiliar territory and direct when they were in previously visited locations. 7/10 of the experienced foragers that failed to home were flying in poor weather conditions, suggesting lighting variation and/or strong winds can have an influence on honey bee visual navigation abilities. The same procedure was repeated with a relocated hive in (Degen et al., 2018); displaced animals that were experienced foragers but had not performed *reorientation* flights were not able to locate the nest. Bees that performed a single reorientation flight were

able to relocate the nest. This ability varied between hives and may depend on the visual differences between old and new sites.

2.5.2.2 Mechanisms of visual homing

In (Cartwright and Collett, 1983), the authors note that honey bees trained to a feeder in the proximity of a visual landmark learn the angle that the landmark subtends on the animals retina, rather than a feature based representation of it. This information prompted Cartwright and Collett to conceptualise the *"snapshot model"*, whereby insect brains compare currently experienced holistic views with equivalent previously experienced and memorised views in a retinotopic *pixelwise* fashion.

The trajectories of homing flights of wasps (Zeil, 1993) and bumble bees (Dittmar et al., 2010) consist of regular sectors of sideways translation. The purpose of these sectors is thought to be concerned with inducing motion parallax for the estimation of distance to prominent landmarks. Wasps preferentially make use of this motion parallax information for early flights from a particular nest but as they get more experienced the sideways translations are less prevalent and retinal snapshots are thought to become the primary cue (Zeil, 1993).

Snapshots captured with a digital camera in natural scenes have an attractor basin or *catchment area* in 3D space (Zeil et al., 2003). That is, the mean squared error between a reference image and an equivalent image generated at a translated location will monotonically increase up to the boundary of the catchment area (explored in fig. 2.13). Thus, a gradient descent algorithm or embodied procedure can be used to home in on the location of the reference image as demonstrated in simulation (Basten and Mallot, 2010) and on robotic platforms (Lambrinos et al., 2000). The size of the *catchment area* of a snapshot depends on the topography of the environment (Philippides et al., 2011; Stürzl and Zeil, 2007).

One issue with the snapshot model is that image comparisons are only meaningful if the animal's current heading is closely aligned with its heading when the snapshot memory was instantiated. Considerable effort to determine how desert ants align their views has been undertaken (reviewed in Collett et al. 2013; Möller and Vardy 2006), with the visual compass being a widely adopted modelling strategy (Labrosse, 2006). However, unlike desert ants, flying insects can rapidly translate sideways, and have scope to perform gradient descent with this type of motion (Collett and Collett, 2002). In (Cartwright and Collett, 1987) it is suggested that flying insects might associate views with heading directions, a mechanism that could help reduce the likelihood of false

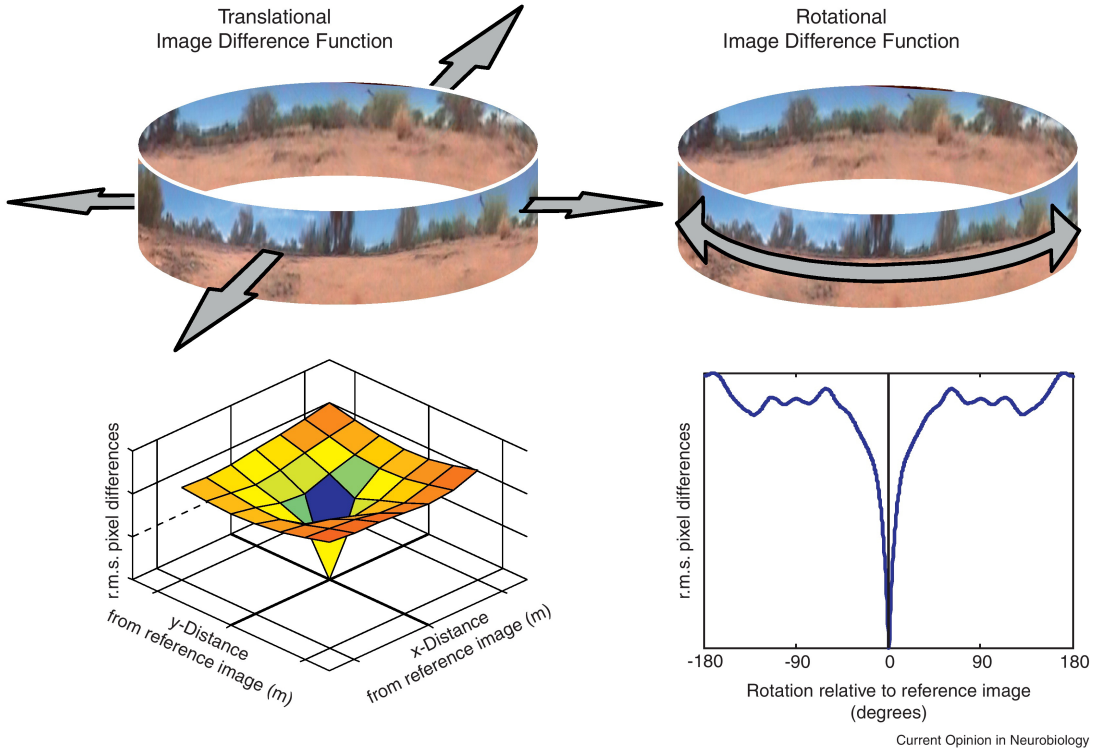


Figure 2.13: Properties of panoramic snapshots and image difference functions, copied from (Zeil, 2012) with permission from Elsevier. **Left:** The view difference of an agent translating in the XY plane generates a smooth monotonic search space (shown beneath) which increases as the agent moves away from the reference image in all directions. **Right:** as the agent rotates in yaw at a fixed point in space, an image difference function (shown beneath) produces a prominent peak when the orientation of the insect is aligned to the reference image.

positives with snapshot memories. Indeed, guided by their celestial compass, bees and wasps have a tendency to approach their goal from a consistent goal direction (Collett and Collett, 2002). The ability to view the landscape from above may also provide opportunity (Gaffin et al., 2015). However, flying insects have the extra dimension of height to contend with. This matter is considered in (Differt and Stürzl, 2020), culminating in the description of a proof of concept for navigation along routes using a helical flight path that facilitates z-axis direction sampling.

2.5.2.3 Visual route following

Zero-vector flying insects can visually home from distances well outside of the catchment area of their nest (Degen et al., 2016). It has been suggested that these animals

could link several catchment areas together using *visual route following*. Conceptually, this involves the periodic storage of snapshots along a route that is first navigated by path integration. In subsequent flights, the animal can follow this chain of snapshots as an additional navigation cue. This provides animals with a long-range backup mechanism, reducing the required performance from the path integration circuit. Additionally, visual routes could provide a direct pathway to the animal's hive from a short range boundary, thus minimising time spent in a search pattern at the end of the flight. The notion of visual route following has been particularly widely studied in desert ants that live in dense vegetation. These animals display idiosyncratic inbound and outbound routes; zero-vector ants that are displaced out of visual contact with their nest but along their inbound route are able to travel back to their nest (Mangan and Webb, 2012).

Models of visual route following have demonstrated that a terrestrial agent can be steered across a route in a simulated world using a visual compass in isolation (Baddeley et al., 2012). Views can be sparsely encoded in a neural network reducing memory and processing requirements (Baddeley et al., 2012). This has been implemented with a biologically plausible mushroom body model (Ardin et al., 2016) which demonstrated a potential neural correlate with sufficient memory capacity for a representative homing route. A terrestrial robotic implementation of this model has also been deployed in (Kodzhabashev and Mangan, 2015). Of relevance to this work, the model has also been deployed in a simulated aerial agent (Müller et al., 2017), the authors report a similar performance as (Ardin et al., 2016) in worlds with trees but the model is less successful in flat featureless worlds.

An interesting aspect of the exploratory and return paths of honey bees is that they are often guided by ground level elongated structures such as ditches, hedgerows and pathways (Menzel et al., 2019). There is sometimes a correlation between displaced animals encountering these features and a transition between an exploratory and return flight path (Riley et al., 2003). While these structures constitute a convenient guidance aid, the mechanisms for navigating with such structures is less apparent. How do the animals know when to join and depart these *"flyways"* for example? Displaced animals generally do not confuse similar elongated structures with the one that they have learned (the gravel paths in (Menzel et al., 2019) for example). Therefore, animals seem to be able to visually identify where they are in relation to a particular ground structure. Although it should be noted that recent experiments in padi fields have shown that honey bees can get confused by very self similar structures (personal communications Andy Philippides). Bio-inspired modelling of algorithms using ground structures

as topological guides suggests the guides have lower spatial aliasing than forests or water regions (Mao et al., 2020), providing an alternative rational for following these structures.

2.5.3 Combining cues

The sensorimotor circuits for path integration, visual homing and visual route following are thought to operate independently and in parallel (Sun et al., 2019). In this section, the requirement to combine these and other cues into a single behavioural output is briefly considered. In (Wehner, 2009), the concept of the insect navigation toolkit was conceived as a means of exploring how different cues might interact. This insect navigation functional decomposition (included in fig. 2.14) provides a useful model for the analysis of possible cue integration schemes. Note that there is no feedback from the cue combination stage to the independent behaviour blocks. This reflects the current belief that behavioural circuits generate independent estimates of goal locations. Note also, there are potentially other behaviours that have not been included in this analysis, for example; systematic searching patterns, local vectors and beaconing.

2.5.3.1 Behavioural experiments

Attempts to reverse engineer the navigational cue combination mechanisms have to date been most abundant and conclusive in cue conflict behavioural experiments performed on desert ants. This process involves artificially placing the path integration system in a state of disagreement with the visual homing system. Existing paradigms include: displacing ants further along their routes such that they would be expected to underestimate their return journey with pure path integration (Collett and Collett, 2009); using a polarisation filter to rotate their path integration frame of reference (Reid et al., 2011); displacements from a feeder site to an unfamiliar location (Narendra, 2007); and artificially manipulating their visual panoramas at feeder sites (Legge et al., 2014). The results from this list converge on a broad consensus that: 1) cues operate independently (i.e. familiar visual panoramas or landmarks do not alter the path integration state); 2) Heading cues are combined to form a weighted compromise (Buehlmann et al., 2020). On this latter point, it has been suggested that cues might be combined in an *optimal* manner (Wystrach et al., 2015). A recent simulation model that followed this approach could account for some of the behavioural evidence outlined in this paragraph (Sun et al., 2019).

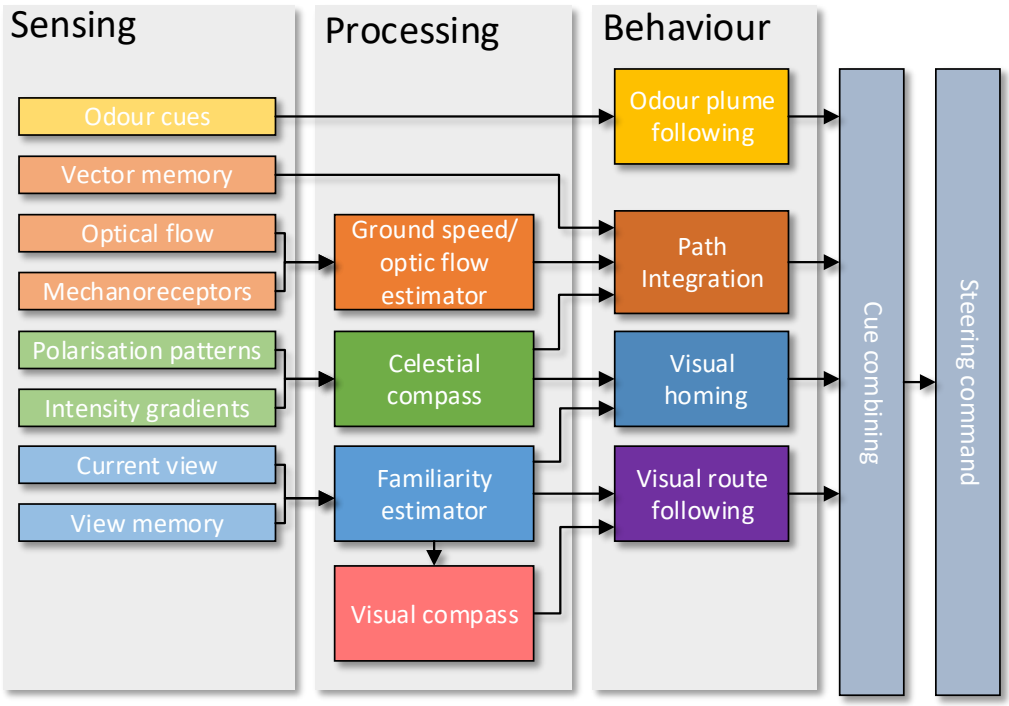


Figure 2.14: Flying insect navigation toolkit, based on the concept defined in (Sun et al., 2019). A simplified mapping between the parallel streams of sensing, processing and output for the independent behaviours of insect navigation.

The situation is less clear in flying Hymenoptera, which forage in larger areas than desert ants. As reviewed in section 2.5.2, honey bees that are displaced to novel locations perform long searching flights before switching to a direct route home beyond visual contact with their nest (Degen et al., 2016). This suggests that either visual navigation occurs over long ranges (100m+), or that there is a mechanism for associating vector memory with view memory. A model for associating view memories with homeward vectors is presented in (Cruse and Wehner, 2011). However, bees that have their celestial compass shifted by anaesthetisation, do not appear to shift their sense of homeward direction when in the location of a visual memory (Cheeseman et al., 2014). This is a somewhat controversial experiment though, due to a failure to rule out alternative possible homing mechanisms (Cheung et al., 2014), such as the work presented in Chapter 5.

Honey bees trained to a feeder in a tunnel have a lower variance in their search space when a prominent visual cue is encountered en route (Srinivasan et al., 1997). This caused the author to speculate that the visual cue could be used to improve the

accuracy of the path integration circuit (e.g. by splitting a journey into several chunks). A follow up experiment that distinguished between *isolated* landmarks (akin to a tree in a field) and *boundary* landmarks (analogous to the transition between forest cover and a meadow) showed that the search patterns are only more accurate for isolated landmarks (Collett et al., 2002). But the authors in (Collett and Collett, 2009) conclude that the close proximity of the landmark may have influenced the results discussed in this paragraph, either due to visual homing snapshot mechanisms or a *local vector* effect.

Honey bees trained to a feeder across a field with prominent artificially placed landmarks appear to centre their searches around the actual distance of the feeder when the landmarks are shifted closer or further from the nest and the feeder is removed (Menzel et al., 2010). This indicates that path integration is the dominant cue when conflicted with visual homing on outbound foraging routes, although the bees do show signs of confusion at the locations where the landmarks have been shifted from and a greater variance of search space at the trained feeder site. This suggests that like desert ants, cues may be combined into a single weighted output. Honey bees that have their visual panorama shifted while they are at a feeder site use the visual panorama to guide their initial flight path (Towne et al., 2017). The fact that the path integrator is overridden in this case supports the notion that visual homing cues are strongest near the site of interest.

Odour cue following has been predominantly studied in foraging honey bees seeking a particular feeder site. A honey bee that has been trained to a feeder site with a particular odour will sometimes deviate its course by relatively small angles in the presence of off-route equivalent odour sources (Menzel and Greggers, 2013). Flight direction is however mainly guided by visual cues. Equivalent tests towards the nest are not possible in social animals because of the persistence of the odour signal. Tinbergen showed that close to their nests, solitary wasps ignore artificial odour cues, and they can be trained to fly directly into another wasps nest by shifting the conspicuous landmarks around their nest to equivalent positions about the new location (Tinbergen, 1972). This suggests an exclusive reliance on visual cues when approaching the nest in these animals.

2.6 State-of-the-art flying insect navigation biorobots

While biologically inspired flapping wing MAVs have recently received considerable research attention (Jafferis et al., 2019), these platforms are not mature enough to support the payloads required for biomimetic navigation work. As summarised in fig. 2.15, neuroethologists studying visual flying insects have usually opted to use quadcopters to embody their behavioural models. Quadcopters can fulfill insect-like flight capabilities such as hovering and holonomic translations, while also being able to carry a vision system and the required test instrumentation. Recent developments in MAV technology have enabled the development of a number of real-world aerial platforms that serve to evaluate biological hypotheses. While the availability of commercially available MAV units is growing in number and sophistication, this survey indicates that researchers tend to develop their own platform using off-the-shelf avionics systems.

2.6.1 Corridor following biorobotic studies

According to Serres and Ruffier (2017), the first implementation of a free flying biorobotic quadrotor configuration was outlined in 2009 (Conroy et al., 2009) (shown in fig. 2.15d). A catadioptric camera was mounted atop the platform, providing a wide-field panoramic view. This was used to measure panoramic lateral optic flow estimates and thereby infer the location of obstacles relative to the heading of the aircraft. It was demonstrated that this platform could navigate along corridors, analogous to scaled up version of those outlined in section 2.4.1.4.

In (Keshavan et al., 2015, 2014), Keshevan et al. present a biologically inspired quadcopter built using a DJI flamewheel frame and selected avionics (fig. 2.15e). Of note, eight optic flow sensors were placed in an array with 45° offsets in azimuthal direction. This array was used to measure the optic flow in their respective viewing directions enabling a detection of corridor proximity at those aspects. This platform was also shown to be robust against constricting corridor scenarios and gaps in the corridor wall.

While not strictly an aerial platform, the LORA hovercraft robot presented in fig. 2.15b could move in a 2D holonomic fashion (Serres et al., 2008). This platform also demonstrated corridor following abilities using a *dual optic flow regulation* scheme. A pair of lateral and forward facing optic flow sensors provided a control signal for the control system governing the lateral position and forward velocity respectively. This

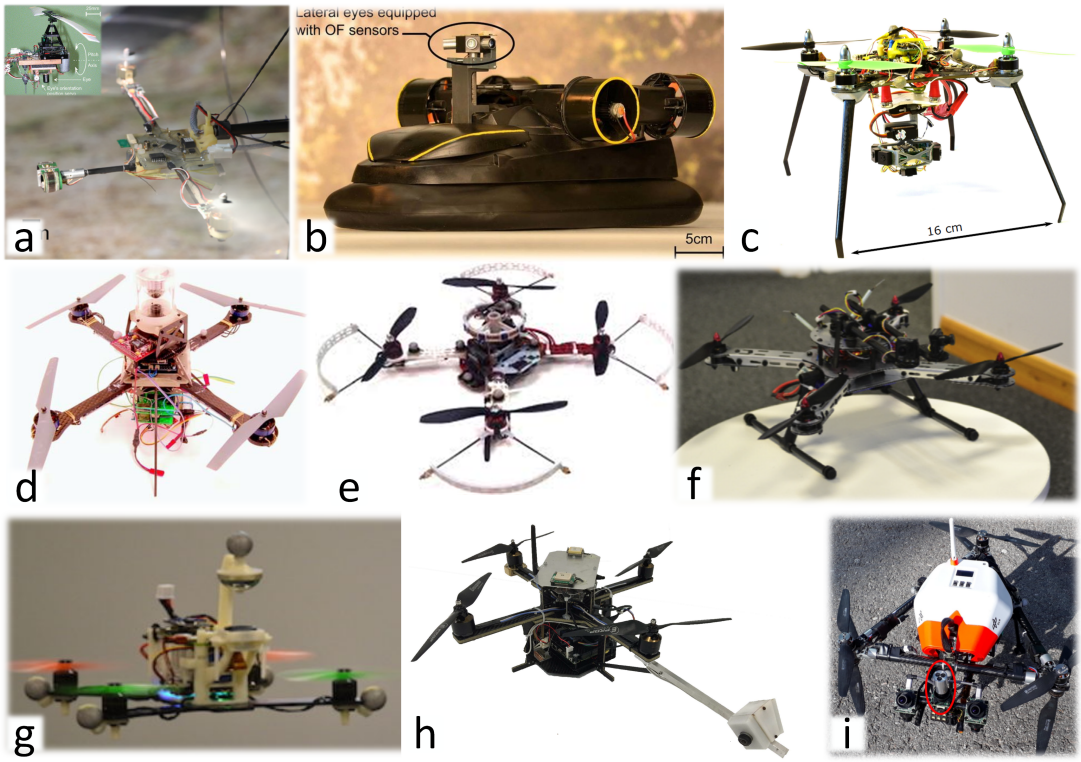


Figure 2.15: Montage of previous biorobotic navigation MAVs. Descriptions of each system are provided in the main text body (section 2.6). Further details about each system can also be found in its corresponding publication: **a** (Expert and Ruffier, 2015) CC BY 3.0; inset: (Ruffier and Franceschini, 2005) with permission from Elsevier; **b** (Roubieu et al., 2012) CC BY 4.0; **c** (Vanhoutte et al., 2017); **d** (Conroy et al., 2009) Adapted by permission from Springer Nature; **e** (Keshavan et al., 2014); **f** (Sabo et al., 2017) CC BY 4.0; **g** (Cope et al., 2019) Adapted by permission from Springer Nature; **h** (Strydom et al., 2016) CC BY 4.0; **i** (Steidle et al., 2019) CC BY 4.0;

setup accurately mimicked the trajectories of honey bees flying through various tunnel configurations, the vehicle slowed down when a constriction or barrier appeared, thus enabling better centering performance and the ability to navigate tight corners.

Another corridor centering robot was outlined in (Sabo et al., 2016). This used a commercially available product indicating that some off-the-shelf the solutions have software development kits that allow them to be re-purposed for biomimetic experiments without the need to engineer new systems. However, the authors updated their platform in (Sabo et al., 2017) to a bespoke solution (fig. 2.15f). This included a customised camera configuration with sampling characteristics commensurate with a honey bee's vision system and optical flow processing that could operate with this arrangement. This resulted in improved corridor centering performance tests versus (Sabo et al., 2016). A smaller variant of this lineup, 'minibee', was reduced to a weight of 200g in (Cope et al., 2019) (fig. 2.15g). Although this strict weight regime precluded onboard calculations, a WiFi link enabled the off-board processing of visual data, so that resource intensive neural model computations could still support online real-world implementations.

2.6.2 Terrain following biorobotic studies

Prior to the widespread availability of MAV components, a rotorcraft, consisting of a single propeller supported by an articulated boom was developed. The craft was mounted on the end of the boom with a pivot joint so that it could pitch forward, thus creating forward thrust. An optic flow sensor was mounted on the boom and directed towards the ground. The sensor's optic flow signal fed into an optic flow regulator control system to keep the rotorcraft at a fixed height above ramped terrain that was swept over according to the arc of the boom (Ruffier and Franceschini, 2005). A more recent, dual propeller, implementation of this rotorcraft concept, "Beerotor" (Expert and Ruffier, 2015)(fig. 2.15a), facilitated the testing of the ability of dual optical flow regulation techniques to allow simultaneous control of horizontal speed and altitude in the presence of terrain with steep slopes. Both of these systems provided reliable altitude control in this constrained environment based on optical flow measurements only (i.e. no inertial sensors were used). Custom built bio-inspired optical flow sensors were also tested with this arrangement.

A wall following quadcopter was presented in (Vanhoutte et al., 2017)(fig. 2.15c). This platform also made use of an array of eight dedicated optic flow sensors which

were mounted on a servo driven gimbal attachment so that the sensors were always level with the horizon. The aircraft could follow a textured planar surface at a fixed angle.

2.6.3 Robots with insect-inspired heading sensors

A number of biologically inspired vision-based heading sensors have been deployed on robotic platforms, here an allothetic and an idiothetic example are presented. In (Steidle et al., 2019) (fig. 2.15i), skyward *degree of polarisation* patterns are measured with a camera sensor to infer the location of the sun and thus a global heading estimate. The camera sensor receives light from three separate wide angled lenses that have polarisation filters orientated with 60° offsets. By mapping the pixel viewing angles for each lens, the relative intensities can be measured and therefore the angle of polarisation can be calculated for many viewing angles. An optical flow rotational rate sensor was recently deployed in (Skelton et al., 2019) using biologically inspired (EMD like) visual processing. This was deployed on embedded hardware and proved to be more parsimonious than state-of-the-art dense optical flow methods.

2.6.4 Multi-purpose biorobots

While most robotic platforms have been developed with a particular study in mind, one quadcopter platform built at the University of Queensland has served a number of projects as summarised in (Strydom et al., 2016)(fig. 2.15h). The platform has a bio-inspired vision system repurposed from (Thurrowgood et al., 2014) that features two wide angled cameras mounted on a beam extending out from the airframe so that body occlusions are minimised, Each camera has a similar view configuration as the left and right eye of a honey bee. Of particular relevance to the current work, the platform hosted a path integration system with optic flow visual input, although this used an engineering approach block matching algorithm as described in (Thurrowgood et al., 2014), with height estimations extracted from a stereo camera configuration.

In (Jouir et al., 2018), the path integration system was extended with the inclusion of skyline extraction for heading information and a block matching optic flow algorithm with software rotation compensation. This setup was used to perform some square shaped missions with a total length of 100m. A closure error of 1.73 ± 0.56m is reported. A second method for *map free* navigating by snapshots was also implemented in (Denuelle et al., 2015) and (Denuelle and Srinivasan, 2016). The approach was to

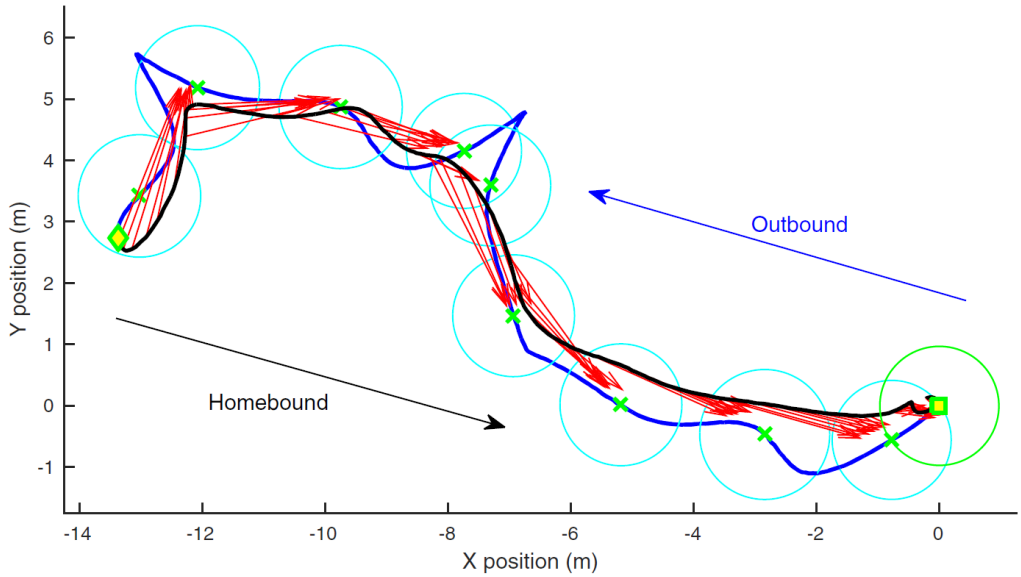


Figure 2.16: Route following mission from (Denuelle and Srinivasan, 2016). The red line shows the homeward route with blue circles showing the catchment area of each snapshot. The magenta arrows show the estimated vector to the next waypoint

record snapshots along a route and use each snapshot as a waypoint. On an inbound route, when a waypoint is reached, the vector to the next waypoint is loaded and the MAV follows this local vector where it calculates the deviation between its current location and the snapshot location. The block matching optical flow algorithm used in the path integration system is used to compare the current image with the reference image captured on the outbound route, thus achieving a local drift free localisation method.

2.6.5 Visual route following in biorobotics

The route following approach has been investigated from an aerial context in simulation Differt and Stürzl (2020); Müller et al. (2017), and with a robotic gantry Philippides et al. (2016). These investigations have shown that the conventional route following approach can be shifted to the aerial domain. In Philippides et al. (2016), the author demonstrated that snapshots taken at a given height have a degree of robustness to height variation. The authors in Differt and Stürzl (2020) demonstrated that a helical flight path could be used to improve tolerance to snapshots taken at different heights.

2.7 Outlook

Central place foraging insects are able to find their way in extended local habitats. This ability is largely supported by a path integration circuit that makes use of a visually derived sense of egomotion and an allothetic heading cue. While the full neural pathway for the sky compass heading cue is now well established (Honkanen et al., 2019), the motion detection pathway has yet to be described, even at the level of information flow. The range parameter encoded in a honey bee's waggle dance provides evidence that the raw unit of a bee's social odometer is related to experienced optic flow. However, it is also apparent that this parameter is a different entity to that used in their path integration circuit (Chatterjee et al., 2019; Dacke and Srinivasan, 2008).

In section 2.6 a review of several aerial platforms that were developed in order to evaluate models of the flight control behaviours of insects. These platforms were instrumental in motivating recent motion perception experiments (Lecoeur et al., 2019; Linander et al., 2015b; Portelli et al., 2011a), which in turn have helped ethologists to establish the fact that flight control is served by multiple motion perception pathways (Lecoeur et al., 2019). It is believed that at least one of these pathways also serves the path integration circuit, whether a fixed circuit or an integration of several areas is used for this task is unknown. It is perhaps surprising therefore that there has been little attempt to model the sensory circuits in combination with a path integration circuit. One platform (Strydom et al., 2016) sought to translate the path integration procedure into a biologically inspired application but no biologically relevant questions were addressed in this work.

Central place foraging insects use a combination of cues when relocating their nests. While the path integration circuit is apparently crucial when these insects perform their learning flights and when they are at extended distances from their nest, visual homing is likely to play an increasingly dominant role in the latter stages of inbound flights. The notional drift rate of a biological path integration circuit is yet to be established, but it is reasonable to assume that the visual homing circuit can operate from at least the maximum likely path integration error that the species-of-interest may encounter in their lifetime (i.e. the maximum distance a returning insect would be from its goal under exclusively path integration guidance). If an estimate of the efficacy of the path integration circuit can be established then a better understanding of the requirements of a visual homing system could be achieved.

The modelling of terrestrial visual navigation techniques has been a popular re-

search activity in recent years, with a particular focus on visual route following. The vast majority of this work has been conducted in simulation (Ardin et al., 2016; Baddeley et al., 2012; Le Möel and Wystrach, 2020). While such environments provide a convenient framework for deploying proof-of-concepts, current solutions do not capture the rich detail of the natural world. In particular, there is a lack of: complex textured scenes; non-rigid body motion; dynamic lighting conditions; vision sensor noise; and environmental forces. By contrast, robotic implementations can operate with the same visual signals and physical disturbances as biological entities.

To date, visual route following models that emulate flying insects have been tested exclusively in simulation (Differt and Stürzl, 2020; Müller et al., 2017) or on a robotic gantry Philippides et al. (2016). Interestingly, these implementations have reused the visual sampling strategies of ground based solutions. Previous work has shown that flying animals often exploit linear ground based features Lipp et al. (2004a); Menzel (2019). Deploying an anatomically constrained route following procedure on a real-world aerial platform would provide an ideal system for investigating whether this system can track these ground based features, and whether this view sampling strategy performs better than holistic or horizontal configurations.

In contrast to desert ants, the manner in which bees and wasps combine path integration and visual memory cues is unclear. Some accounts claim that honey bees possess cognitive maps whereby visual memories are stored in a flexible spatial representation (Cheeseman et al., 2014; Menzel et al., 2005, 2011). While others dispute this claim and suggest that the behaviours can be explained with independent path integration and view memory systems (Cheung et al., 2014; Cruse and Wehner, 2011). Given the existence of navigation models for visual homing and path integration in the terrestrial domain. An application of these models on an aerial platform could provide new insights into the respective roles of path integration and visual navigation in flying insects.

Chapter 3

An aerial platform for testing insect inspired navigation algorithms

3.1 Introduction

In the previous chapter, the benefits of evaluating insect inspired path integration and visual navigation models with a physical aerial biorobot was established. A survey of state-of-the-art flying insect navigation biorobotic platforms (see section 2.6) determined that the majority of previous studies have adopted a quadcopter aircraft for the embodiment of behavioural models. Amongst this literature it is widely stated that quadcopters are a good analogue for flying insects because they are capable of; agile flight, the ability to hover, vertical takeoff and landing (VTOL), full holonomic motion and a relatively low translational speed (in aviation terms) (Cope et al., 2019; Sabo et al., 2016; Strydom et al., 2016).

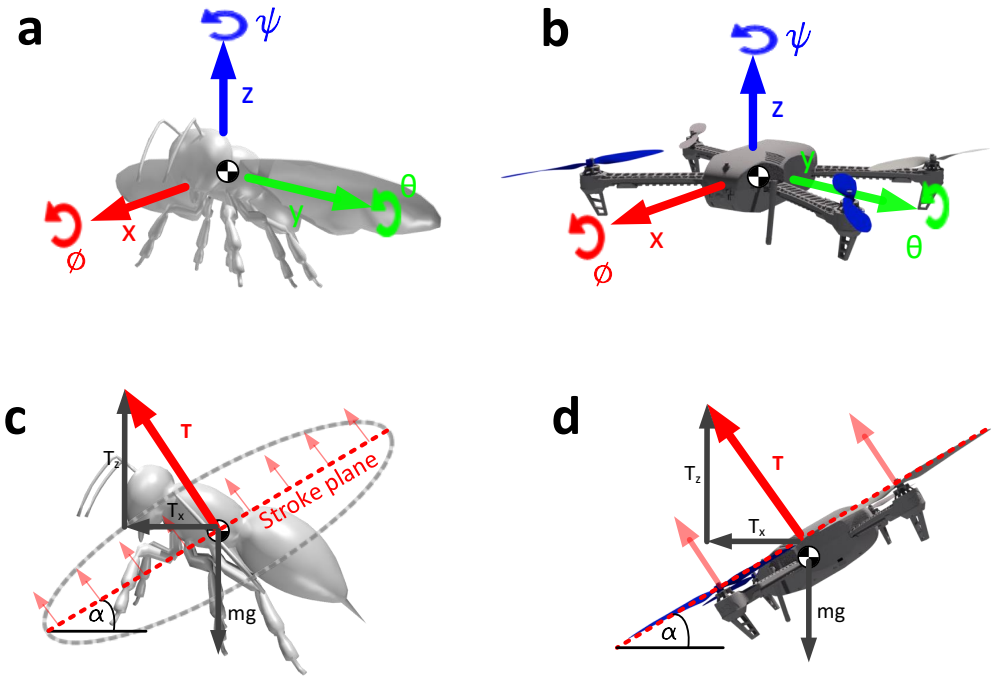
In this chapter the development of a new quadcopter biorobot is presented. The suitability of a quadcopter as a biologically relevant biorobot is first examined in section 3.2. The requirements for a biorobot that will test path integration and visual navigation behavioural models are then considered in section 3.3. The development of the biorobot is then outlined, this is divided into hardware (section 3.4) and software (section 3.5) sections. Finally, the biorobot is evaluated against the design requirements and the results are presented in section 3.7.

The number of commercially available MAV platforms and systems has substantially increased in recent years. In this chapter an attempt at combining this technology into a modular hardware and software platform is described. The resultant system will serve as a platform for the remainder of this thesis.

3.2 Quadcopters as flying insect biorobots

3.2.1 Kinematics

Quadcopters and flying insects generate thrust by principally different means. Flying insects achieve thrust via the leading edge vortex that forms on their pivoting wings and at least three other aerodynamic principles that are significant in low Reynolds number bodies (Dickinson et al., 1999)[1]. Quadcopters by contrast, generate thrust via four independently controlled motorised propellers. Despite these propulsion system differences, both quadcopters and flying insects are commonly modelled as 6-degree of freedom (DOF) rigid body mechanical system. In flying insects, which are certainly not rigid bodies, this modelling assumption is particularly common for the analysis of their flight control, where the *helicopter model* has been previously adopted as a kinematic basis (Badrya et al., 2017; Dickinson and Muijres, 2016). An overview of this kinematic comparison is presented in fig. 3.1, these free body diagrams serve to outline the basis of the biorobot's biological relevance.



[1]There is a significant variation in the mechanics of insect flight depending on the insect's size, wing structure and ecology (Bomphrey et al., 2010). In this work all data sources are restricted to fruit fly and bee studies unless otherwise specified.

Figure 3.1: Free body diagram of a flying insect (**a**) and a quadcopter (**b**) modelled as 6-DOF rigid bodies. The body fixed coordinate system is aligned with the centre of mass which is proximal to the centre of pressure. A right hand rule Tait-Bryan coordinate system is adopted with forces resolved into xyz directions and moments resolved about roll (ϕ) pitch (θ) and yaw (ψ) axes. Note that this is a local tangent plane coordinate (LTPC) coordinate system and therefore the coordinate system itself does not track the pitch and roll of the agent. Viewed from the side, a comparison can be made between the flying insect (**c**) and quadcopter (**d**) propulsion systems (Badrya et al., 2017). Both systems need to generate enough thrust (T) to overcome their weight but can also generate thrust and therefore motion in the horizontal plane. The quadcopter achieves this by pitching the whole aircraft whereas the flying insect achieves this by pitching the stroke plane of its wings, relative to gravity. In both cases the angle of attack, α, can be used to calculate the horizontal component of force (T_x).

3.2.2 Control

Free flying quadcopters are subject to aerodynamic disturbances, in order to trim a quadcopter to a desired attitude, the aircraft must be powerful and agile enough to overcome these disturbances. Quadcopter stability control is usually subdivided into three modes; attitude, thrust and heading. Each of these is achieved by modulating the speed and thereby thrust generated by the aircraft's four motors. Note that the attitude controller itself is divided into roll and pitch resulting in four independent control channels. The effect of modulating the relative angular velocity of the four motors is reviewed in fig. 3.2. Stability control depends on closed loop feedback control of the four control channels at a sufficiently high rate and with sufficient thrust and/or response time to overcome the disturbances. Note that each control channel is served by the same four motors. Provided that the quadcopter has a sufficient thrust margin, the control action from each mode can be superposed via a control mixer. That is, the motor setpoints for each control area are combined by addition.

Beyond stability control, the control channels can be interfaced with a unified trajectory controller with the purpose of moving the quadcopter around its environment in a desired fashion. Stability and trajectory control is typically achieved using an integrated, cascaded control loop structure such as the one outlined in fig. 3.3. For each control channel, the position controller feeds into the velocity controller which propagates into the stability controller. A separate PID control mechanism is implemented

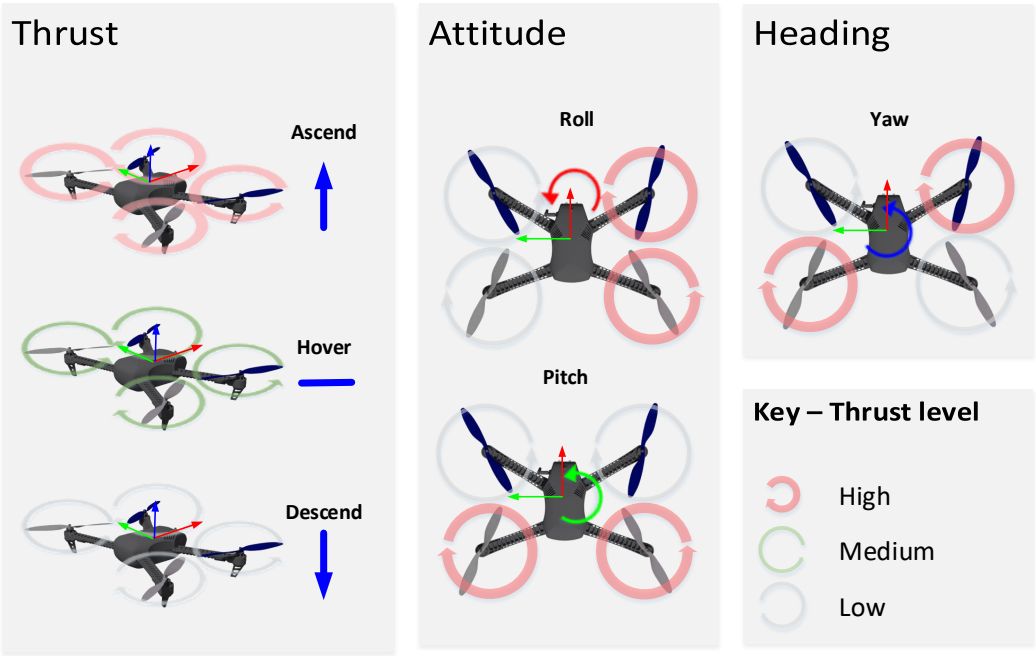


Figure 3.2: Per axis control of an x-configuration quadcopter. Arrows at each motor show the direction of rotation and magnitude of thrust generated by that motor. **Left** - hovering is achieved by all four motors generating just enough thrust to overcome gravity whereas ascending and descending is achieved by generating more or less thrust than the aircraft's weight respectively. **Middle** - the aircraft can roll or pitch if the a pair of motors at one end of the aircraft generate more net thrust than the opposing motor pair. **Right** - in addition to lift force, quadcopter motors also generate a tangential force. In order to prevent this force from spinning the aircraft, two of the motors spin clockwise while the other two spin anticlockwise. Motors that are spinning in the same direction are mounted diagonally opposite. Yaw motion is generated by increasing the thrust generated by one of the diagonal pairs while reducing the thrust in the other pair by the same amount.

for each level of the cascade. In order to deal with the nonlinearities associated with quadcopter attitude control, a conversion block is usually placed between the trajectory and attitude control blocks (Gudmundsson et al., 2018; Tal and Karaman, 2018). When extreme manoeuvrability is required, model based control can improve control performance further but this requires an accurate model of the aircraft and significant software development (Chikasha and Dube, 2017).

Recent work has demonstrated that manoeuvres conducted by free flying fruit flies

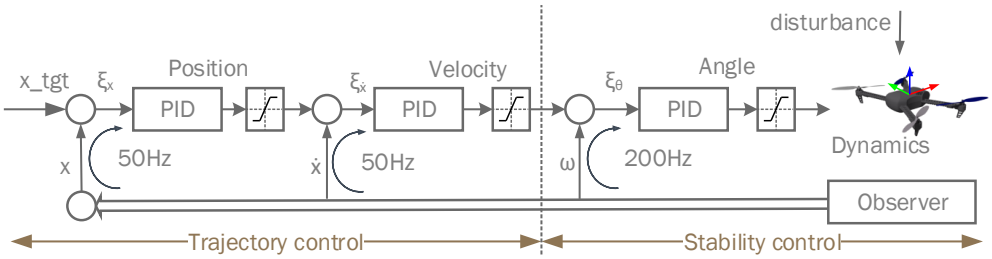


Figure 3.3: Cascade control system for an attitude control axis (roll or pitch). The desired attitude angle is governed by a velocity controller which, if required, is in turn governed by a position control loop. Each loop has a PID controller followed by a saturation block. The latter facilitates the specification of soft system limits, e.g. maximum pitch angle or forward velocity. The inner attitude loop is required to operate at a high frequency (usually 200+ Hz) to reject disturbances, by contrast the outer loops can usually operate considerably slower rates without any degradation in performance. Here ζ indicates the error at each level, $x, \dot{x}, \omega$ are the linear position, linear velocity and angular rate properties respectively, x_tgt is the current position setpoint.

can also be categorised into four control modes; throttle, roll, pitch and yaw (Dickinson and Muijres, 2016; Lindsay et al., 2017). While it is beyond the scope of this work to explain the mechanics of flapping flight, the interested reader will find a striking resemblance between figure three of (Dickinson and Muijres, 2016) and the manoeuvres outlined in fig. 3.2.

3.2.3 View stabilisation

A noteworthy comparison between quadcopters used to film videos and flying insects is the convergence on the mechanical stabilisation of the visual sensor as illustrated in fig. 3.4. A camera that is fixed to the body of a quadcopter is an example of an *underactuated* system. That is, in order to translate the camera in a particular horizontal direction, the quadcopter must pitch or roll to achieve this motion, thus also inducing a potentially undesirable view angle change in the camera view. In order to counteract this, multirotors with a filming application usually mount the camera on a motorised gimbal. This provides additional degrees of freedom to the system and enables pure camera translations relative to the global coordinate frame. Flying insects also have a reflex to shift their body location during flight depending on their airspeed (Taylor

et al., 2013). Flying insects are known to stabilise their head during flight, such that the roll and pitch remain at constant angle with respect to gravity (Viollet and Zeil, 2013). It therefore would seem important to include mechanical stabilisation to the vision system of the biorobot.

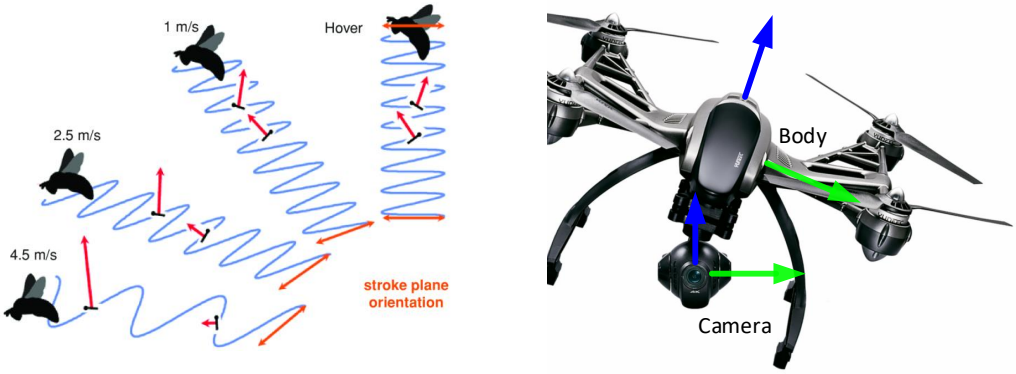


Figure 3.4: **Left:** Side view of a bee during flight indicates that the angle of attack of a bee's body varies during flight as a function of its forward speed but the head remains at a constant angle with respect to gravity at all speeds - image from (Shyy et al., 2013). **Right:** The camera of a Yuneec Typhoon 4K filming quadcopter remains level while the aircraft's body is rolling. In many applications, mechanical stabilisation is required for smooth, non-blurry video. Image subject to copyright, used here with permission.

3.2.4 Limitations

It is worth considering the differences between a quadcopter biorobot and its biological target. In addition to the propulsion mechanism differences, there is a stark variation between the scale and mass of the two entities. A large bee might reach 3cm in length and weigh no more than 2g. A quadcopter with the required payload capacity is a minimum of 40cm from motor to motor and weighs around 2kg. Therefore, there is at least a factor of 10 difference in scales and a factor of 1000 in mass. Aerodynamic properties, such as the Reynolds number of a body are known not to scale linearly with size (Badrya et al., 2017). Modelling limitations due to scale, mass and aerodynamic are considered below:

1. The inertia of an insect is considerably lower than that of a quadcopter. Wind gusts are the primary source of disturbance in the natural world. The acceleration

experienced by an insect due to a particular wind force is considerably higher than the acceleration that would act on a quadcopter (Jakobi et al., 2018). However, an insect subject to such a disturbance is able to compensate for this within 0.2s and with a course offset of less than 53mm (Jakobi et al., 2018). The time constant of a well tuned MAV reacting to a control input is around 0.1s, therefore there is some equivalence between the flight dynamics of the two systems.

2. The scale difference may preclude comparisons with biological datasets. For example, some work was trialled on datasets relating to the learning flights of wasps in a natural sceneStürzl et al. (2016) and a similar process was conducted for bumble bees Philippides et al. (2011). Because these flights turned out to be confined to less than a 50cm cube, MAVs were an unsuitable platform for deploying this kind of model.

3. The vision system is automatically close to the centre of pressure for a flying insect, therefore detected motion in the visual frame corresponds closely to the actual motion of the insect. By contrast, a camera mounted on the extremities of a MAV is shifted relative to the centre of pressure and therefore motion detected at the vision sensor may deviate from the aircraft's motion. Depending on the geometry, a transformation could resolve this problem but this structural divergence may be undesirable for some investigations, ascertaining the optimal viewing arrangement for example.

4. The frame rate of a conventional camera is lower than the flicker fusion of an insects visual system. This could be compensated by reducing the rate of ego-motion but the effects of external motion cannot be mitigated in the same way. Wind speed for example, is not directly controllable.

In summary, quadcopters have sufficient degrees of freedom to recapitulate trajectories of flying insects, but there are caveats that need to be considered when using a quadcopter to model insect behaviour. The flight dynamics of bees during foraging behaviour has not been well described (Sabo et al., 2017). Since the overall goal is to deploy already abstracted behavioural models, this is not considered a prerequisite here.

3.3 Biorobot functional requirements

3.3.1 Behavioural model embodiment requirements

As motivated in Chapter 2, there are two navigation behaviours that the biorobot is required to serve; path integration and visual navigation. The primary input and output interfaces for these sensorimotor circuits are considered in fig. 3.5. It is apparent that both models have similar embodiment interface requirements at this level of abstraction. A visual stream is the primary input and four axes of motion control are required relative to a four-DOF inertial frame (x,y,z,ψ). Where the x,y and ψ axes require velocity control and z requires position control. Fortunately, this control arrangement is readily achieved with quadcopter trajectory controllers using the principles detailed in section 3.2.2.

The visual navigation behaviours of flying insects are thought to operate over several kilometers, it is therefore important to demonstrate the ability of the behavioural models to operate well over the ranges that these models have currently been tested using real world platforms (approximately 10m (Kodzhabashev and Mangan, 2015; Stone et al., 2017)).

Existing insect visual navigation models are served by wide angle holistic panoramic views. Such views enable terrestrial compass functionality which has become a fundamental aspect of land-based models. One element of the modelling work to be performed here is to establish what portion(s) of an insect's view is salient to the visual navigation task. Therefore a fundamental requirement of the vision system is to be able to flexibly sample from different viewing perspectives.

3.3.2 Experimental requirements

In order to evaluate the homing ability of behavioural navigation algorithms deployed on the biorobot, a number of operational requirements must be considered. In particular, it is necessary for the aircraft to travel to the test starting location, it is necessary to log data relating to the aircraft's state throughout and the aircraft must be able to continuously operate for the duration of an entire test.

In previous biorobotic navigation modelling work, the outbound route is a required capability. The biorobot must be capable of enacting prescribed trajectories to serve this purpose. Preliminary tests also indicated that a safe landing site helps to prolong

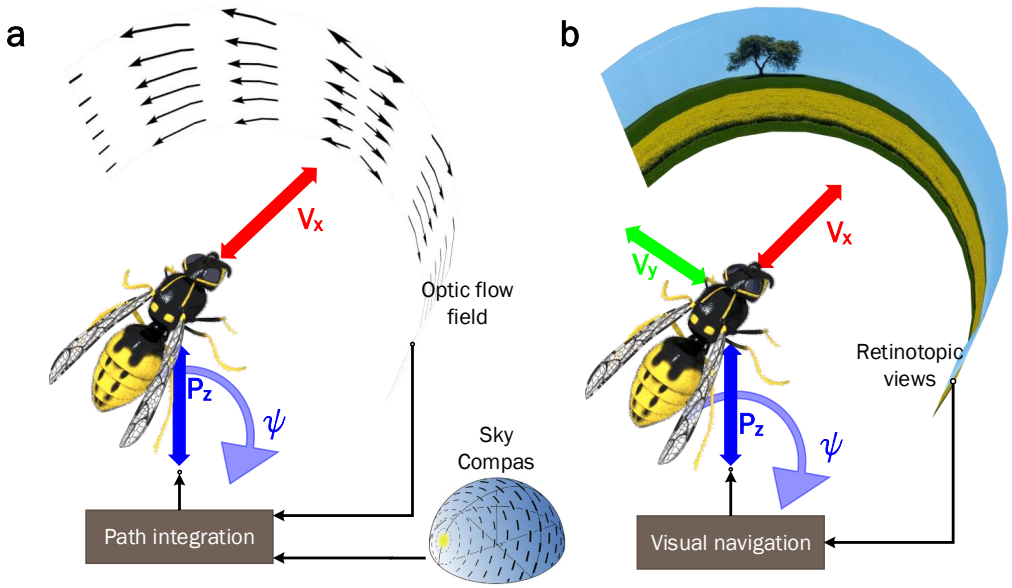


Figure 3.5: The anticipated sensorimotor input and output interfaces for a navigational biorobot applied to path integration (**a**) and visual homing (**b**). The insect's visual input is a widefield, frontal view and is shown here as a curved screen. The main control outputs of existing models are forward velocity (V_x) and a steering rate (ψ_z). In an aerial implementation, the biorobot is required to control its height relative to the ground as a form of position control in the z-direction (P_z). In visual homing it is believed that a transverse component of velocity (V_y) may have a role to play in image matching whereas for path integration the V_y setpoint is probably zero. Roll and pitch head angle setpoints are assumed to be set to zero and actively rejected for both behaviours. Photograph of landscape in **b** by Arno Hoyer, CC2.

the life of the vehicle and this site may not be commensurate with a useful test start location. The ability to fly to a prescribed starting location in geographic coordinates is therefore required. In order to support flying to an offset location and performing an outbound and inbound route, both in the order of 100m, a target flight time of 10 minutes is set for the biorobot. The behavioural models must be able to run onboard, given the intended outdoor nature and range of the biorobots operation.

The success of an embodied homing algorithm may be evaluated by measuring how close the agent is to the starting location at the end of a given trial. Beyond this, the work here seeks to understand how well navigational algorithms function in relation to different environmental conditions. Therefore, it is desirable to log the state of the

aircraft and behavioural model (where it estimates it is) in relation to a ground truth global coordinate system. The required rate and accuracy of this logging is somewhat dependent on the model and associated test. Furthermore the analysis requirements are not necessarily known a priori. In accordance with the current capabilities of MAV related technology a minimum requirement of $\pm$10cm accuracy in the XY plane and logging frequency of 10Hz is set as a specification.

3.4 Hardware

In this section the hardware component selection is justified and manufactured parts are detailed. From the outset, three principally different approaches for developing an autonomous MAV hardware configuration were identified:

1. **Off-the-shelf** solutions. Some MAV manufacturers such as Parrot[2] or DJI[3] offer software development kits for programming their products. No hardware or system architecture development is required and these systems usually have a good flight time. However, the user is usually limited to the manufacturer supplied sensing apparatus. Manufacturers provide an application programming interface (API) but the main code base is commercially sensitive and closed source which limits the scope of integrated development.

2. **Custom** solutions. Some research projects choose to custom build all components specific to the application. This enables a product to be optimised against the project requirements. However, this approach requires considerable development effort and multi-disciplinary expertise.

3. **Customisable off-the-shelf (COTS) components** solutions. COTS components enable a compromise between the first two options. The designer creates a system with customisable hardware parts that meet the project requirements without having to reinvent parts which have existing solutions.

Of the three options, an off-the-shelf solution would have substantial development time benefits if a model with the correct attributes could be sourced. However, at the time of writing there were no platforms that had a sufficiently customisable mechanically stabilised gimbal system and option to mount accurate GPS systems. It was

[2]https://www.parrot.com/
[3]https://www.dji.com/

also apparent that there were many airframes and flight controllers that could form the basis of a solution without the need to design these from scratch, therefore, a design philosophy of using COTS components was adopted.

3.4.1 component selection

In this section the selected COTS component choices are outlined.

3.4.1.1 Airframe

A Lumenier QAV400 airframe with stock motors, electronic speed controllers (ESCs) and propellers was selected for the airframe. This aiframe has been developed for generating aerial footage, with very agile flight compared to other, typically larger, filming platforms. The airframe has two sections, the *dirty* section to which the motors are attached and the *clean* section, which is connected to the dirty section via silicone dampers in order to minimise motor vibration transfer. Since insect flight does not feature the high frequency vibrations caused by electronic motors and stiff airframes, the camera system is mounted on the clean part of the airframe.

3.4.1.2 Flight controller

In terms of flight control unit (FCU) hardware, the open source Pixhawk[4] was the only solution that offered all of the desired features without having to develop new hardware interfaces. The Pixhawk 2.1 unit was selected as this was the most up-to-date model at the time of manufacturing and it could be readily accommodated by the airframe. Various software stacks have been developed for Pixhawk hardware. In this work the PX4 stack was adopted on the basis that it is already ubiquitous in research and is supported by professional software engineers. PX4 features an *offboard* flight mode and an associated interface for aircraft setpoints to be sent from a separate control module or *companion computer*.

3.4.1.3 Companion computer

Although the Pixhawk 2.1 has some spare resources for developing onboard applications, there is only 2mb of flash memory and a single core ARM processor. A typical

[4] https://pixhawk.org/

arrangement for vision based autonomous missions is to package a companion computer on the airframe. The PX4 flight stack supports a flight state called *offboard mode* where control setpoints are provided by a serial communications link and vehicle state information is passed to the connected node. The main attributes sought in a companion computer are low weight, real-time performance and sufficient memory and processing speed for vision processing tasks. At the time of writing the Odroid XU4 was deemed to meet these requirements. It features a 5422ARM® Cortex™-A15 with eight cores which supports multi-process pooling operations which can significantly speed up vision processing pipelines. The whole unit weighs 40g, lighter than a mobile phone and with comparable processing power. While an external WiFi dongle is required for this board, this only adds a few extra grams and the extended range that an external aerial provides proved to be a useful operational feature.

3.4.1.4 Camera

A matrix vision bluefox 2.0[5] camera sensor was selected for this project. The unit supports different lenses yielding maximum flexibility in terms of view aspect. Throughout this thesis we used a lens with a 42° horizontal field of view which is much narrower than would be typically used to emulate insect vision. At 752x480, the camera has a relatively low resolution but offers a useful frame rate (90fps) and high dynamic range (110dB), this combination is ideal for modelling insect eyes. Note that the camera's resolution is typically downsized further to match insect eye resolutions. The camera is also only 10g in weight and the PCB only variant has a small form factor. Importantly, the camera operates with a global shutter thus eliminating a risk of rolling shutter distortions which are known to affect odometry pipelines in MAV applications (Vautherin et al., 2016). Since insect eyes process light in a parallel fashion, global shutter is arguably a better model for their eyes. Finally, this camera offers a hardware-triggered capture mode which enables frame capture on demand and the ability to accurately timestamp images.

3.4.1.5 Gimbal controller

A Basecam[6] SimpleBGC 32-bit board was sourced for the control of a custom built 2-axis gimbal. The unit drives brushless DC motors according to various control schemes. In this work, the controller was configured to maintain a fixed camera pitch and roll

[5] Manufacturers part number: mvBlueFOX-MLC200wC
[6] https://www.basecamelectronics.com/

orientation with respect to gravity. Here, constant roll and pitch setpoints were defined at the start of a given mission but in principle it is possible to connect the unit to an external control signal. The controller has two dedicated inertial measurement units (IMUs) that are used to monitor the angular position of the gimbal and the aircraft. Basecam supply an automated PID tuning software package for the convenient optimisation of the unit for a particular hardware configuration.

3.4.1.6 GNSS receiver

Conventional GNSS systems provide horizontal (latitude, longitude) measurements with an accuracy of $\pm 5m$. In order to accurately measure the location of the biorobot during operation it was necessary to use a real-time kinematic (RTK) GNSS system. The Here+ RTK unit with a reported horizontal accuracy of $\pm 6cm$ was selected for this purpose. In order to facilitate this, a radio link between the ground base GNSS receiver and roving GNSS receiver was required. A Holybro[7] 433Mhz telemetry kit was used for this purpose.

3.4.1.7 Lidar

While the selected GNSS provides high degree of horizontal plane accuracy, the height accuracy of the aircraft remains in the order of meters. In order to improve this situation a Garmin lidar lite (V3) unit was added to the system. This provides a height estimation performance of $\pm 10cm$. The lidar data was also interfaced with the PX4 state estimation module improving the height control performance of the aircraft and enabling *terrain following* functionality.

3.4.2 System architecture

The selected avionics are arranged according to the interfaces diagram in fig. 3.6. Custom wiring harnesses were built for all connections. Fortunately, all hardware drivers already existed for each of the peripherals. The main link between the companion computer and FCU was a serial line using the MAVLink protocol[8], this was configured to operate at a baud rate of 921600bps. A logic level converter was included in the harness to enable safe communication between the companion computer and Pixhawk serial communication ports which operate at 1.8V and 3.3V resepctively. Note that for

[7] http://www.holybro.com
[8] https://mavlink.io/en/

this work the gimbal controller was not connected to the FCU because the performance didn't require this (i.e. feed forward control was not required).

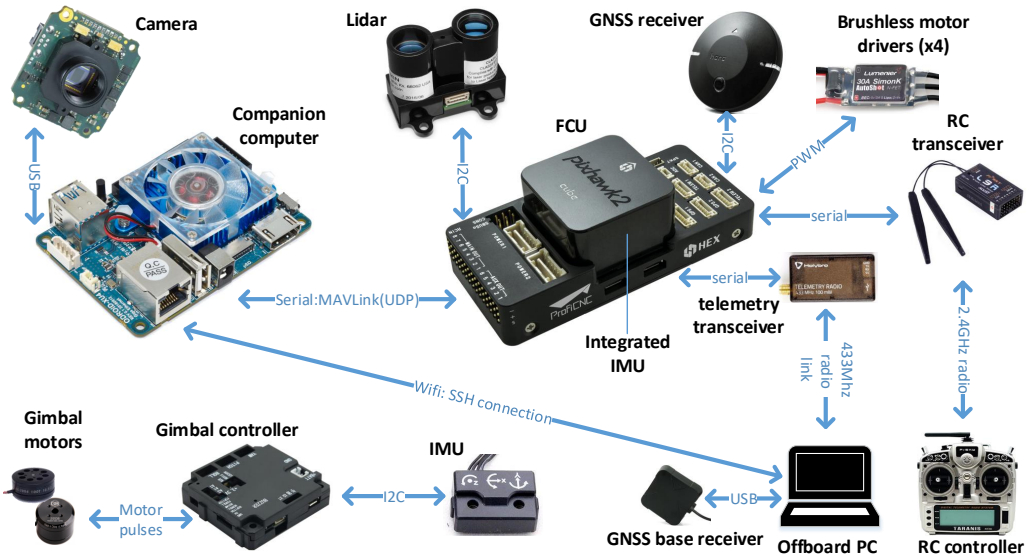


Figure 3.6: Biorobot avionic interface diagram.

3.4.3 Custom mechanical components

3.4.3.1 Airframe top plate

Due to the relatively small airframe and high number of avionic components to package, the airframe's top plate was replaced by a custom plate optimised for the mounting of the selected avionics (see fig. 3.7 for orientation). The new plate was machined from aluminium (alloy 6061-T6). This approach also facilitated the rigid attachment of sensors and their known pose offsets to the flight control unit IMU. This pose information can be used to improve state estimation accuracy. The GNSS and remote control (RC) receivers have special requirements in terms of exposing their aerials and minimising interference with other components that are sensitive to electromagnetic noise. Special mounting brackets were therefore developed in order to meet the recommended electromagnetic compatibility (EMC) requirements of each component. Additional spacers were also required to mount the control modules. All custom mechanical parts are shown in fig. 3.7. The computer aided design (CAD) files are publicly available Stankiewicz (2020), all parts can be manufactured using a 3-axis milling machine or a 3D printer as specified in the drawings. Mechanical fasteners are secured with M3

bolts, medium strength retaining compound should be applied to each bolted joint during construction to prevent fixings from loosening during flight.

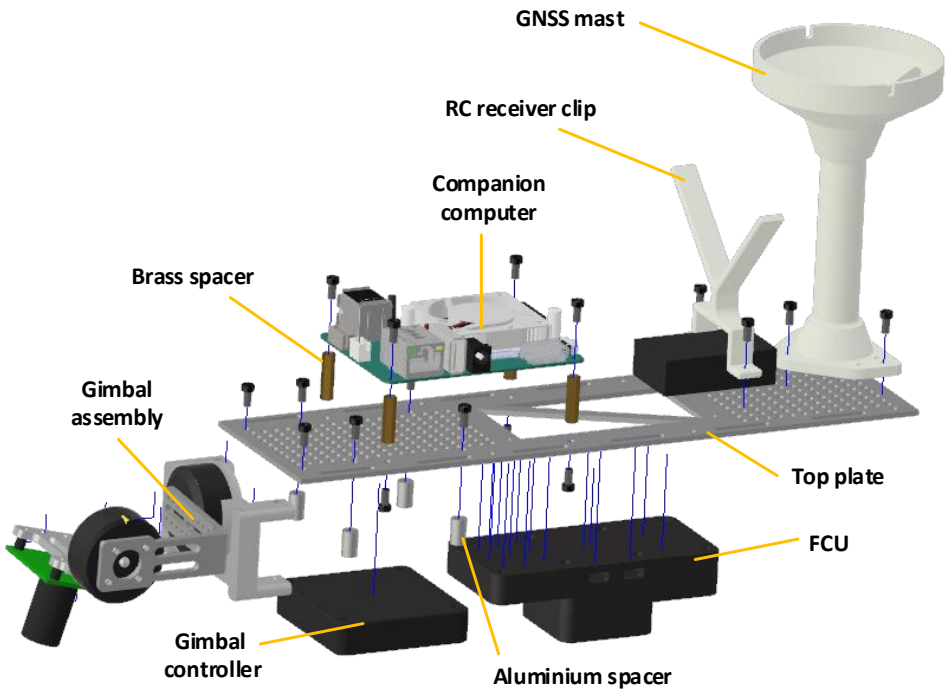


Figure 3.7: Exploded CAD diagram showing the custom top plate and associated custom hardware parts and key electronic control modules. Holes and cutouts were included to reduce weight and provide fastening points for electrical harnesses and to enable future hardware alterations to the avionics system.

3.4.3.2 Motorised camera gimbal

Motorised camera gimbals are widely used to stabilise video footage and minimise the effects of motion blur by actively restricting a camera's orientation to a predefined view direction. Camera gimbal hardware is now commonplace, however at the time of writing an off-the-shelf model that could be readily mounted on the airframe without adding excessive weight to the system was not available. Therefore, a custom 2-axis gimbal was developed for the biorobot.

The motorised gimbal comprises two rotating arms and an airframe attachment. Each mechanical interface is made about a brushless DC motor forming a pivot joint

as shown in fig. 3.8. These joints are exactly aligned with the aircraft's roll and pitch axes. The load connected to each gimbal arm must be well balanced in order to achieve good levelling performance and prevent excessive motor load. That is, the centre of mass of the load on each gimbal arm must be close to the axial centre of rotation for the corresponding motor throughout the desired angular range of operation. The CAD model (fig. 3.8) helped to achieve this goal. Several equally spaced mounting holes were included along the outer gimbal arm in order to provide some fine tuning in this regard. Since the roll and pitch motors have different load profiles and the pitch motor requires a hollow shaft in order to prevent the camera cable from tangling, two different motors were selected. A turnigy ax2804 was chosen for the roll axis and a and T-motor GB2208 was selected for the pitch axis, both models are 14 pole brushless DC motors designed for low weight gimbal constructions and have the same mounting footprint.

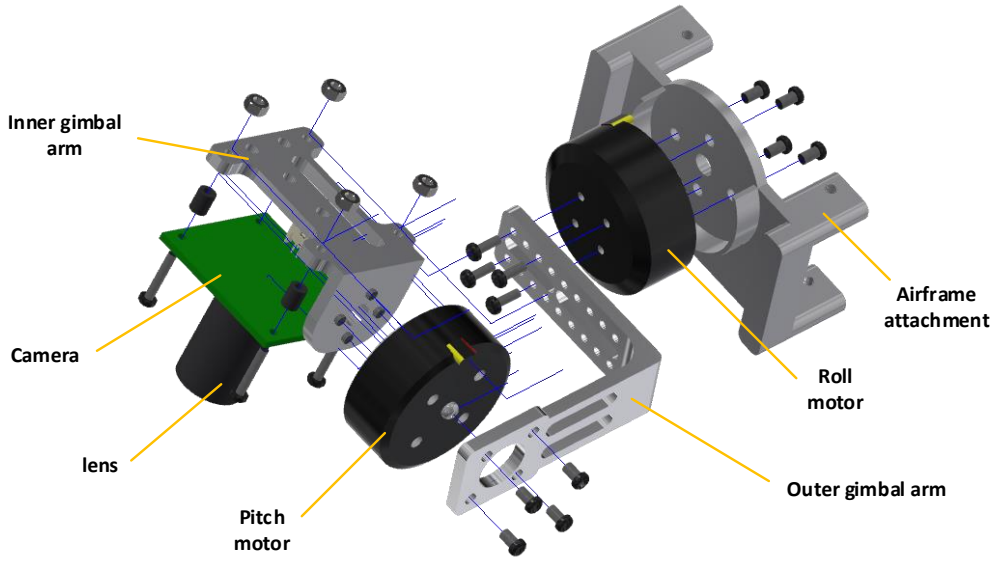


Figure 3.8: Exploded CAD diagram of the motorised gimbal construction, annotated with the key assembly parts.

3.4.4 Full assembly

An image of the full assembly is provided for reference in fig. 3.9. The final aircraft weighs 1.45kg without a battery. All subsequent work in this thesis was performed with a 5000mAh Hacker 3S 20C ECO-X battery weighing 350g.

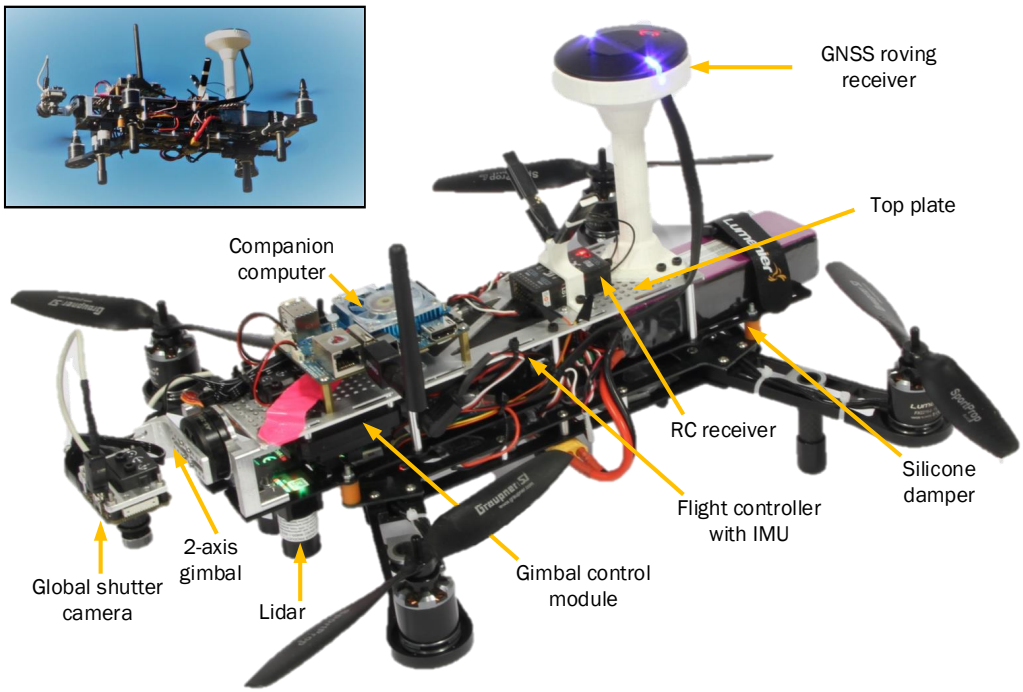


Figure 3.9: **Below:** Annotated image of the physical biorobot build in a landed state. **Inset:** Image of the biorobot in flight.

3.5 Software

In this section an overview of the software development undertaken for this project is detailed. While the hardware architecture contains three control modules (see fig. 3.6), software development was only required on the companion computer with existing communication interfaces and parameter settings used to configure and/or interface with the FCU and gimbal controller. It is beyond the scope of this thesis to outline all of the features implemented in this work, the interested reader is directed towards the code base[9] for this level of overview. Here the overall software framework and the key software features required for subsequent chapters is outlined.

3.5.1 Software stack

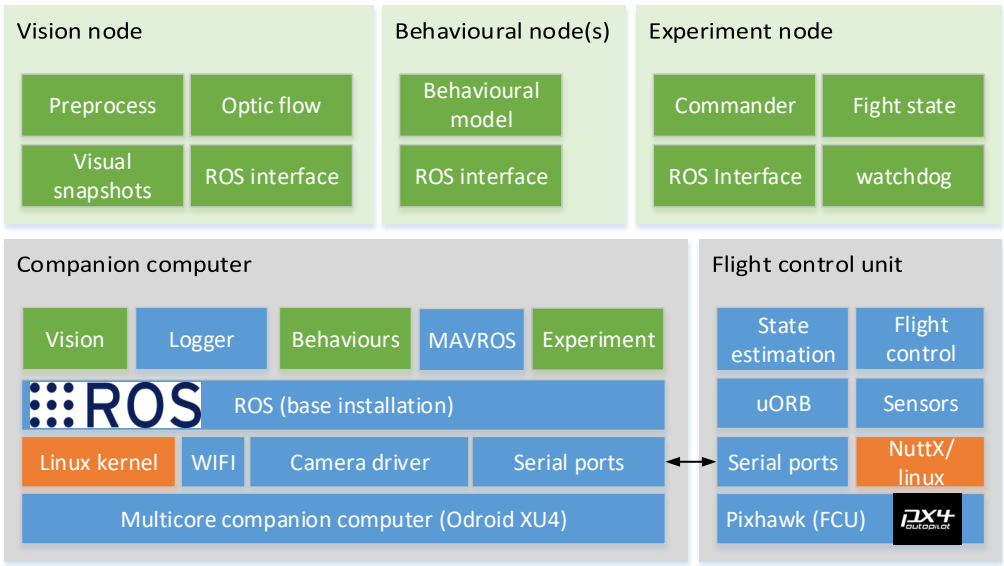


Figure 3.10: **Below:** Software stack of the Companion computer and FCU. Orange boxes indicate the operating system, blue boxes denote existing software modules and green boxes indicate application modules developed for this work. **Above:** Overview of submodules for the software developed in this thesis.

The software stack of the companion computer and the relevant parts of the FCU stack are outlined in fig. 3.10 with their serial link also outlined. In a physical imple-

[9] Code base for this chapter and behavioural code base expanding on this chapter

mentation, the FCU is compiled on Pixhawk hardware which runs on top of the NuttX[10] real-time operating system. By definition, this real-time operating system guarantees that all software modules (flight control, state estimation, sensor updates etc.) update within a guaranteed time interval. However, the real-time scheduler adds some CPU overhead. The companion computer has only been built for operating systems with a Linux kernel. While a real-time Linux kernel is available, this was not pursued due to the associated CPU and development time overheads. Note that if the flight control functionality in the FCU fails to operate the aircraft could catastrophically fail whereas a CPU lockout in the companion computer would result in a stuck position or velocity setpoint. A radio controlled manual override was included in the architecture to mitigate against companion computer failure modes.

The entire software stack can be built on the same Linux operating system in order to interface with a simulation environment. Communication between the virtual companion computer, virtual FCU and the simulation environment is handled with UDP loopback ports. The GAZEBO[11] simulation environment was adopted as a simulation package in this work. To this end, sensor data is generated from the existing GAZEBO sensor model plugins for IMU and GNSS sensors. The ability to test the entire codebase (including the datalink layer) in a simulation environment facilitates a rapid prototyping development cycle. That is, the vast majority of software bugs are identified in the simulation environment leading to quicker and less costly design iterations compared to a scenario where the codebase is tested solely in the physical entity.

In the remainder of section 3.5, the software modules developed for this thesis are presented with the exception of the behavioural nodes which are described in subsequent chapters.

3.5.2 ROS as the companion computer middleware

In order to facilitate the exchange of data between custom software modules and make use of existing robotic software modules ROS[12] was adopted as a middleware layer for the companion computer. Software modules (known as ROS *nodes*) built with the ROS API have access to a common ROS server in which they can share data with other nodes. This is achieved via an asynchronous multicast messaging service, data is broadcast by nodes to the server at a predefined rate with predefined message structure.

[10] https://nuttx.apache.org/
[11] http://gazebosim.org/
[12] Robotic Operating System - https://www.ros.org/

These messages are known as ROS *topics* and they can be subscribed by any other ROS nodes on the server at a predefined rate. A summary of the main ROS nodes and topics used in this project is presented in fig. 3.11.

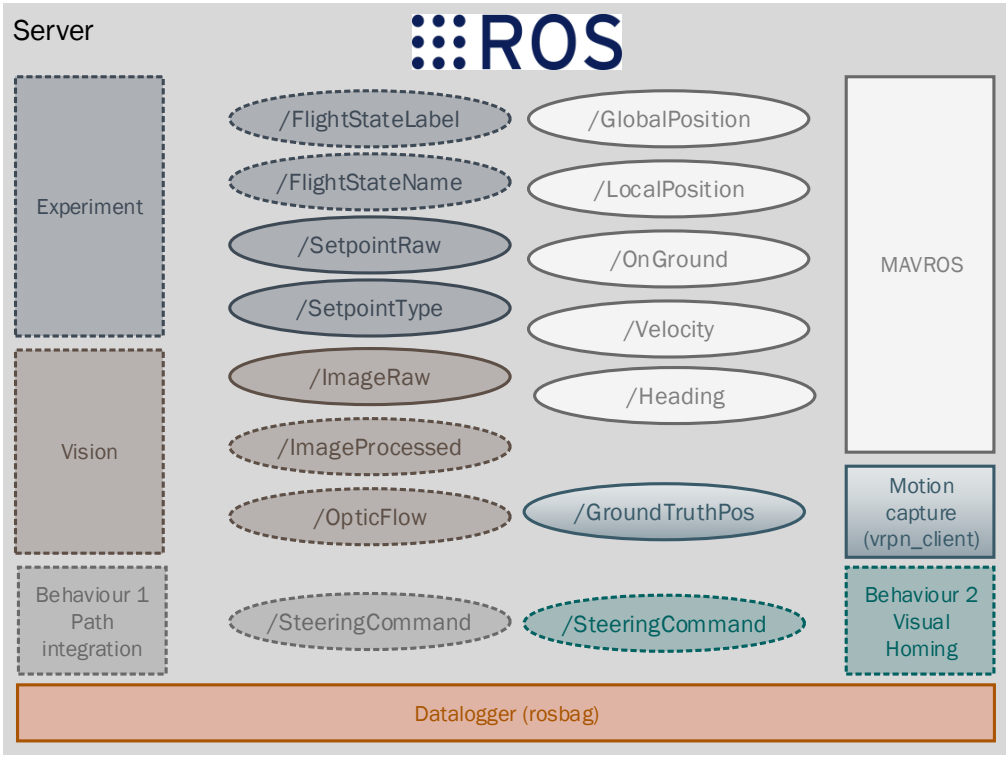
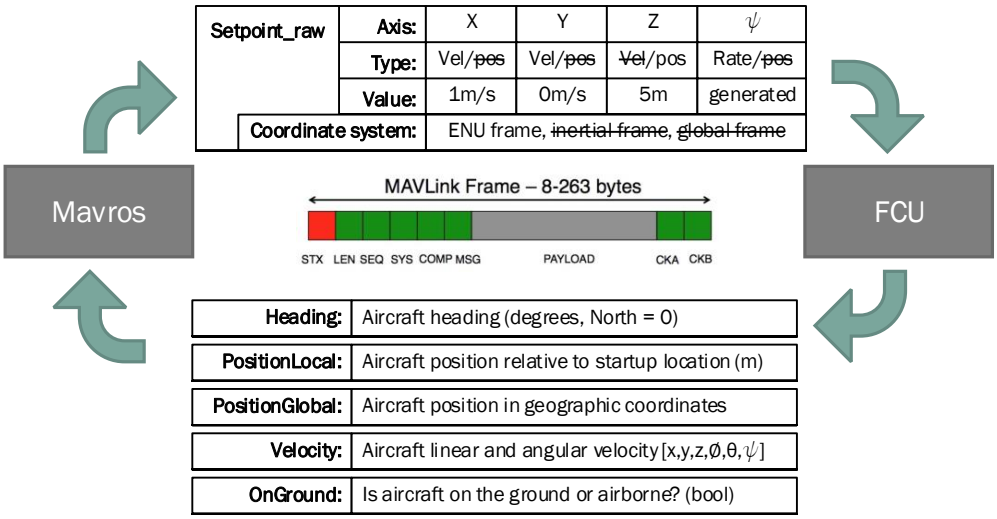


Figure 3.11: ROS server message topics. ROS nodes shown as rectangles and ROS messages (topics) shown as ellipses. Solid borders indicate existing re-purposed nodes or topics and dashed borders indicate new nodes or topics developed for this thesis.

In terms of existing ROS software packages, MAVROS[13] was adopted in order to handle the MAVLink[14] communication channel with the FCU. This enabled the other ROS nodes on the companion computer to send setpoint messages to the FCU, and receive state estimation data back from it. This data exchange is summarised in fig. 3.12.

Another existing ROS software package used for this work is the *rosbag*[15] data logging node (see fig. 3.11) which can log ROS topics from any other ROS node. All ROS topics are timestamped and MAVROS provides a real-time clock synchronisation

[13] http://wiki.ros.org/mavros
[14] MAVLink (https://mavlink.io/) is a lightweight messaging protocol for transferring data with drones.
[15] http://wiki.ros.org/rosbag



Setpoint_raw	Axis:	X	Y	Z	ψ
	Type:	Vel/~~pos~~	Vel/~~pos~~	~~Vel~~/pos	Rate/~~pos~~
	Value:	1m/s	0m/s	5m	generated
Coordinate system:		ENU frame, ~~inertial frame~~, ~~global frame~~			

Heading:	Aircraft heading (degrees, North = 0)
PositionLocal:	Aircraft position relative to startup location (m)
PositionGlobal:	Aircraft position in geographic coordinates
Velocity:	Aircraft linear and angular velocity [x,y,z,Ø,θ,ψ]
OnGround:	Is aircraft on the ground or airborne? (bool)

Figure 3.12: MAVROS $\Leftrightarrow$ FCU data transfer summary. Top table illustrates the setpoint_raw message that allows flexible (velocity or position) control setpoints to be set independently for each control axis according to a desired coordinate system. The *value* row and text without strikethrough markings shows a typical setup for the path integration behavioural node. The bottom table indicates the data frames that are periodically transferred from the FCU to the ROS server where it is either logged or used as an input into the behavioural models or experiment node. The graphic in the middle demonstrates the structure of a MAVLink message used to send data.

function between the companion computer and the FCU. This means that sensor data sent from the FCU can be timestamped with the measurement time rather than the time it is received at the subscriber node. This enables an analysis of the state of software modules, given their inputs. Raw and processed images can for example be stored along with the aircraft pose and the geographic location of their capture. In the same dataset, the steering response of the behavioural models in response to this visual input is also recorded providing a convenient dataset for offline analysis.

3.5.3 Experiment node

The experiment node is the executive for all mission logic relating to an experiment. This includes behavioural model embodiment and other mission tasks including arriving at the experiment starting location and safely landing at a required location. For

the navigation experiments performed in this thesis, a minimum of five phases were required; arming, takeoff, outbound route, homing (with the behaviour model) and landing. In most experiments there was a return to home phase before landing and additional waypoints were common between the takeoff and outbound route phases. The experiment node therefore cycles through flight states (section 3.5.3.1) using a commander submodule (section 3.5.3.2). The experiment node is responsible for sending the setpoint_raw topic to MAVROS.

3.5.3.1 Flight states

In order to flexibly develop experiments, a flight state class was derived for each of the typical phases. Since flight classes share many of the same properties and attributes, an object orientated class inheritance approach was adopted, using the inheritance structure outlined in fig. 3.13.

Each flight state as a minimum has a desired setpoint which specifies for each control axis (x,y,z,ψ), the control type (velocity or position) and the control setpoint. The means of generating this setpoint could be predefined (e.g. a position waypoint), generated as a time series (e.g. following a sine wave trajectory) or as a reactive feedback loop (e.g. visual homing). For the latter type, it is essential for the node to be able to access the biorobot's state and potentially other information from the ROS middleware layer. For example, the path integration homing flight state must be able to access the steering signal data from the behavioural model in order to parse this as a setpoint. It is also often convenient for behavioural models to access flight state data. For example, the path integration behaviour node is required to ignore optic flow input for flight states that are not related to the inbound or outbound journeys. This was achieved here using the flight state *flight_label* property (i.e. outbound states must always labelled as 'outbound' for this mechanism to work).

3.5.3.2 Commander

The commander facilitates transitions between flight state classes. Here, this involves iterating through a list of predefined parameterised flight classes but if required, flight states could alternatively be instantiated procedurally *on the fly*. When a particular flight state reports that it has completed, or it times out, the next state is invoked. Once the new flight state is loaded, the final setpoint from the previous flight state is preserved until the prerun routine is complete. The prerun routine serves to check whether data

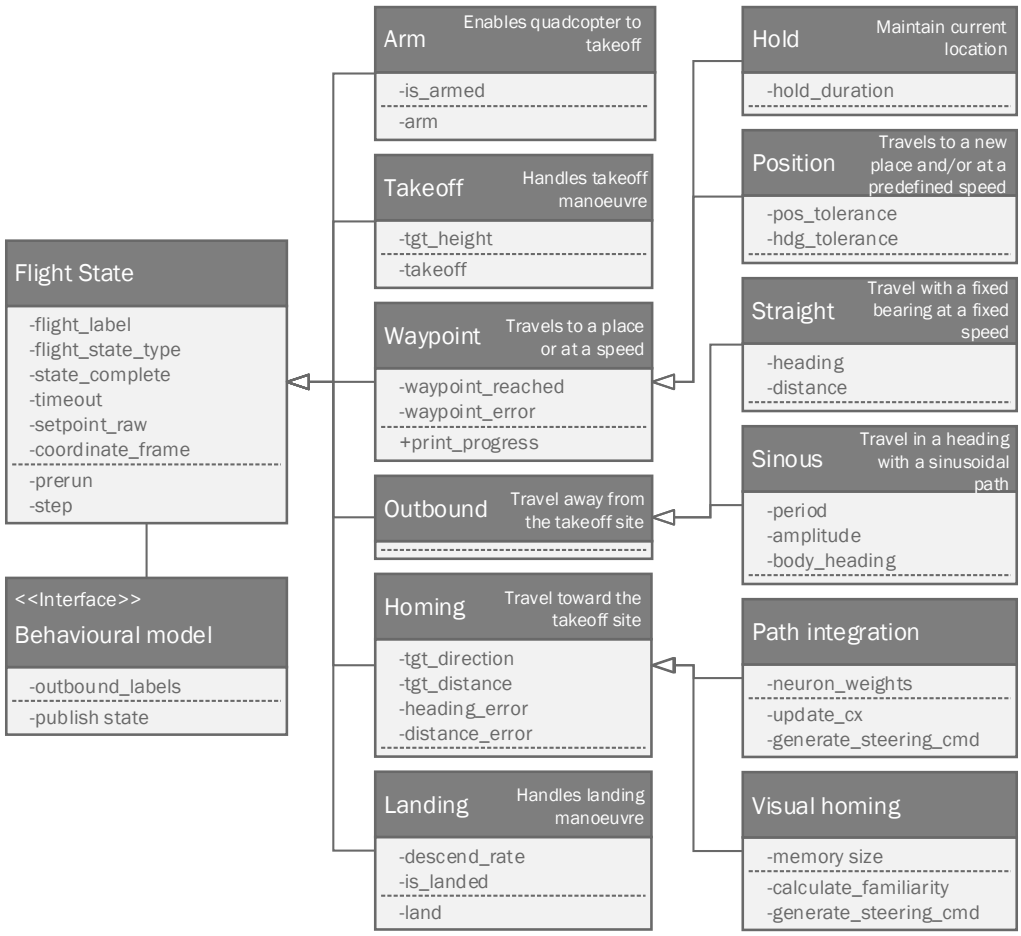


Figure 3.13: Flight states UML class diagram. This outlines the main flight states used in this thesis. Specialist flight states inherit the properties of the generic flight class for minimal code reuse. New flight states can be created by inheriting an existing class and adding any new desired properties or methods.

required to support a flight state is available and valid. In order to comply with the FCU offboard control mode requirements, the setpoint is published at 50Hz in a separate thread. Flight states are created at initialisation time and then deleted once they are complete. Behavioural models by contrast, persist throughout the entire experiment.

3.5.4 Vision node

The vision node fulfils three purposes. Firstly, it acts as a camera trigger and secondly it provides basic vision processing pipeline functionality - e.g. greyscale, resizing images and lens rectification. For these basic image processing steps the ROS image

processing nodelets were used[16]. For large dimension data structures like images, the process of copying of memory between the ROS server and each subscribing node becomes a significant overhead. Nodelets by contrast, provide a mechanism for zero copy transport between algorithms. Finally, the image node performs behavioural-specific operations, such as converting images into optic flow (outlined in Chapter 4) or performs general visual processing tasks such as the wavelet transform on the incoming images (outlined in Chapter 5).

3.6 Simulation environment

The Gazebo simulation environment leverages an amalgam of a graphical rendering pipeline and a suite of physics simulation codebases in order to provide a unified physical modelling environment. This environment was used extensively to support software development throughout this thesis. In particular, it enabled fast design iterations and an ability to repeat tests many more times than would be possible with the physical entity. For the most part it was possible to use existing Gazebo resources developed by the PX4 software-in-the-loop package[17]. Here development effort was restricted to configuring a sufficiently analogous MAV model, and incorporating the test worlds outlined in subsequent chapters. A schematic of the simulation environment used for this work is included in fig. 3.14, this serves as a reference for the remainder of this section.

3.6.1 Physics simulation

Robot models are specified in Gazebo using a hierarchical tree of *links*, each link has an associated mass and inertia tensor. Links can be interfaced using *joints*, here joints are either specified as revolute (freely rotating about one axis) or fixed (static interface). The robot structure is defined using a simulation description format (SDF) file. It is also possible to attach sensor models onto any of the physical links, several parameterised plugins have already been implemented for this purpose including camera, lidar, IMU and GPS sensors. Each sensor type has its own associated adjustable noise model.

In terms of aerodynamics and fluid mechanics, Gazebo has a lift-drag plugin that can be used to calculate the thrust and tangential force (drag) generated by a motorised

[16] http://wiki.ros.org/image_proc
[17] https://github.com/PX4/sitl_gazebo

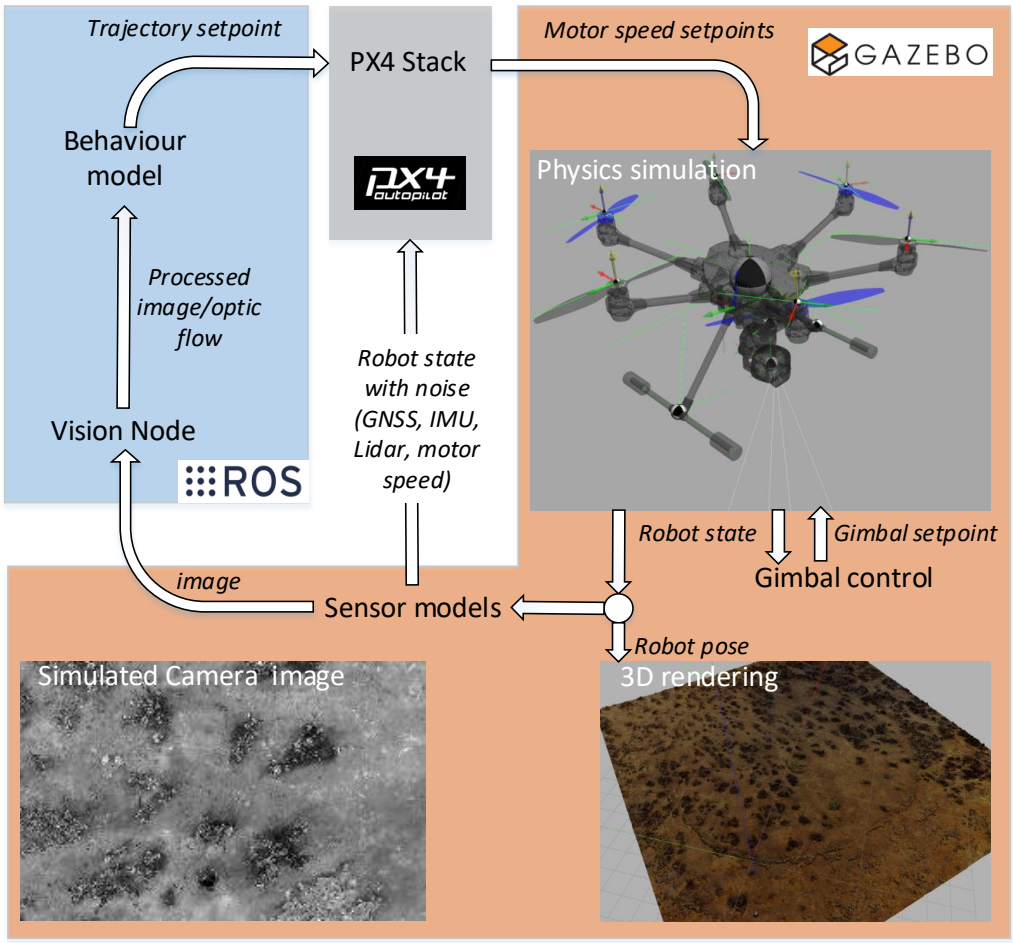


Figure 3.14: Architecture of the simulation *software-in-the-loop* setup. The PX4 stack and ROS server are discussed in section 3.5.3.1. The physics simulation processes are outlined in section 3.6.1 and the 3D graphical rending functionality is presented in section 3.6.2.

propeller when it is rotated at a given speed, the mathematical formulation of the electric motor physics and lift-drag equations are reviewed in (Furrer et al., 2016). The physics simulator can therefore subscribe to the commanded motor speeds from the PX4 flight stack which are used as an input for the corresponding simulated MAV motors. Fluid dynamics is notoriously difficult and computationally expensive to simulate accurately and the first order lift-drag plugin formulation and lack of airflow modelling mean that the Gazebo aerodynamic simulations are a simplification of the physical system, with several force and moment terms being neglected. As a consequence, these simulations are most useful as an interface test rather than an accurate prediction of how a particular

aircraft flies. However, given that the physical aircraft is an abstracted model of flying insect dynamics and control (i.e. systems dynamics are not equivalent) and this is not deemed critical for the target behaviour models, this physics simulation is nevertheless considered a useful auxiliary biorobotic evaluation platform.

In light of the physical modelling limitations, the existing *typhoon_480* hexacopter model in the PX4 repository formed the basis of the robotic simulation model because it has a built-in gimbal system (shown in fig. 3.14). Some modifications are required to the original model in order to support the work herein. The camera resolution, sampling rate and default viewing angle were modified, a ground facing range sensor was added and the remaining unused camera interfaces are removed. All simulation modelling updates are included in the software repository. Note that the gimbal controller is a completely simulated entity (i.e. this is a different code base to what is used in the basecam module (section 3.4.1.5)), as with the biorobot this is decoupled from the FCU.

3.6.2 3D rendering

The Gazebo 3D rendering pipeline provides a user-controlled third person view of the world. In addition, images can be rendered from a simulated camera with a known pose in the world. The Gazebo camera sensor plugin is used for this purpose, the resultant image feed can be broadcast to the ROS server at a predefined rate. In order to provide visual realism in the simulator, it is possible provide a 3D graphical model for each link. It is usually more efficient to also provide a simplified structure for each link for the collision calculator, especially when specifying large meshes for the world file. Gazebo also has a scene or environmental modelling API which facilitates adjustment to the lighting schemes, shadow simulation (on/off) and model transparency (on/off).

In this work a subset of the *Habitat3D*[18] models was converted into the required collada file format so that it could be used in the Gazebo simulation. This and all other attributes used in this thesis are publicly available in the main software repository.

3.7 Evaluation

In this section the aircraft functional requirements are briefly verified and some metrics about its overall performance are quantified as a reference for future work. Note that

[18] https://insectvision.dlr.de/3d-reconstruction-tools/habitat3d

the purpose of this section is quantify the control characteristics of the aircraft rather than to produce an optimal or fully justified biological equivalent controller.

3.7.1 Control axes

In order to verify that the main control axes have been implemented, the aircraft was commanded to hold a height of 5m then, with a fixed heading angle command of 0°, the aircraft was given the following ordered velocity vector sequence as a control input; [1, 0], [-1, 0], [0, 1], [0, -1], [1, 1], [-1, -1]. Where each vector duplet is the aircraft velocity setpoint $[x, y]$ in ms^{-1}, each vector setpoint was maintained for 10s. The aircraft heading angle setpoint was then twice commanded to rotate by 180°. The weather was fair but there were wind gusts of up to $10ms^{-1}$, the terrain was a flat arable field (Lat:55.7633185 Lon:-2.2386655).

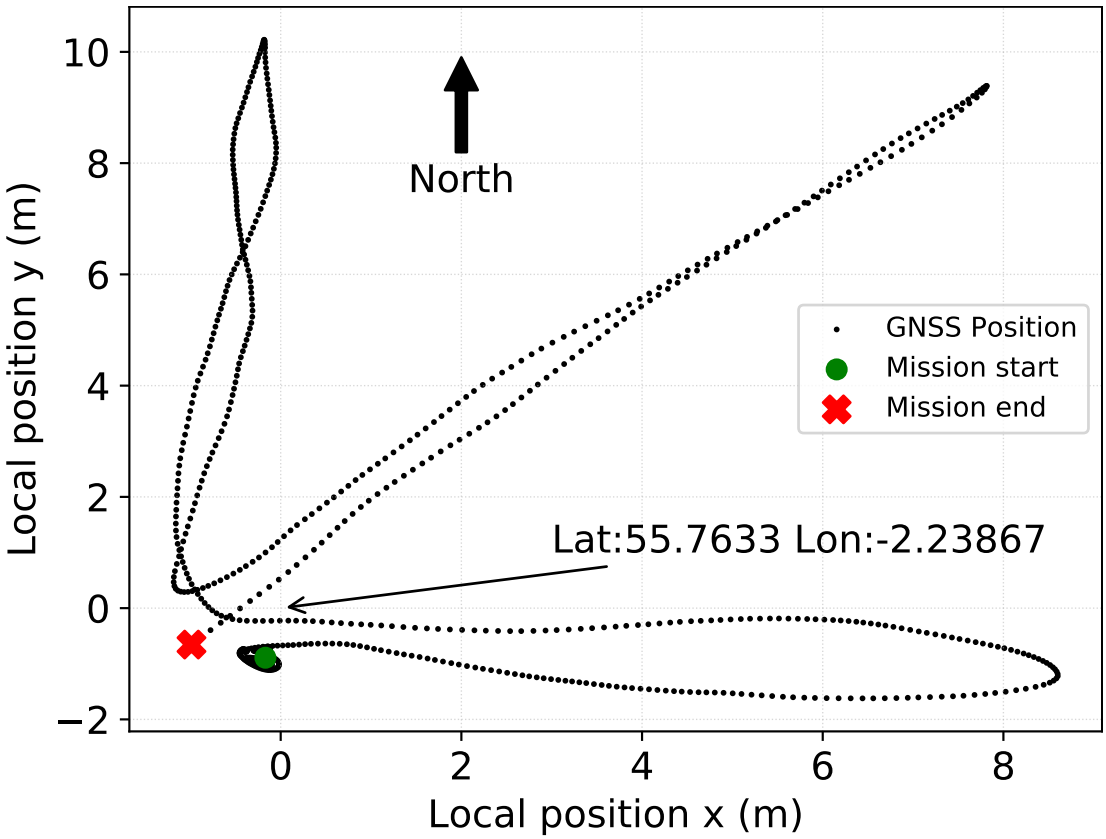


Figure 3.15: RTK GNSS estimates of aircraft's local horizontal position while performing the control axes evaluation procedure. The latitude and longitude coordinates of the origin are included in the figure.

The control signal and control response for each axes of the behavioural model interface are plotted as follows; velocity control in x and y dimensions of the body

frame (fig. 3.16), aircraft yaw rate (ψ - fig. 3.17) and the aircraft height above ground (fig. 3.18). All figures correspond to the same flight which is was presented in fig. 3.15. The control response is relatively sluggish for the velocity control axes, with a median rise time of 1.5s and persistent steady state error, especially with motion in the x direction (governed by pitch control). By comparison the yaw angle and height position control systems have a faster rise time.

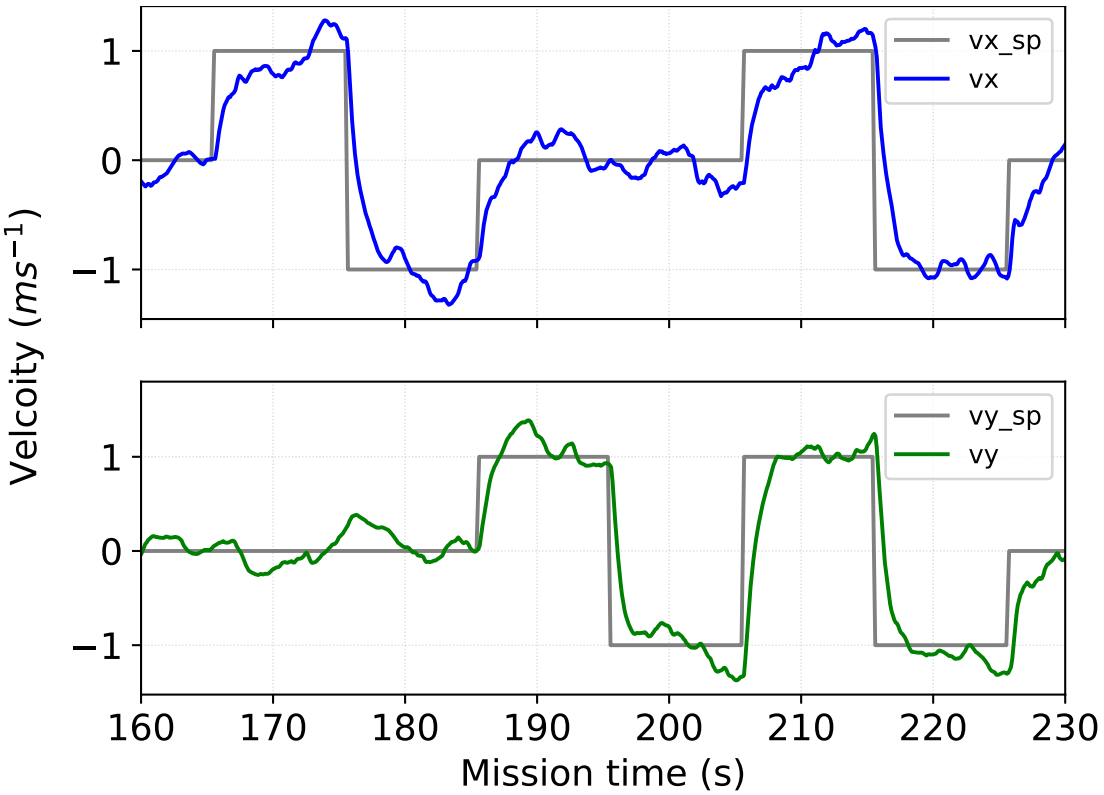


Figure 3.16: Aircraft velocity estimates vx (**top**) and vy (**bottom**) logged while performing the control axes evaluation procedure. The setpoints for vx and vy are also presented in the corresponding subplots (shown as vx_sp and vy_sp in the legend).

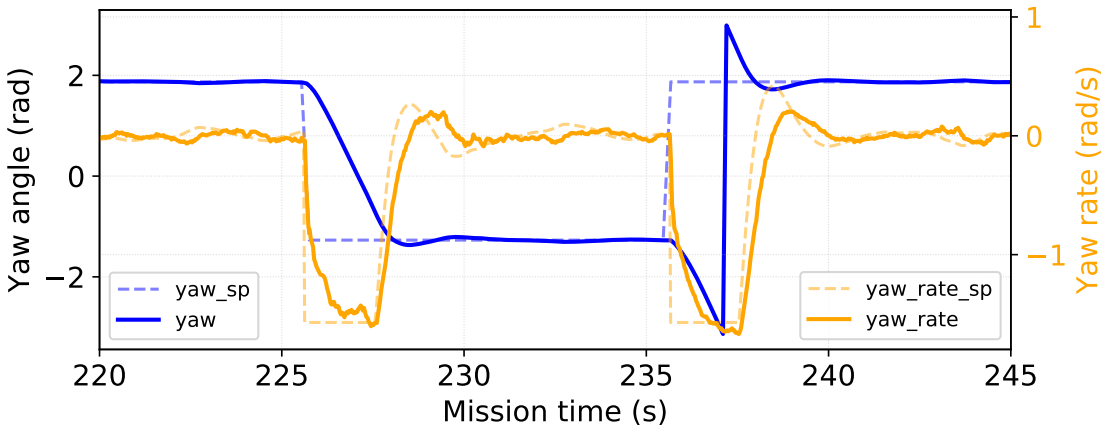


Figure 3.17: Yaw angle and body yaw rate estimates while performing the control axes evaluation procedure.

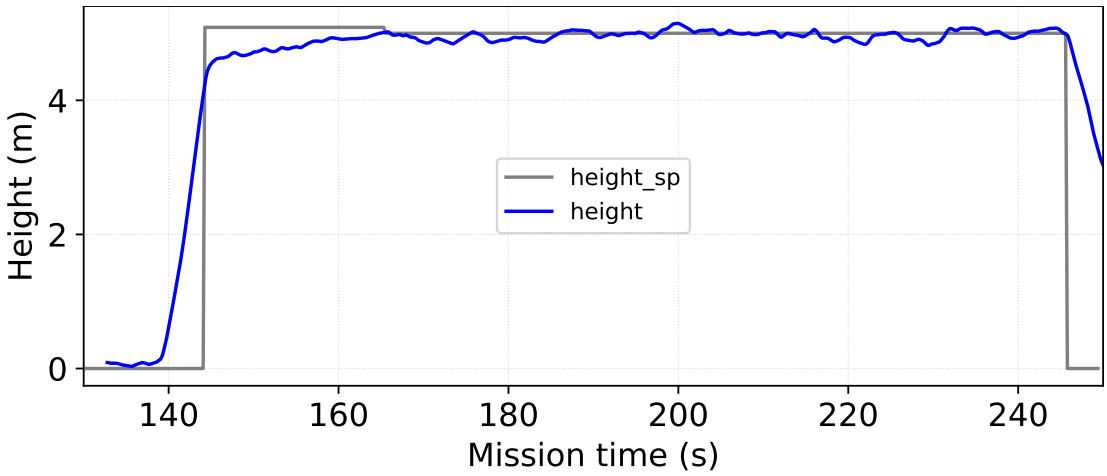


Figure 3.18: Aircraft height estimates while performing the control axes evaluation procedure.

3.7.2 Procedural trajectory following

In order to evaluate procedurally generated flight states a random walk was generated whereby the desired heading of the biorobot was randomly selected from a von Mises distribution at a predefined (parameterised) interval (see fig. 3.19). The aircraft heading is aligned with the direction of travel. A periodic wave form trajectory generator was also produced, the sinusoidal trajectory in fig. 3.19 was generated using this. Here the aircraft's heading is aligned with the x axis.

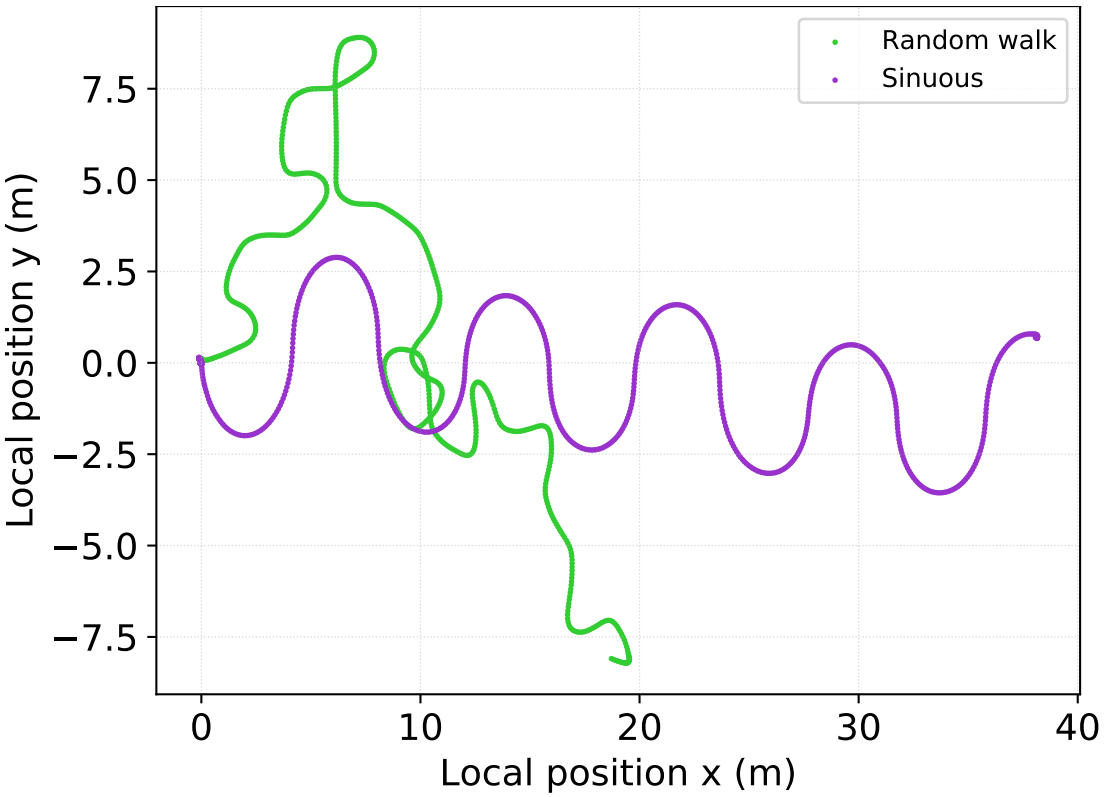


Figure 3.19: Top view of trajectories generated from the 'random walk' and Sinuous' flight states.

3.8 Discussion

Embodying flying insect path integration and visual navigation behavioural models requires a platform that can move around in 3D space and change its heading according to the behavioural model's steering command. Here a quadcopter has been identified as a suitable platform for this purpose. Existing hardware units and software modules have already been developed for solving the problem of quadcopter state estimation and control. In this chapter an architecture of existing control modules, related software modules and peripheral avionics was outlined and their interfaces were defined. Based on this architecture, a software platform for deploying behavioural models in the context of an outdoor experimental environment has been developed.

In serving the work in the upcoming chapters (Chapter 4 and Chapter 5) the biorobot has undertaken over 150 experimental missions and has proved to be a reliable platform. Nevertheless, there are some limitations with the current setup that could be improved in future design iterations as listed below:

1. The camera USB cable currently prevents the gimbal remaining level at large

attitude angles. This is because the cable has an outer shielding mesh (to mitigate against the magnetic fields generated by the gimbal motors). If higher vehicle speeds and therefore higher body attitudes are required in future, a slip ring could be added to the motor. Alternatively, a wireless connection could be implemented but this would require a battery to be added to the gimbal assembly.

2. Large yaw angle commands result in two of the motors reaching their maximum power level and a corresponding unintended loss of height. In order to mitigate against this the FCU maximum yaw rate parameter is set to $120°/s$. A higher voltage, four cell Lipo battery would help to increase the thrust generated by the motors and, given the same mAh capacity, lead to a longer flight times.

3. Fitting a lightweight RGBD sensor or stereo camera system to the aircraft could help to improve the offline analysis. This would be useful for generating virtual versions of the habitats, measuring a ground truth for the distance metric used in Chapter 4 and adding obstacle avoidance functionality thereby extending the environments that the biorobot can operate in.

Chapter 4

Deploying the central complex path integration circuit on an MAV

4.1 Introduction

In addition to the strong behavioural evidence that path integration is an important component of the flying insect navigational toolbox (Heinze et al., 2018), the first anatomically constrained candidate neural path integration circuit was recently presented in (Stone et al., 2017). This publication identifies neurons that deliver speed and heading information into the central complex and proposes how the circuit formed by the connected downstream neurons could accumulate this information and ultimately produce a steering signal that guides the animal back to a place of interest. This model has been previously tested in a simulator and on a land based robot over small distances indoors. In this chapter, the central complex path integration model is deployed on the aerial biorobot that is detailed in Chapter 3. The initial motivation for this undertaking was to validate the central complex model on a non-holonomic, aerial platform and gain an understanding of its operating range. In particular, it was desirable to understand the utility of the circuit over relatively long range distances in outdoor natural scenes.

A biologically plausible ground speed estimation system was regarded as an important element in achieving a useful estimate of the operating range of a path integration circuit. The speed estimation process is treated as a something of a black box in (Stone et al., 2017)[1]. The matched filter approach (Franz et al., 2000) is a previously defined

[1]Tracing invertebrate neural pathways through the optic lobes is currently an intractable process because of the dimensionality of the problem (requiring cells to be recorded simultaneously) and the density of small neurons in the medulla (making the task of repeatably finding even a single target neuron fiendishly difficult). Neuroscientists are often able to bypass this issue by finding a downstream large

functional approximation of how EMD signals could be combined into a single measure of optic flow experienced about a particular axis. However, a major challenge when using this value for odometry is that induced flow is modulated by the distance to the flow inducing surface. This issue is addressed in (Franz et al., 2000) by the use of statistical information about the animal's behaviour and environment in order to estimate a static depth estimate for each view direction. However, this model was designed with flight control rather than odometry in mind. The implications of incorrect speed estimates generated by this circuit when it is an integral part of an odometry system have yet to be investigated.

Harmonic radar tracks currently provide the best means of quantitatively describing the long range navigational capabilities of the species of interest. Most harmonic radar datasets have been collected in wide open flat areas where lateral optic flow is only present on the horizon at an effectively infinite distance. A hypothesis that flying insects must therefore be able to perceive optic flow from a ventral, or fronto-ventral aspect is evidenced by flight control mechanisms (Linander et al., 2017) and tracking of ground based features (Menzel et al., 2019). However, existing datasets that describe the ommatidial view directions of the species of interest suggest that there is little or no visual acuity directed towards the ground (Taylor et al., 2019).

With the above information in mind, finding the optimal view configuration for reliable odometry in flying insects was deemed a worthy endeavour. Understanding the limitations of the central complex circuit for a given habitat was also seen as a useful task because this information informs ethologists of the expected error of the path integration circuit after a complete journey. Thereby also indicating the minimum required operating range of drift free cues such as visual homing. Furthermore, by understanding what parameters are important in the matched filter approach, the neuroscientist has a better idea of where in the anatomy to search for this circuit. This approach may also provide roboticists with some inspiration on view configurations when designing low resolution odometry circuits. The task of ground speed estimation with low resolution sensors is of particular interest in engineering applications because a speed signal can be easily integrated into existing state estimation algorithms. In this chapter these questions have been addressed. The following sections have been extracted from a recently published paper Stankiewicz and Webb (2020).

---

neuron that is correlated with a visual stimulus. For example, in (Stone et al., 2017), the firing rate of the TN neurons is shown to be correlated with an optic flow pattern commensurate with forward motion presented to the animal under test using an LED arena.

4.2 abstract

We have deployed an anatomically constrained neural model for path integration on a real world, holonomic aerial platform. Based on the insect central complex, the model combines estimated heading and ground speed information to maintain a location estimate that can be used to steer the agent directly home after convoluted outward journeys. We implement a biologically plausible method to estimate ground speed using optical flow. We discover that a downward viewing, mechanically stabilised and height compensated vision system performs well in a range of natural environments, even when visual acuity is reduced to 3°/pixel. In a flat outdoor environment, the worst case final displacement error increases at a rate of 1.5m per 100m outbound travel. The field of view of the vision system has no impact on odometry performance.

4.3 Introduction

Many species of flying insect routinely conduct long foraging journeys from which they can return directly to their starting location. We have recently developed an anatomically constrained model of how this behaviour can be accomplished by a path integration (PI) circuit located in the central complex (CX) of the archetypal insect brain Stone et al. (2017). Conceptually this circuit forms an allothetic navigation scheme that is globally anchored via a sky compass. It has been tested in simulation and over small distances on an indoor wheeled robot, but as yet has not been evaluated under the same natural conditions as a flying insect. Here, we present the results of deploying the CX model on an autonomous micro-aerial vehicle (MAV).

A key challenge was how to obtain biologically plausible ground speed estimates. Flying insects are thought to sense their egomotion primarily through optical flow percepts which are computed with matched filters: *"ensembles of neurons which extract crucial components of stimuli while ignoring irrelevant information."* Kohn et al. (2018). While neurons with receptive fields that resemble matched filters for rotational motion are well documented, circuits for inferring translational motion are less well understood. A problem with estimating translatory egomotion from optic flow is that flow magnitude is inversely proportional to the distance to the flow inducing surface (depth). In Franz and Krapp (2000) the average horizontal flight depth value per photoreceptor is used to weight flow according to viewing direction. Here, we suggest that one reliable depth estimate can be used to adaptively tune all photoreceptor depth weights.

We show that this approach can be successfully integrated to the CX circuit to produce effective PI behaviour for an MAV operating in real outdoor conditions.

4.4 Methods

4.4.1 Robotic platform

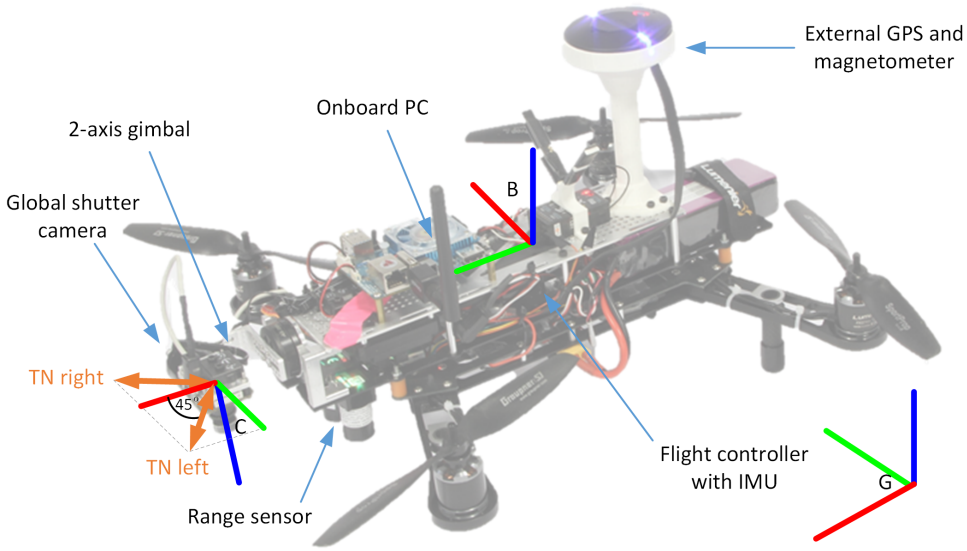


Figure 4.1: Photograph of the MAV platform with annotation of key components and relevant (camera, C, global, G, and body, B) coordinate systems.

The MAV (depicted in fig. 4.1) was developed with an off-the-shelf airframe (Lumenier QAV400), avionics (flight controller, Pixhawk 2.1), differential GPS (Here+V2 RTK GNSS), range sensor (garmin lidar-litve v3), gimbal controller (Basecam simpleBGC 32-bit), RC receiver (FrySKY RX8R), singleboard PC (Odroid XU4)) and global shutter camera (Matrix vision Bluefox2, 200w). A custom top plate and the gimbal arms were fabricated from aluminium, the onboard GPS mast and RC receiver mounting brackets were 3D printed.

We made use of the PX4 flight stack on the pixhawk flight controller which was set to autonomous mode and communicated with via a UDP serial link handled using MAVROS. The CX module was wrapped in a ROS node and acted as a state in our generic behavioural state machine. A separate ROS node was developed to handle ground speed estimation. A major benefit of using the opensource PX4 ecosystem is that a customisable simulation model was available for use in the GAZEBO environ-

ment. This provided a realistic physical and graphical simulation setup, supporting rapid development and evaluation in otherwise untestable configurations. The PX4 flight stack features an extended Kalman filter software module (EKF2) which is used to provide *ground truth* location estimates. The differential GPS system that supports EKF2 in our setup has a reported horizontal accuracy of ±6cm, the heading error of the ground truth is unspecified.

The primary function of the camera in this system is to facilitate ground speed estimation. We therefore opted for a downward facing configuration on both the real and simulated MAVs. Insects are known to stabilise their head during flight Viollet and Zeil (2013) and because MAVs have a high degree of body vibration and the body necessarily pitches and rolls to generate translational thrust, we mimicked head stabilisation with a 2-axis gimbal and found this essential for good ground speed estimates (see fig. 4.6).

4.4.2 Ground speed estimates from optical flow and matched filters

Flying insects have been shown to estimate distance travelled from optic flow Srinivasan et al. (1991). Recent neural recordings in 2 species of bee have revealed two pairs of motion sensitive neurons that respond most strongly to flow-fields corresponding to animal head motion along orthogonal axes ***left*** (-45°) and ***right*** (45°) in azimuth with respect to the animal's anterior Stone et al. (2017) (labelled TN left/right in fig. 4.1). We mimic these neural properties, but rather than replicate the near-panoramic vision system of an insect, we used a conventional camera with a 42° field of view, pointing downward. One advantage of this configuration is that optical flow induced by pure yaw motion sums to zero when combined with translational matched filters with preferred directions that are parallel to the ground. However, as it is not apparent that the insect visual field actually includes the area directly below the animal Taylor et al. (2019), we also investigate the effect of raising the camera view angle in section 4.5.5.

The processing pipeline for each speed cell is presented in fig. 4.2a. The camera operates at 10Hz. The preprocessing stage converts to grayscale and resizes to 150x235 pixels. A 2-frame Farnebäck dense optic flow algorithm Farnebäck (2003) is used to compute the optical flow field. This is combined with matched filters for self-motion and weighted by a depth estimate for each pixel to obtain a ground speed estimate as follows:

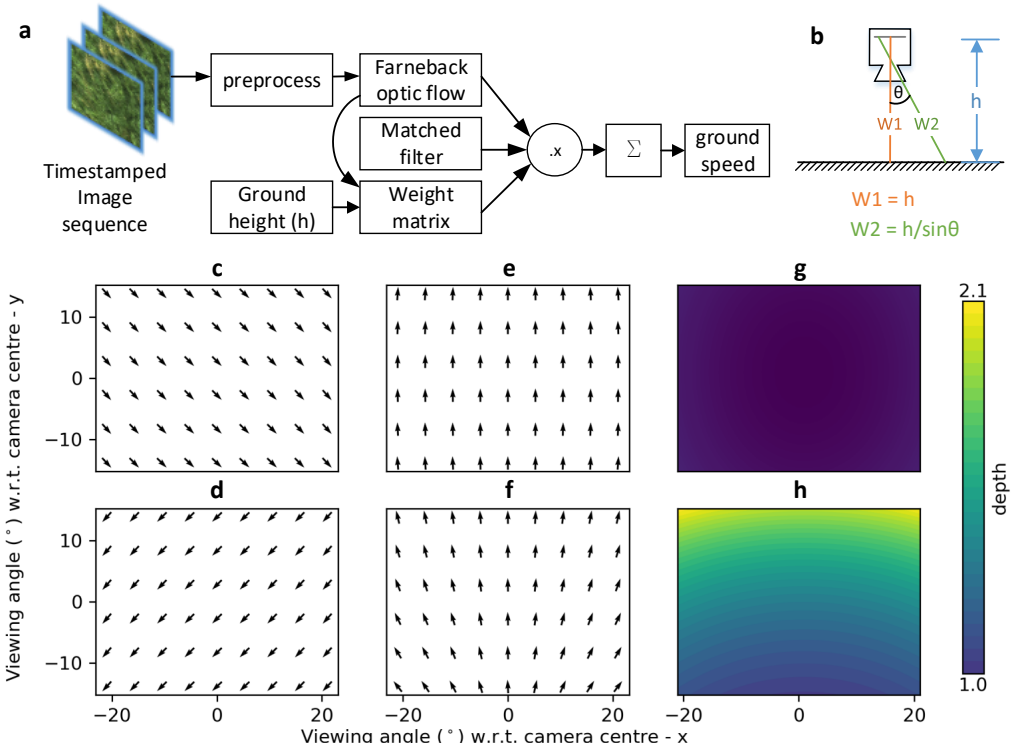


Figure 4.2: **a** Pipeline for ground speed estimation. **b** Illustration of example weight factors, W1 and W2, based on pixel viewing image sensor height (h) and pixel viewing direction. **c**&**d** Matched filters for the left and right speed cells respectively when the camera is aligned with gravity and the camera x-axis is towards the front of the aircraft. **e**&**f** Matched filters for the dual camera setup (see section 4.5.5) with the camera rotated $\pm 45°$ about the body frame Z-axis, for **f** the camera is also pitched upwards +45°. **g** Depth weights for a downward facing camera. **h** Depth weights for a camera pitched up by 45°.

We adapt the method in Franz and Krapp (2000) to generate matched filters, i.e., neurons with a pattern of preferred motion across the visual field that corresponds to the predicted flow for self motion around or along a particular axis. In keeping with the speed cell neurons from the central complex model we create a left and right pair of orthogonal matched filters as depicted in fig. 4.3. The equations for these matched filters are included in eq. (4.1) and eq. (4.2) for the left and right TN neurons respectively. Quiver plots of example filters generated by this procedure are shown in fig. 4.2c-f.

$$\mathbf{u}_{m,n}^{left} = \mathbf{d}_{m,n} \times \mathbf{a}^{left} \times \mathbf{d}_{m,n}, \quad for \quad m \in \{1,w\} \quad n \in \{1,h\} \tag{4.1}$$

$$\mathbf{u}_{m,n}^{right} = \mathbf{d}_{m,n} \times \mathbf{a}^{right} \times \mathbf{d}_{m,n}, \quad for \quad m \in \{1,w\} \quad n \in \{1,h\} \tag{4.2}$$

where:

$\mathbf{u}_{m,n}^{left}$ = matched filter for the left speed cell
$\mathbf{u}_{m,n}^{right}$ = matched filter for TN right speed cell
$\mathbf{d}_{m,n}$ = view direction of pixel m,n
$\mathbf{a}^{left}$ = preferred direction of left matched filter with vector $(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0)$ in the camera frame
$\mathbf{a}^{right}$ = preferred direction of right matched filter with vector $((\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0))$ in the camera frame
m = horizontal pixel coordinate
n = vertical pixel coordinate
w = image width
h = image height

4.4.2.1 Matched filters

4.4.2.2 Weight matrix for depth

The world is modelled as a flat plane and the projected distance between a given pixel and the ground plane is calculated using the camera's commanded orientation and the aircraft's instantaneous height (see fig. 4.2b):

$$\mu_{m,n} = \frac{height}{sin(\angle \text{gravity}, d_{m,n})} \quad for \quad m \in \{1,w\} \quad n \in \{1,h\} \tag{4.3}$$

where:

$\mu_{m,n}$ = Depth weight matrix
$\angle$gravity = Angle with respect to gravity
$height$ = instantaneous ground height estimate

Example depth maps according to the weight matrix are included in fig. 4.2g&h. We labelled any pixels with a flow magnitude of less than 0.2 were labelled *noisy*.

4.4.2.3 Ground speed estimate

Egomotion (displacement) in the speed cell orthogonal basis can be estimated by summing the elementwise product of the flow field, matched filter and the depth weight

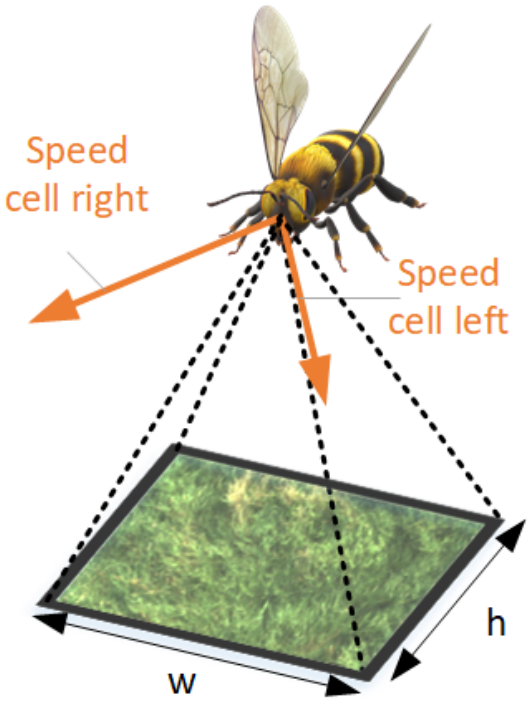


Figure 4.3: Speed cell geometry. The preferred direction of the left and right speed cells and camera viewing frustum is shown relative to the agent. The current image width and height are shown as w and h respectively.

matrix. This is functionally equivalent to the operations performed by wide-field tangential cells found in the lobular plate Franz and Krapp (2000). For best results this value is normalised by the number of non-*noisy* pixels. An absolute ground speed estimate can be found by multiplying the projected displacement with the image frame time interval:

$$speed\_cell^i = \frac{\sum_{m=1}^{w}\sum_{n=1}^{h}\left(F \cdot \mathbf{u}^i_{m,n} \cdot \mu^i_{m,n}\right)}{(wxh) - \sum(\text{noisy pixels})} \cdot \delta t \quad for \quad i \in \{left, right\} \tag{4.4}$$

where:

$speed\_cell$ = speed cell (TN neuron from Stone et al. (2017))
F = Optic flow field (generated with the Farnebäck algorithm)
$\angle$gravity = Angle with respect to gravity

4.4.3 The Central Complex (CX) circuit

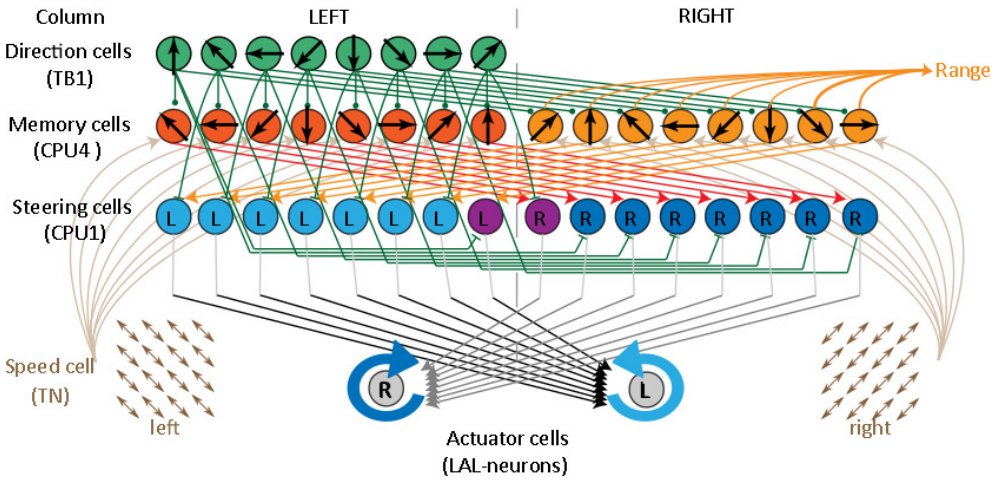


Figure 4.4: CX model topology and key interfaces, adapted from Stone et al. (2017). Labels on the left hand column specify the neuron type for each row with the exception of the bottom row which contains speed and actuator cells. Coloured circles represent individual neurons and internal arrows show the preferred direction of that cell. Purple units in the CPU1 layer represent CPU1b cells which close the ring of the left (light blue) and right (dark blue) CPU1 columns respectively.

The neural circuit for PI (see fig. 4.4) was adopted from Stone et al. (2017), and is an anatomically constrained model of the insect CX, that is, every neuron type and

connection in this model has been mapped in the insect brain. The inputs to this system are the speed cells (TN) (described in section 4.4.2.3) and the global heading, which is encoded by a set of 8 direction cells (TB1). The direction cells in insects receive input from their polarisation compass system Homberg et al. (2011), and exhibit a bump of activity that is correlated with the agent's current heading Seelig and Jayaraman (2015). This activity can be modelled as a discretised (8 samples, 1 per cell) phase-shiftable cosine function $(-\pi, \pi)$, where the phase denotes the agent's azimuth in global coordinates.

The next layer consists of 16 memory cells (CPU4), split into left and right parts according to the input speed cell. Excitatory or inhibitory action is applied to each memory cell, at a rate proportional to the output of the relevant speed cell, according to the signed magnitude of the vertically aligned heading cell state (connectivity shown in fig. 4.4). The activity of each cell thus reflects the total translation that has occurred up to that time, i.e., they collectively represent a constantly updated vector pointing back to the origin (a home vector). Specifically, the activity across the left and right sets of memory cells will also be a discretised cosine function where the phase represents the home direction, and the amplitude represents the range.

The third layer of 16 steering cells (CPU1) is also divided into left and right sets. These receive input from the memory cells, shifted by one column to the left or right respectively. They are also inhibited by the direction cells. Consequently, the relative activity of the left and right set of steering cells will reflect whether turning left or right would more closely align the two cosines: the current heading direction; and the home vector direction. Hence homing can be achieved by using their output to control steering.

This circuit is implemented as a firing rate neural model (see Stone et al. (2017) for full details) on the MAV's onboard PC, using the magnetometer for the direction cell input, and mapping the steering cell output to the yaw rate. We additionally extract a range estimate from the current state of the memory cells, and attenuate both the steering command and translational speed $\propto$ 1/range when coming within 10m of the origin. This produces an effective 'stopping' behaviour at the estimated home location, preventing continuous search. However, as yet there is no direct neural evidence to support this hypothesised output from the memory cells.

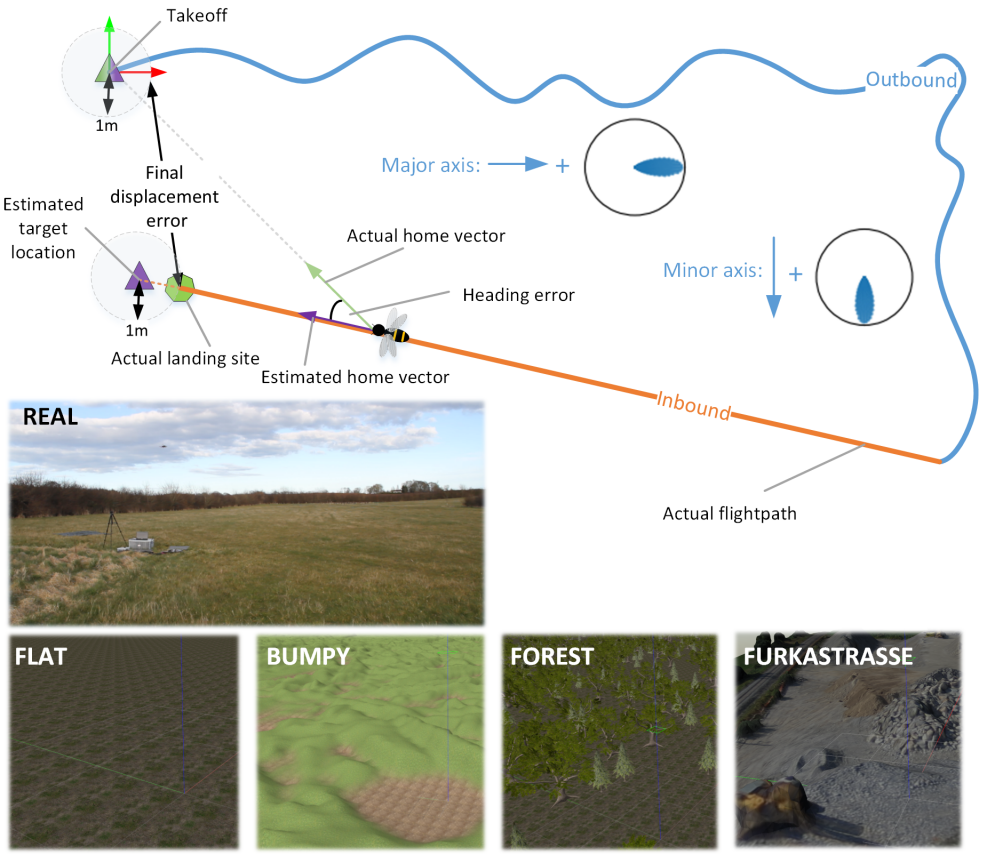


Figure 4.5: Top: the general test procedure, and a photograph of the **REAL** test world (GPS:55.763377, -2.238153). Below: renderings of the simulated environments where **FLAT**: perfectly level, textured plane at z=0 and normal to the XY plane in the world coordinate system. **BUMPY**: flat with a depth map in the interval Z=0→2m according to the depth profile inferred from a depth image of a natural scene. **FOREST**: flat but with 500 randomly placed oak and pine trees. **FURKASTRASSE**: repurposed mesh from https://wingtra.com/ of a natural scene containing asymmetrical ramping sections. Furkastrasse is also reskinned with a highly textured soil image to form **Furkastrasse retextured** (not shown).

4.4.4 Test procedure

A schematic of the general test procedure is shown in fig. 4.5, and the different test worlds (real and simulated) are depicted. The profile of the outbound route is designed to meet the following criteria: each flight path should be constrained in the overall direction to enable repeatability, but should not be identical; and the inbound route should cover different terrain to the outbound route. Consequently we model the outbound route as an L-shape with the major axis being twice as long as the minor axis.

At 2 second intervals, a random sample from a Von Mises distribution ($\kappa = 10$) was added to the heading setpoint in order to to give the outbound route a sinuous profile. Using this configuration, the impact of a particular parameter can be evaluated by repeating a given test several times with different preselected parameter values. Unless otherwise specified the test was repeated 10 times for each parameter with a default outbound duration of 100s. The aircraft was commanded to fly at 5m above the ground using the range sensor. The trial is terminated when the memory cell range estimate falls below 1m, at which point the trial ***final displacement error*** is calculated as the horizontal euclidean distance between the EKF2 and CX model position estimates. The absolute ***heading error*** was also calculated throughout each trial as the minimum angle between the memory cell and EKF2 homeward bearings.

A second mode of motion was used when testing the effect of camera pitch, because the optic flow induced by yaw rotations is no longer cancelled out if the view direction is not parallel to the ground plane. We hence separate translational and rotational optic flow by splitting the agent's motion into 4 second translations followed by rapid, saccadic rotations, during which optic flow is disregarded. This follows evidence of such motor patterns and visual suppression in flies Tammero and Dickinson (2002).

4.5 Results

4.5.1 Accuracy of ground speed estimation

The accuracy of the optic flow ground speed estimate (transformed to body frame coordinate system), compared to the EKF2 estimate, is shown for two flights, first without stabilisation by the gimbal (fig. 4.6a) and then with gimbal stabilisation enabled (fig. 4.6b). It is clear that the stabilisation is essential to reduce a high level of noise in the estimate. Some large deviations remain after stabilisation but overall a close estimate of the actual ground speed is obtained. By resampling a single image sequence at different spatial resolutions before processing in the visual pipeline of fig. 4.2a we tested the minimum image resolution that still produces acceptable speed estimates. A brute force search strategy was used to optimise the Farnebäck *window size* parameter for each resolution. fig. 4.6c shows a good estimate is still obtained with only 3° acuity, which is commensurate with the eyes of flying insects Taylor et al. (2019).

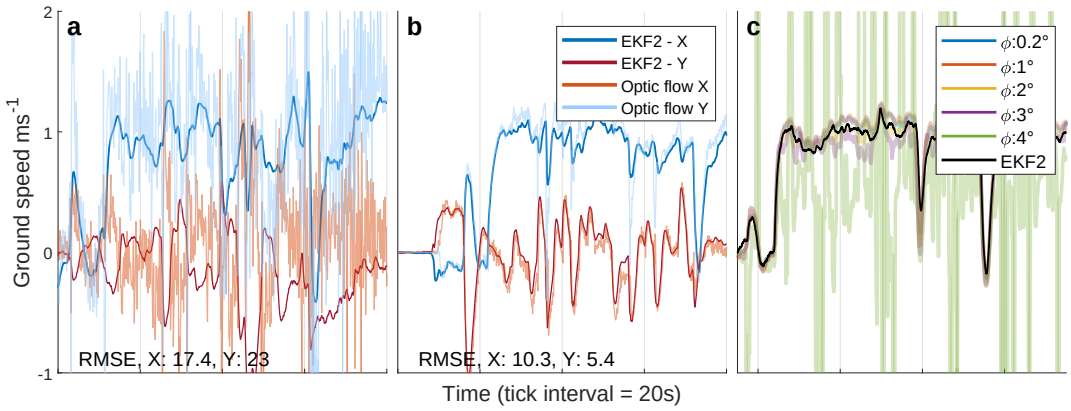


Figure 4.6: Optic flow ground speed estimates vs. EKF2: **a** without stabilisation; **b** with stabilisation; **c** for different visual accuities, ϕ = °/pixel

4.5.2 Effect of terrain and outbound distance on homing performance

We evaluated the homing performance of the CX model for three different outbound distances (50,100,150m) on the MAV in the real world, and in each of the simulated worlds described in fig. 4.5. The results are summarised in fig. 4.7a. For the real MAV and the flat simulated environment, the homing procedure always returned to within 10m of the starting point across all trials which is comparable to the accuracy of a GPS system. As expected, the final displacement error and its variance increase as a function of outbound distance. Note that even in the flat world there is a final displacement error bias suggesting that there is scope for improvement in the performance of the overall system. A linear interpolation of the real dataset yields the equation $y = 0.011x + 1.74$, indicating that the mean performance decreases by 1.1m per 100m. An interpolation of the worst performing values produces a slope of 1.5m/100m.

There was a small performance decrease for the bumpy and forest environments, which feature smooth and discontinuous height variations respectively. The performance was severely impacted for the natural Furkastrasse environment, but this was found to be due to some featureless areas of the environmental mesh. When the mesh was reskinned with a uniformly textured pattern, performance is similar to the other non-uniform height worlds. The forested world was the worst performing textured world, suggesting high depth discontinuities are more problematic than smooth slopes. It should be noted that in order to achieve this performance in the forest world, the altitude control had to be adapted to maintain a fixed height above the ground, rather

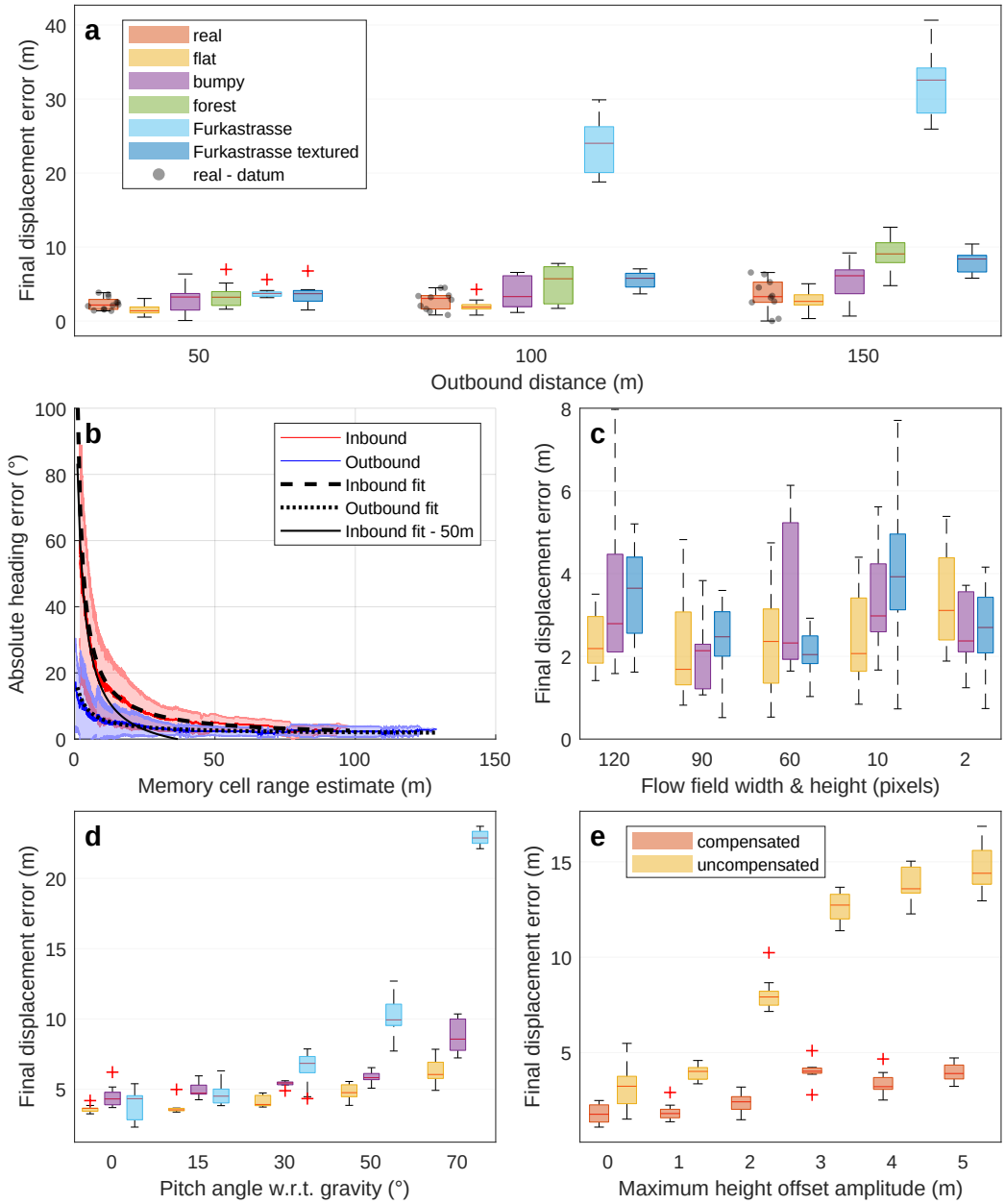


Figure 4.7: Homing performance of the CX model. **a** For most test conditions, final displacement error is less than 10m, increasing gradually as outbound journey length increases. **b** Heading error decreases with distance. Central line shows the moving average value for a pool of 10x150m flights in the flat world. Shaded regions show 1 standard deviation from the mean. **c** Reducing the field of view of the camera does not affect homing performance. **d** Increasing camera pitch angle negatively affects homing performance, especially in uneven terrain. **e** Removing height compensation negatively affects homing performance.

than using range sensor readings. Insects sometimes fly over gaps without adjusting their height but the ability to maintain a fixed height relative to the ground while flying over a tree canopy is speculation.

4.5.3 Directional certainty increases with distance from the origin

Previous work predicts that while the positional uncertainty of a noisy PI circuit increases with outbound foraging range, directional certainty increases Wystrach et al. (2015) . The first of these assertions is supported by fig. 4.7a and in fig. 4.7b the second is apparent. Here the moving average mean heading error is plotted for all range ordered samples of the *real* dataset as a function of radial distance from the trial origin. From geometric considerations the heading error should be $\propto$ positional error/range Wystrach et al. (2015); hence inbound heading error will be lower for shorter outbound distances when less positional error has accumulated (see 50m curve in fig. 4.7b) but it is evident that for all distances, the inbound heading error increases rapidly from under 10m. This suggests accurate relocation of the exact origin would require an alternative "drift free" cue such as local visual information.

4.5.4 A reduced field of view does not affect homing performance

We investigated whether the size of the field of view used to estimate ground speed would affect the accuracy of homing performance, by cropping the input before the final summing step in the visual pipeline. In fig. 4.7c it can be seen that there is no tangible impact for any of the tested environments, even when the speed cell ground speed estimate is based on a visual field of only 2x2 pixels. An expanded field of view could potentially be detrimental given the proportional relationship between depth and optic flow, as in a natural scene this would increase the likelihood of encountering non-ground-plane objects in the visual field. Conversely, wider field of view measurements could provide a filtering and/or voting mechanism to reject flow field outliers.

4.5.5 Ventral views provide more robust optic flow information

In order to understand the practical implications of sampling optic flow from different view elevations, the view direction of the vision sensor was systematically raised. In simulation, the single downward facing camera was replaced with a pair of time synchronised cameras that were rotated -45° and 45° in yaw respectively. These cam-

eras were then pitched equally upwards by 0,15,30,50 or 70°. It was necessary to use saccade mode (see section 4.4.4) for these trials to prevent a catastrophic reduction in homing performance in the Furkastrasse_retextured world. The results in fig. 4.7d show that increasing the pitch angle reduces homing performance in all cases to some extent. This can be explained by the greater variance of μ weighting factors for higher pitch angles (see fig. 4.2) resulting in those areas with higher weighting contributing more noise to the speed calculation. However, given the use of saccade mode, performance degradation due to pitch angle increase in the flat and bumpy worlds is modest in comparison to the natural scene which features large asymmetric ramping sections that cause large divergences between the assumed and actual depth values.

4.5.6 Height compensation is required for successful homing

Since neither the use of height estimation or maintenance of a constant ground height is a given for flying insects, we explored the impact of removing height input in the flat world, while introducing a randomly varying height profile along the outbound journey. Specifically, height profiles were generated by adding a random positive offset to 10 equally spaced points along the outbound journey (z_offset = U(0, maximum z offset)), and then making a linear interpolation along this height series according to the time into the outbound flight. The results, included in fig. 4.7e, indicate that even with a modest maximum height offset of 2m, homing performance is severely impacted if no height compensation is used in the visual processing pipeline.

4.6 Discussion

We have implemented the CX model presented in Stone et al. (2017) on an autonomous MAV. This anatomically constrained neural circuit integrates ground speed in different compass directions to maintain an estimate of location with respect to an origin, providing a steering output to return home. Here we have introduced a biologically plausible matched filter to obtain ground speed from ventral optic flow. We show that in real world testing, which includes asymmetric inbound/outbound routes, random yaw saccades, and dynamic lighting and weather conditions, the worst case final displacement error increase is 1.5m per 100m of outbound distance. Thus this bio-mimetic system constitutes a medium range navigational aid with comparable accuracy to a GPS module over the range of 1km.

This accuracy could bring a flying insect near enough to its goal to find it by search, but the decreased heading accuracy at short range (in fig. 4.7b) suggests that efficient relocation of the nest requires alternative mechanisms. Moreover, some species of bee are known to commute several kilometers von Frish K. (1967), so it seems likely that additional guidance cues such as odour and vision are used over longer ranges. In future work we will examine biologically plausible ways of extending the navigational range and improving close range homing capability of our biomimetic platform using *drift free* visual cues.

Our efforts primarily focused on developing a biologically plausible visual ground speed estimation system, consisting of a matched filter based optical flow subsystem. We have demonstrated that the acuity and frame rate of this system could be as coarse as 3°/pixel and 10Hz respectively. Best results were achieved with 1) the camera directed towards the ground, 2) mechanical stabilisation against aircraft pitch and roll, and 3) height compensation. No discernible improvements in homing performance were observed in relation to an increasing field of view of the matched filter output, which could explain why neurons with a receptive field commensurate with a translatory optic flow matched filter have remained elusive in comparison to their rotational counterparts Longden et al. (2017).

In future we aim to replace the magnetometer heading sensor with a biologically plausible sky compass Gkanias et al. (2019). We also need to consider a biologically plausible mechanism for height estimation as this appears to be necessary information for accurate PI. An alternative approach would be to implement an optic flow regulated autopilot Serres and Ruffier (2017), although this may not compensate for different visual terrain observed on the inbound route versus the outbound route. We suggest that further neurophysiological investigation of the CX speed cells (TN) using stimuli representing surfaces moving directly below the animal at different depths would also provide significant insight into this question.

4.7 Additional Material

4.7.1 Effect of topography on height and speed estimates

In section 4.5.2 it was demonstrated that matched filter based optic flow speed estimates perform well in flat worlds but the accuracy of this method deteriorates in uneven worlds. In this section we compare the time series of the speed estimation process in a flat and an uneven surface to see how these errors arise. Figure 4.8 contains a plot of the optic flow in a flat world, where the range sensor reading closely matches the height of the aircraft (the slight offset is because the range sensor is not in the same location as the flight controller). In this scenario, the matched filter output tightly follows the aircraft's actual speed. Large errors are occasionally experienced over a short time scale, these are typically caused by high angular velocities around the yaw axis. In the forest world (see fig. 4.9), where there are several height discontinuities, the matched filter output tracks the biorobot's velocity with a lower degree of accuracy. Errors are at a maximum when the aircraft passes over a tree object. In these regions the average error between the depth weight matrix prior and the actual depth of the scene is large.

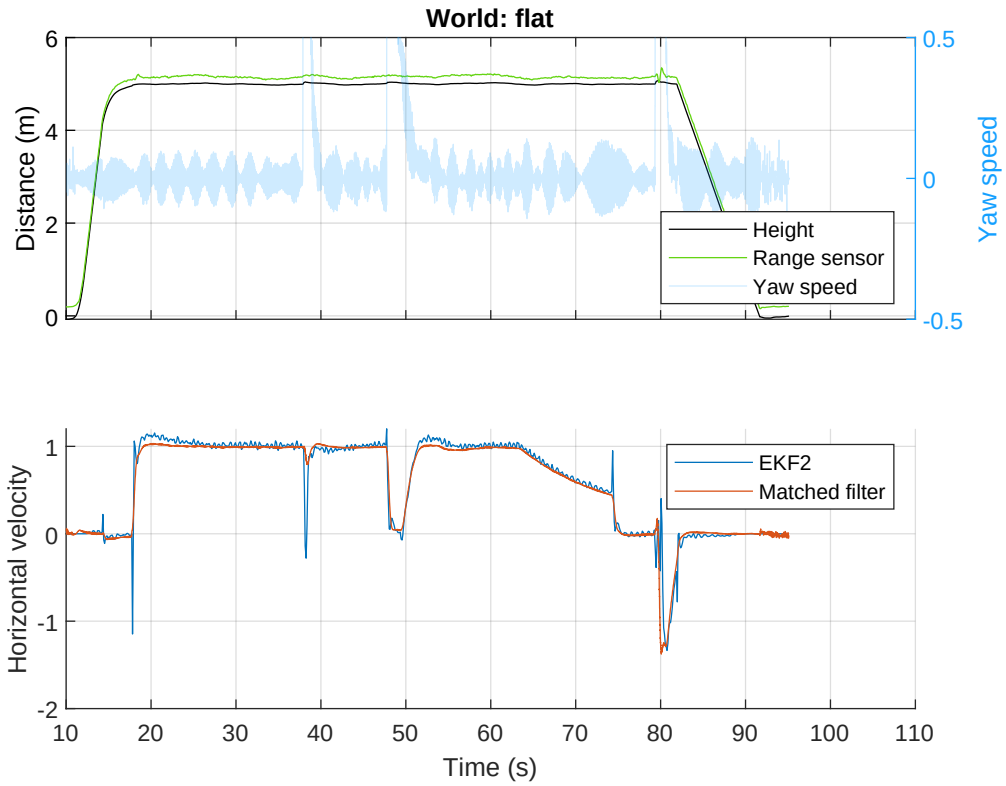


Figure 4.8: Time series of height and speed estimates in the flat world. The upper panel shows the biorobot's height and yaw state estimation output. The lower panel shows the EKF2 estimate of vehicle's velocity (our ground truth) and the matched filter's vehicle velocity estimate based on the optical flow input.

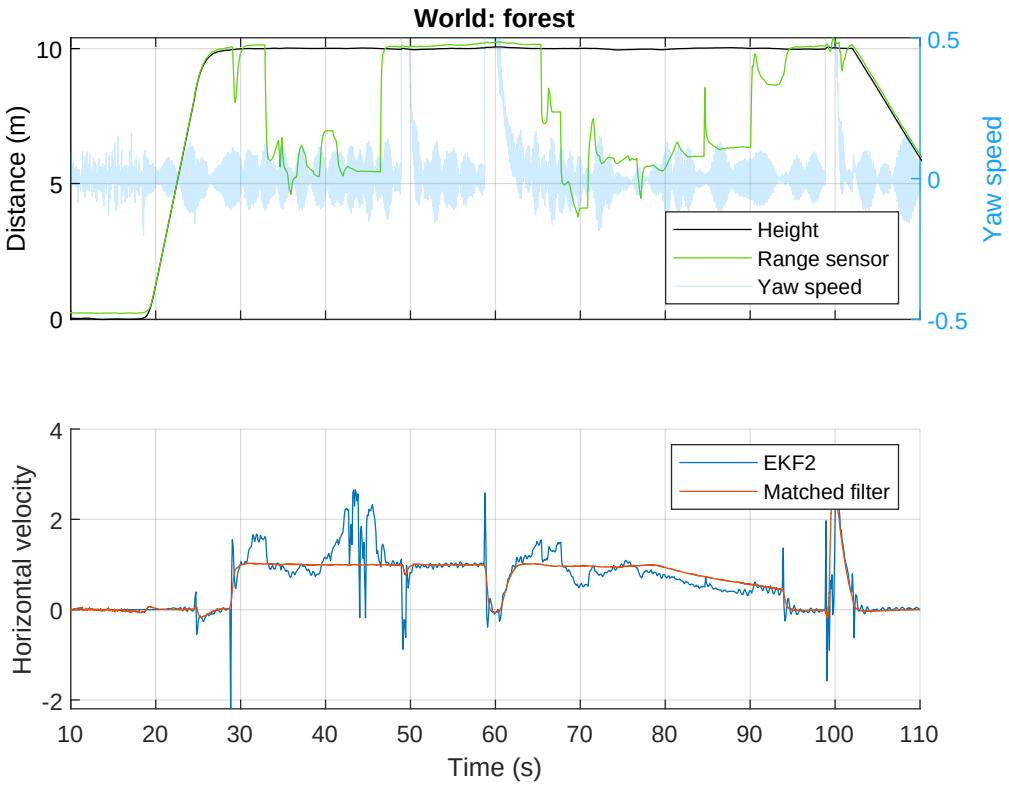


Figure 4.9: Time series of height and speed estimates in the forest world. Plot details are consistent with fig. 4.8

Chapter 5

Complex wavelet structural similarity for view based image matching

5.1 Introduction

The optical flow path integration model presented in Chapter 4 had a final position error in the order of tens of metres. However precise the sensing apparatus of a path integration or dead reckoning system is, its position estimate will drift further from the true value over time. In order to mitigate this problem, roboticists typically augment motion sensing systems with drift free sensing modalities such as GNSS, motion capture systems or vision. In this chapter the use of vision as a drift free navigation sensor is investigated.

5.1.1 Why vision?

Since Tinbergen demonstrated that vision is the dominant cue for ground nesting wasps returning to their nests (Tinbergen, 1972), biologists have sought to derive a functional model of this behaviour (Stürzl et al., 2016). Efforts towards this have focused on the short range orientation flights that foraging flying insects always perform when they first leave their nest. The prevailing theory is that orientation flights provide a visual snapshot of the nest area that can be used to guide future return paths (Stürzl et al., 2016). The trajectories of these flights have been recorded with high spatial and temporal resolution stereo camera rigs and their structure is well described for several Hymenopteran species (Jayatilaka et al., 2018; Philippides et al., 2013; Stürzl et al., 2016). In ground nesting wasps, these learning flights are characterised by the

insect turning back and looking at the nest while flying outwards in a series of arcs (see fig. 5.1a). During this process the insect remains within a 50cm Euclidean distance from the nest entrance, such flights are hereby referred to as *short range* learning flights.

After the initial short range learning flight, it is common for wasps to embark on several subsequent, longer range orientation flights. Anecdotally (Spradbery, 1973) (page 133), and from personal observations, these are reported as a circling above the nest at a height of 5-10m followed by a short horizontal trip of approximately 10m. The trajectories of these *medium range* (<15m) flights have yet to be recorded with any precision. Although *long range* (15m< length <200m) exploration flights have been recorded with harmonic radar systems and are also often characterised by direct outbound and inbound journeys with a constant direction of travel (Capaldi et al., 2000; Degen et al., 2015, 2016; Osborne et al., 2013) (see fig. 5.1b).

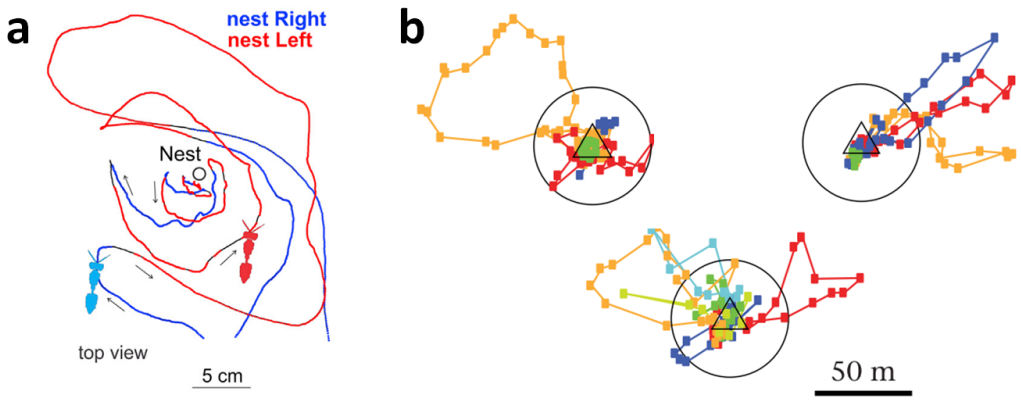


Figure 5.1: **a** Short range learning flights of ground nesting wasps adapted from (Stürzl et al., 2016) with permission from Elsevier. **b** Harmonic radar recordings of the long range learning flights of bumble bees adapted from (Degen et al., 2015) with permission from Elsevier. While the intricacies of the flight path are not captured, the route followed on a given inbound and outbound flight pairing appears to be relatively consistent.

Experiments on both terrestrial (Zeil et al., 2014) and aerial (Degen et al., 2015) foraging insects reveal that experienced animals are able to directly return to their nests when they are displaced to novel locations. This ability is commonly explained by the use of gradient descent procedures operating on image difference function search spaces. Conceptually, the insect stores a holistic view at a salient location. At a future instance in time, the insect can move towards the desired location by minimising the difference between its current view and a reference view (explored in detail in section 2.5.2). However, foraging insects are known to inhabit landscapes where single

reference images provide limited navigational information. Dense forests for example, where the attractor basin of each view is small. Or conversely, flat featureless terrain where visual cues are distant and therefore the insect is required to translate large horizontal distances if it is to experience robustly discernible changes in *view space*.

5.1.2 Towards visual route following on an aerial platform

In (Baddeley et al., 2012), the idea of visual route following was proposed as a means by which terrestrial animals could navigate through cluttered environments. The central idea is that a dense sequence of views is stored by the commuting insect while it is en route to its nest from a suitably lucrative feeder site. When recapitulating this route, the insect periodically rotates its head through a large angle (this behaviour is known as *scanning*) and adjusts its course so that it is headed in the direction of greatest familiarity. While contemporary visual route following models predict the behaviour of foraging desert ants more accurately than visual homing models (Baddeley et al., 2012), several implementation issues persist with current visual route following models:

- **lack of robustness** - even in simulation environments, visual route following agents become 'lost' multiple times per route (Ardin et al., 2016).
- **excessive visual scanning requirements** - the rotational image difference function, key to the success of most terrestrial navigation models, requires the insect to scan its head at short intervals (typically every 10cm). While scanning is observed in ants, there are typically large sections of route where the animal does not scan. Alternative solutions have been proposed for this (Kodzhabashev and Mangan, 2015; Le Möel and Wystrach, 2020; Sun et al., 2019), but more work is required to confirm the viability of these methods in the real world.
- **arbitrary view capturing** - current models lack a biologically plausible mechanism for the triggering of view capturing.

Given these problems, extending the same paradigm directly to a higher dimensional space would seem doomed to failure. However, previous studies have hinted that a downward facing aerial perspective is a useful substrate for visual navigation. In (Gaffin et al., 2015), insect inspired algorithms were successfully applied to the tracing of tortuous routes through a geographic map rendered with real-world satellite imagery data. A perspective that is inherently constrained against vertical motion and roll/pitch attitude rotations. It is also known that the flight paths of both insects (Menzel et al.,

2019) and birds (Lipp et al., 2004b) have a tendency to track flat linear features such as rivers, hedgerows and roads. This indicates that at least part of the visual field of these animals is directed significantly below the horizon.

5.1.3 Why use a downward view?

To build a rationale towards why a downward perspective may be effective, consider an ant traversing a natural scene, its head is just millimetres above the ground. At this height, a downward view covers a small area of ground. As the insect moves towards a goal (i.e. translates on the XY plane), its angular velocity with respect to the ground is large and therefore the downward view alters drastically, even for small translations. It is not surprising therefore that terrestrial visual navigation models attend visual information primarily at the horizon (Wystrach et al., 2011). A potential problem with the information derived from a horizon-centric view is the risk of aliasing. That is, while an agent is headed in a relatively constant direction in relatively open ground, views towards the horizon are likely to be similar across a large range of locations.

Now consider a flying insect several metres in the air. As it gains height its downward view spans an increasingly large area of ground. As the insect translates steadily in the XY plane, the content of this downward view shifts gradually across the retinular array of the insect's eyes. The question arises, does sufficiently stable visual structure exist in natural scenes with this type of view? If so, are the properties of the compound eyes of flying insects capable of exploiting this?

5.1.4 View matching paradigms

It is known that insect view memories are related to the apparent size and shape of landmarks rather than feature constellations (Cartwright and Collett, 1983; Narendra, 2007; Wehner et al., 1996). This precludes many non-retinotopic approaches used in computer vision such as the visual bag of words. The mean squared error (MSE) has become the de facto method for image matching in insect navigation simulation studies. In the real world the MSE is a poor choice because its output is significantly modulated by exposure and lighting variations (Wang and Bovik, 2009). Previous works have investigated this effect in natural scenes (Stürzl and Zeil, 2007) and contrast adaption mitigation schemes are implemented as a matter of course in bio-mimetic robotic platforms, e.g. (Kodzhabashev and Mangan, 2015). In the compression algorithm evaluation literature, structural similarity (SSIM) and complex wavelet structural similarity (CWSSIM) are

widely cited as the state-of-the-art methods for evaluating image similarity according to perceptible changes by the human vision system. CWSSIM is also reported as being invariant to small translations and rotations. This property is of considerable interest for a view matching scheme where test sites are unlikely to exactly coincide with training locations. CWSSIM is also inspired by the isotropic, bandpass characteristic of the V1 cells in the human vision system and therefore has a degree of biological plausibility but its efficacy has yet to be investigated by the invertebrate vision community.

5.1.5 Outlook

As previously motivated in Chapter 2, bio-mimetic modelling has proved an invaluable means of evaluating hypotheses relating to animal behaviour. The autonomous MAV presented in Chapter 3 provides an ideal platform to probe the questions developed in this section. The chapter goals are formally outlined below:

1. Investigate the efficacy of a biologically plausible ventral/ventrofrontal visual route following system deployed on an aerial platform.

2. Establish whether natural scenes provide sufficient information for an anatomically constrained agent to navigate to within the a short range (<1m in the XY plane) of a site of interest.

3. Establish an effective view matching procedure for biologically plausible visual navigation models that are deployed on a real-world aerial agent.

5.2 Methods

5.2.1 Concept overview and assumptions

A high-level overview of the concept is presented in fig. 5.2. Central to this scheme is the notion that views directed towards the ground are well suited for localisation in the XY-plane of a Euclidean coordinate system. It is hypothesised that the medium range learning flights observed in foraging flying insects provide a useful opportunity for these animals to memorise a breadcrumb trail of downward facing views between the outer boundary of the medium range navigation zone and the corresponding site of interest (destination in fig. 5.2). According to (Zeil et al., 2003), given sufficient world view variety and an effective view sampling strategy, systematic scanning of the environment in the XY plane with an effective view matching function should produce a manifold like that of fig. 5.2C. This search space is characterised by a prominent ridge of familiarity along the route, with a noisy response of significantly lower magnitude in all other locations.

This work examines the possibility of following a familiarity ridge line in a scan free fashion using lateral oscillations. Such a paradigm has been explored for other insect behavioural modalities, chemotaxis (Wystrach et al., 2016b) being a prominent example. Since the insect's navigation toolkit features global compass information, the question of which direction to travel along the ridge can also be hypothetically resolved. Furthermore, flying insects are know to move holonomically (specifically, the insect's XY velocity vector and body yaw heading can be offset from each other by an arbitrary angle) which forms an ideal basis for transverse oscillations with respect to the familiarity ridge while maintaining a globally consistent heading direction.

In order to test the feasibility of this concept, the arrangement is tested in a closed loop behavioural model on an anatomically constrained (at the algorithm level) robotic platform (*the biorobot*). The MAV developed in Chapter 3 is used as a hardware platform and a view matching scheme is developed to support this work. A number of assumptions about the capabilities of flying insects and behavioural simplifications were made throughout this process. Relevant assumptions and implementation details are outlined in the remainder of section 5.2.1.

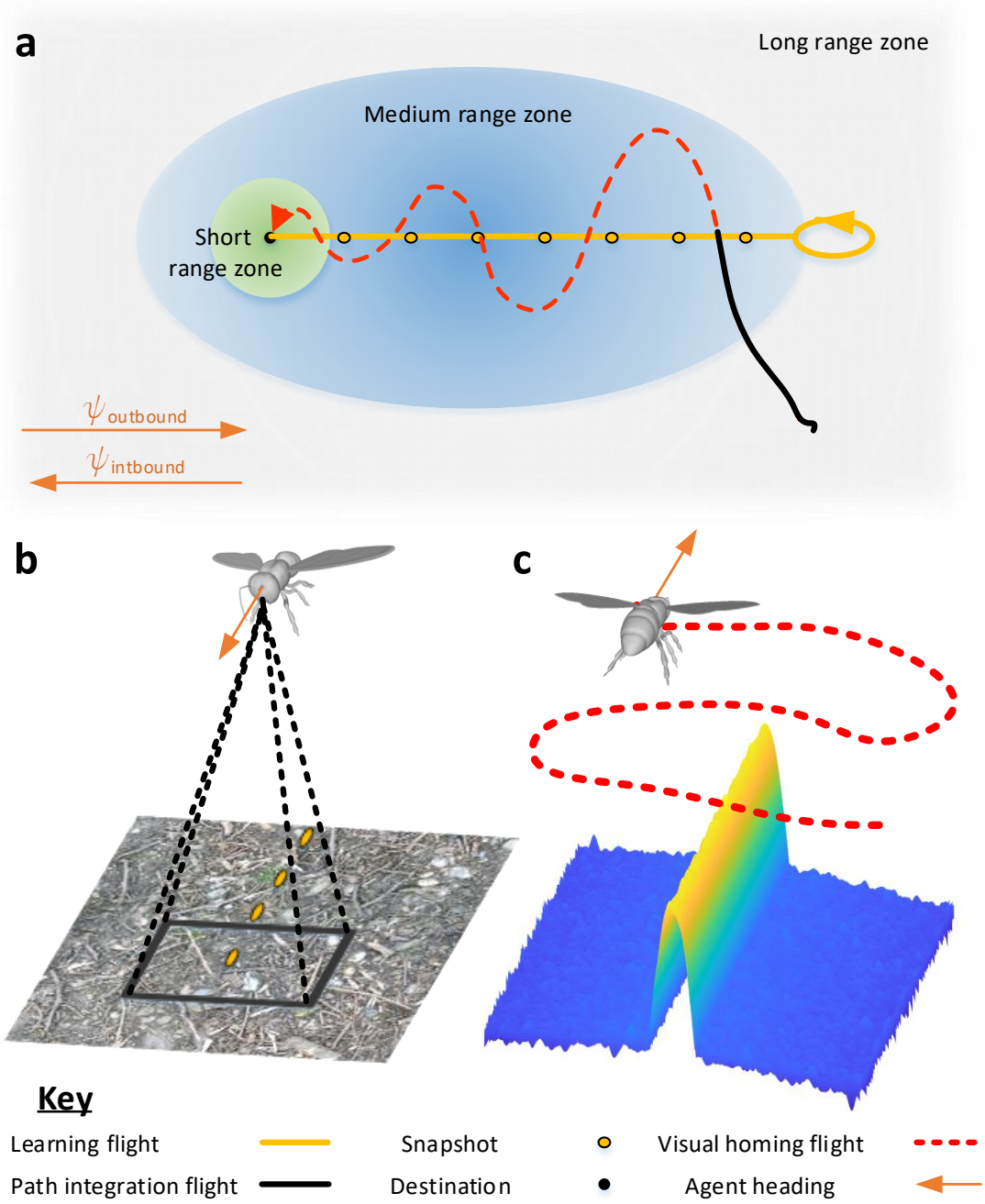


Figure 5.2: **a)** Top down view outlining the main components of the aerial visual route following concept **b)** Ventral views are acquired at regular intervals during an outbound route **c)** The search space of an effective view matching scheme features a prominent ridge of familiarity (yellow→familiar,blue→novel) that can be used to guide an inbound journey.

5.2.1.1 Route following procedure

A route following procedure is required in order to steer the biorobot along the familiarity ridge. Line following is a simple, commonly adopted solution for this kind of problem in robotic applications. This involves the following of a pre-laid track of high-intensity signal that is bordered by a low-intensity surround (or vice-versa). The biorobot requires a sense of which direction to steer towards in order to remain on the track. For a conventional robotic system this typically involves an array of two or more spatially separated sensors. When a line is detected by one of the sensors, the robot rotates in the direction that prevents the sensor from crossing the line. Since the eyes of insects are not sufficiently spatially separated, an insect must use memory to monitor spatially disparate locations over time as explored in fig. 5.3.

In this work the transverse oscillation method (fig. 5.3B) was selected over the gradient descent method (fig. 5.3C). This was largely because insect flight appears to feature lateral oscillations but also to avoid the issue of confounding variations in familiarity longitudinal to the flight path. Mental image shifting (i.e. auto-correlation of the current view with each view in memory) was considered as a means of coping with this. Given sufficient image overlap a steering command could be inferred from the required image pixel shift. Ultimately this configuration was not pursued due to its computational demand in the context of a perfect memory pipeline and lack of clear biological plausibility. The full implementation of a transverse oscillating route following (TORF) procedure is developed in section 5.2.6.

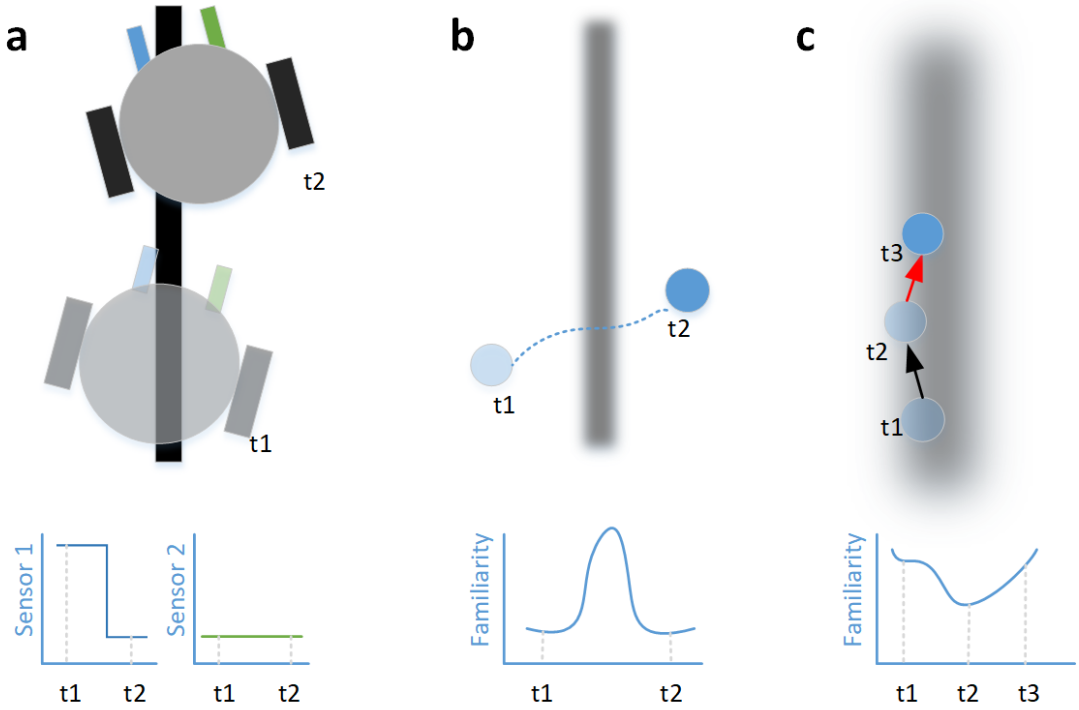


Figure 5.3: line following approaches. **a** *Multi-sensor* - For scenarios when there is a solid line and spatially separated sensors as is common in robotic setups, when one sensor detects the signal (t1) the robot rotates in order to avoid departure from the line (t2). **b** *Transverse oscillation* - When only a single sensor exists, the spatial separation can be obtained over time, the biorobot stores the peak signal strength and direction over a particular time window (t1→t2) and the search of the next pass is centered at the maximum signal value **c** *Gradient descent* - For signals with a wide, smooth and longitudinally homogeneous gradient, the biorobot can use gradient descent to remain on the path. **b&c** as presented here are contingent on knowledge of which way to travel along the path

.

5.2.1.2 Vision system

Due to current theories about hyper-acuity, the spatial resolution of compound eye bearing animals is a controversial topic and remains unspecified. It is however clear that this resolution must be related to the inter-facet angle of such eyes and a conservative estimate of the spatial resolution of the visual system for a given view direction is simply the average inter-facet angle in that region. As with any insect-mimetic vision system, the images generated by a standard modern digital camera must be blurred and downsampled in order to match the properties of compound eyes. In fig. 5.4 the results of a recent anatomical investigation into the resolution of the eyes of two relevant species of insect are included. Based on this information, it is assumed that a vision system with an acuity of 3°/pixel meets this biological constraint, assuming that the minimum image elevation (bottom row of pixels in the image) is directed at no lower than 60° below the horizon. The biorobot's camera has a vertical field of view of approximately 30° so the nominal viewing angle should be -30°→-60° beneath the horizon. However, in this work, for purely practical reasons, view memories are downward facing (as illustrated in fig. 5.5) so that the inbound and outbound routes can be completed in the same trial. In order to demonstrate that the principles of TORF extend to biologically relevant configurations, a batch of experiments was also conducted with the camera pitched upwards in the biologically plausible range fig. 5.28.

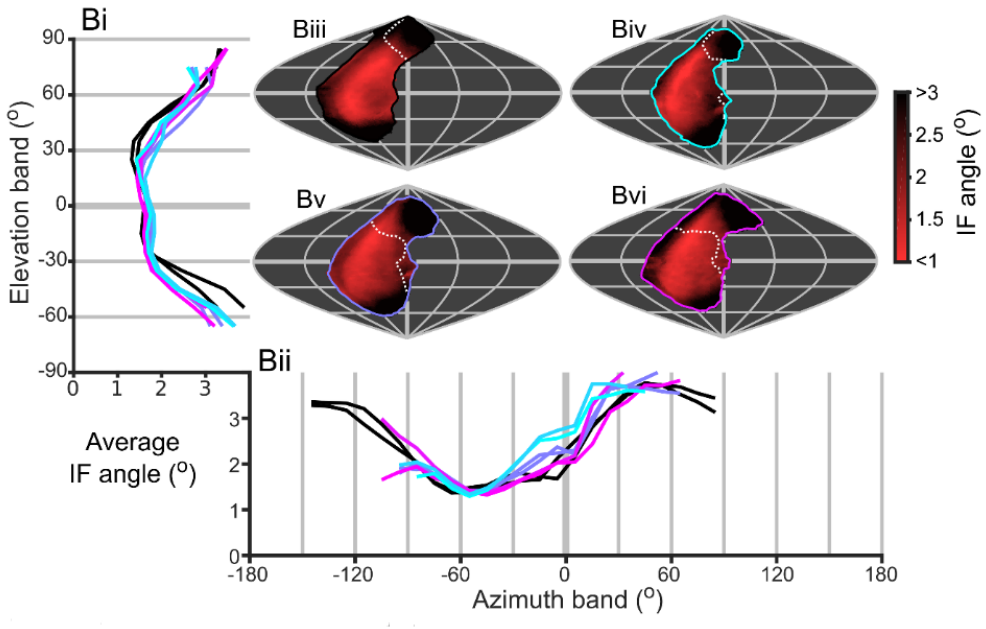


Figure 5.4: Adapted from (Taylor et al., 2019) CC BY 4.0. Estimated inter-facet (IF) angles for the eyes of several honey bees (black lines) and bumble bees (blue and pink lines) indicate that while the maximum resolution is forward facing, there is ample resolution in the ventrofrontal region for the view matching procedures used in this work

Figure 5.5: The biorobot's camera has a relatively narrow field of view and is downward facing.

5.2.1.3 View memory acquisition

The image sampling rate is set at 10Hz, a somewhat arbitrary value but one that matches (Ardin et al., 2016) and therefore memory capacity assumptions can be retained as well as being well within the refresh rate of a flying insect's vision system. By default, the biorobot travels at $1ms^{-1}$ for 15s thus arriving at the boundary between the medium and long range zones. For fixed bearing outbound routes, the heading of the biorobot is commanded to be aligned with the direction of travel.

5.2.1.4 View matching methods

Several different view matching methods were considered for the view matching subsystem. Due to the resolution constraint these were all dense, holistic methods. The image comparison scheme should not produce false positives (i.e. produce a high familiarity when confronted with a novel view) but it must be insensitive to small rotations and scaling changes. While downward views have previously been extolled for specificity in the XY plane, a relatively large degree of translation invariance in the x and y dimensions of view space is also desirable. This is because the width of the ridge and the number of snapshots required to make it continuous longitudinally are both dependent on the catchment area of each snapshot. Clearly, if this area is small, the number of views that are required to be sampled and memorised in the learning phase becomes unfeasible.

5.2.1.5 Head stabilisation and compass information

In order to prune the 6DOF search space some assumptions are made. It is known that flying insects stabilise their heads in pitch and roll (Viollet and Zeil, 2013), as outlined in Chapter 3, the biorobot's camera is gimbal mounted and can hence also exhibit this functionality. Foraging insects are also known to be capable of discerning their heading with respect to a geographically fixed frame of reference using the sun's location, even on a cloudy day (Heinze and Reppert, 2011). For a previously visited location this could also be achieved with panoramic view matching of the horizon, thus adding redundancy and a means of encoding the heading in the snapshot itself. By associating the images captured during a learning flight with a heading, the yaw component can also be simplified. Note that in common with the insect, the biorobot has imperfect sensing and control actuators and therefore view matching strategies must be tolerant to small, noisy changes in the constrained dimensions.

5.2.1.6 Height control

Flying insects are known to be capable of maintaining a steady height above the ground. Behavioural experiments have demonstrated that edge tracking, wide-field stabilization, and expansion avoidance altitude contribute to the fruit fly's altitude control (Straw et al., 2010). Another popular hypothesis for how this can be achieved is the use of optic flow regulation (Franceschini et al., 2007).

Here it is assumed that the animal can effectively travel at a desired distance above the ground in a previously visited location. The best performing view matching algorithm is relatively insensitive to height variation, although this is a function of ground height and familiarity threshold as explored later in section 5.3.5.

5.2.1.7 Search patterns (Finding the familiarity ridge)

The purpose of this work is to understand if TORF could be used as a strategy for visual homing. While the utility of this strategy is dependent on being able to locate the familiarity ridge in the first place, there are many existing works on searching patterns that are typical of insect behaviour e.g. (Reynolds and Rhodes, 2009). In this work, the inbound flight begins close to the end of the outbound route and the searching pattern is simply large oscillations in the compass direction 180° offset to the outbound route. More biologically relevant approaches were avoided so that the testing effort could be directed towards the visual route following system under test.

5.2.2 System deployment

The MAV platform outlined in Chapter 3 is used for this work. Two new states were required to create a visual homing state machine, the learning state and the homing state. The learning state needs no further specification than is provided in section 5.2.1. The homing state is more nuanced and its logic is developed in section 5.2.6. The addition of the view matching subsystem was also required. This is described in section 5.2.3.

5.2.3 View matching subsystem

The goal of the view matching subsystem is to generate a familiarity score for the current view, by comparing it with previously experienced, memorised views that were acquired as part of a learning behaviour sequence. The view matching subsystem is required to output a high score when the geometric distance between the biorobot's

current pose, and the pose of the spatially closest memory is small. A progressively lower score should be returned as this distance increases. This trend is limited by the availability of overlapping view frustum i.e. the subsystem is not expected to distinguish between completely unfamiliar views that are varying distances from a given view memory.

5.2.3.1 Perfect memory view matching pipeline

Perfect memory view comparison algorithms tend to outperform biologically plausible (and therefore lossy) memory schemes (Ardin et al., 2016), at least when the view information is captured with a digital camera and is stored as a complete image. In this work, the goal is to understand how natural scene information can be exploited, rather than establishing an underlying neural circuit. Therefore, the perfect memory pipeline presented in fig. 5.6 is adopted. This pipeline includes two modes of operation, test and train. In both modes, incoming images can be optionally preprocessed and/or transformed into a frequency or spatio-frequency space. In training mode, the resultant coefficients are sequentially indexed and stored in a database. The coefficients are stored in the processed state so that all preprocessing and transformation steps are only performed once when the image is acquired. In test mode, the transformed coefficients of incoming images are compared with each of the training image coefficients. The pipeline outputs the highest familiarity score and its corresponding database index. There is no biological basis for the provision of the latter piece of information but it is later demonstrated that an inference of the biorobot's proximity to the nest can help to improve the robustness of TORF.

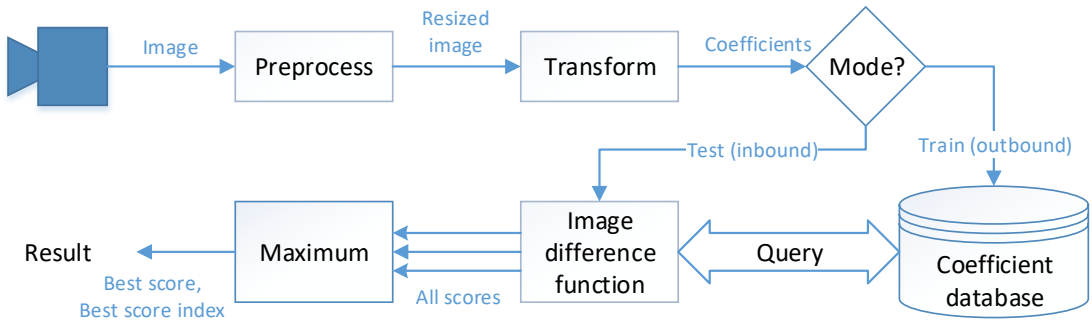


Figure 5.6: Perfect memory image processing pipeline.

The pipeline outlined in fig. 5.6 provides a flexible means of testing different combinations of transforms and comparison schemes. Each of the preprocessing and view

matching techniques that were trialled in this work are outlined in section 5.2.4 and section 5.2.5 respectively.

5.2.3.2 View matching pipeline configurations

The evaluation of view matching schemes according to their effectiveness within the context of the view matching pipeline is a major component of this work. The comparison of the CWSSIM method with current view matching methods was deemed important as was reporting the relative importance of key preprocessing steps. Each of the trialled configurations is presented in fig. 5.7. The implementation of the preprocessing and comparison techniques are detailed in section 5.2.4 and section 5.2.5 respectively.

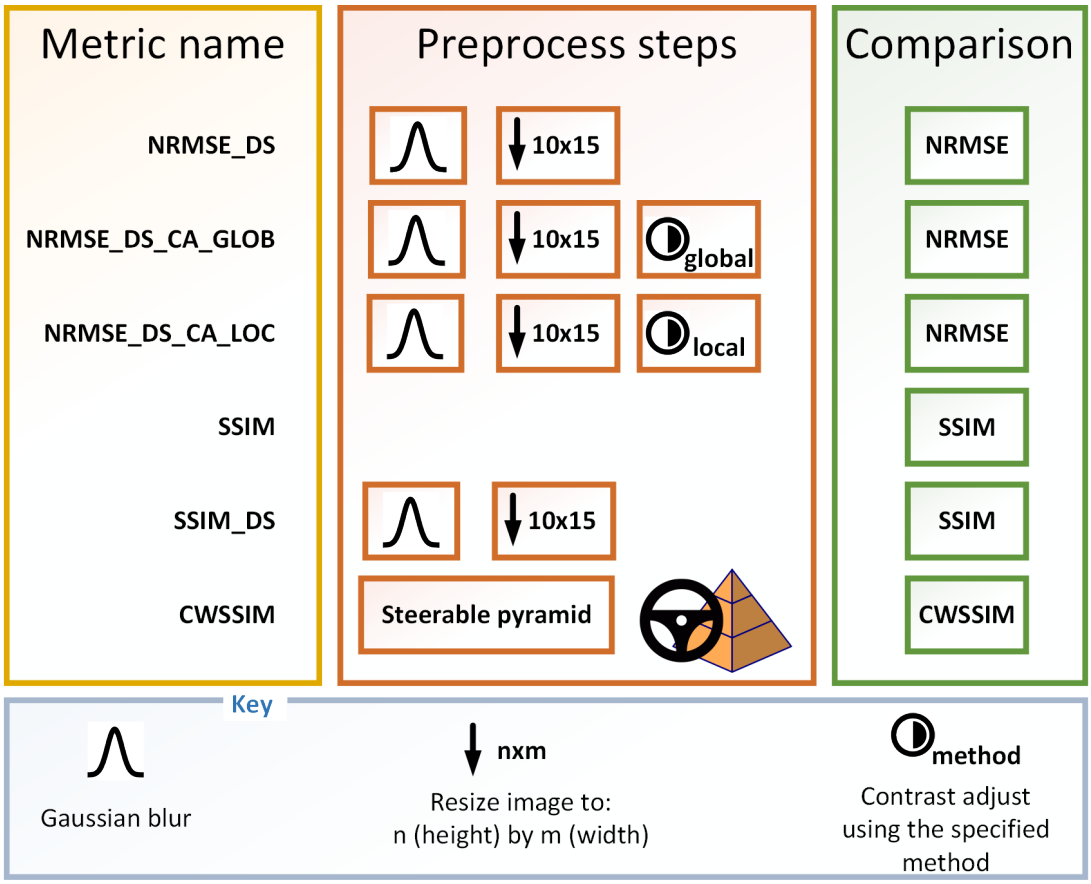


Figure 5.7: Definition of the image comparison pipelines that are used in this section. For implementation details of each operation see section 5.2.4 and section 5.2.5. Note that the ordering of the pipeline operations (from left to right) is important and should be considered immutable.

5.2.4 Preprocessing techniques

In order to mimic the low resolution, blurred characteristics of compound eyes, some well-established preprocessing techniques were employed and the implementation details are outlined in this section. For a visual impression of the image resizing and contrast enhancement procedures see fig. 5.8 and fig. 5.9 respectively.

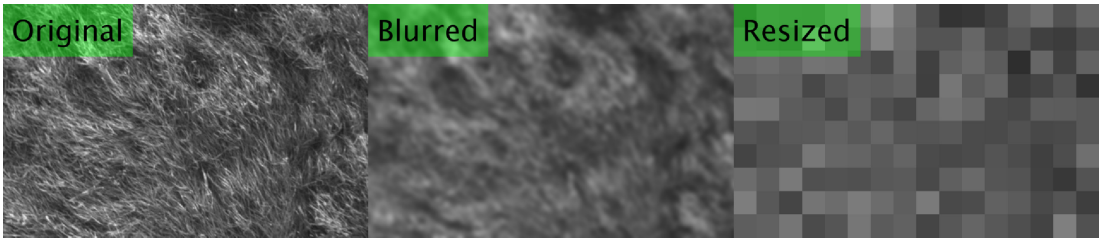


Figure 5.8: Image resizing steps for a sample downward view.

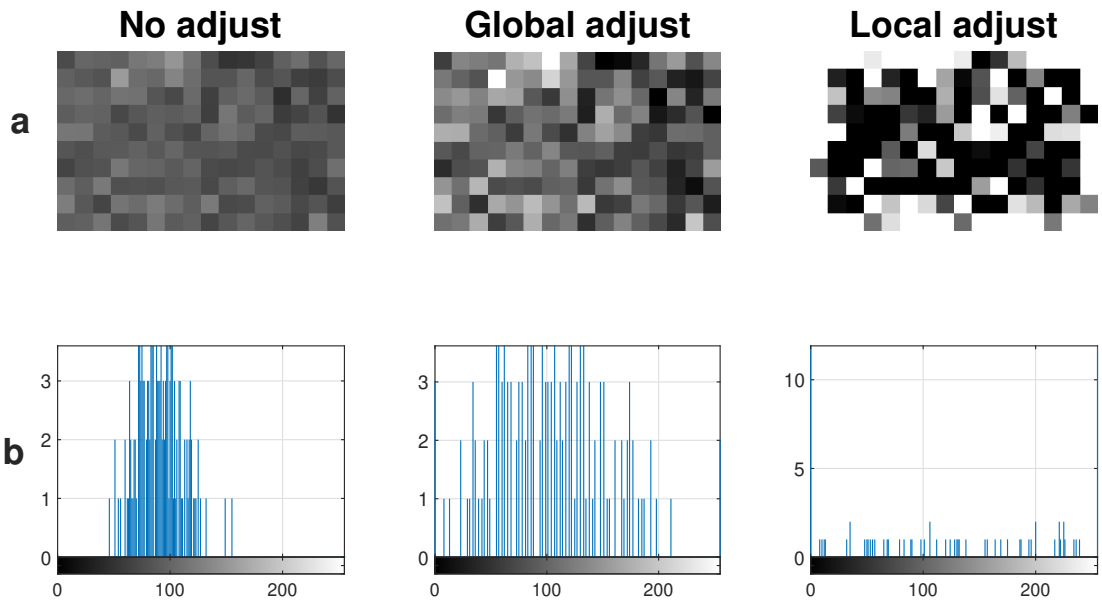


Figure 5.9: Contrast adjustment procedures applied to the resized image from fig. 5.8 **a** Image output according to the adjustment method specified by the column title. **b** Intensity histogram for pixels in the corresponding (above) image. Notice that the adjusted images use the full uint8 intensity range (0 - 255). The globally adjusted intensity histogram has the same profile as the unadjusted image histogram whereas the local adjustment measure has a modified distribution and a proportionally larger number of saturated pixels.

5.2.4.1 Image blurring

A Gaussian kernel function is a good approximation of the blurring effect caused by the acceptance angle of a given ommatidium (Land, 1997). Unless otherwise specified, the blurring factor (σ) was set at five.

5.2.4.2 Image resize

The biorobot resizes the raw camera images to 150x235 using the imag_proc nodelet in the imag_proc ROS package. For analysis, further resizing was often required. Unless otherwise specified, bicubic interpolation was performed with the Matlab® *imresize* function. Note that bicubic interpolation introduces a secondary blurring mechanism.

5.2.4.3 Global contrast adjustment

For the global contrast adjustment, a lookup histogram normalisation procedure was adopted so that the image intensities occupy the full range of the uint8 data type. Here, the *imadjust* function from Matlab® is used for evaluation in section 5.3 and the OpenCV *equalizeHist* method was used to implement contrast adjustment onboard the MAV.

5.2.4.4 Local contrast adjustment

Local contrast enhancement was achieved for an image, $I(i,j)$ with local mean ($\bar{I}(x,y)$) and local variance ($\sigma(x,y)$) as specified in eq. (5.1). The kernel size was five with the centre at the pixel under evaluation. Image coordinates are denoted (i,j) and local kernel coordinates are denoted (x,y).

$$I_{enhanced}(i,j) = \frac{I(i,j) - \bar{I}(x,y)}{\sigma(x,y)} \tag{5.1}$$

5.2.4.5 Complex wavelet steerable pyramid

The complex wavelet steerable pyramid provides an efficient wavelet multi-resolution decomposition of an image into different orientations (subbands) at different resolutions. A diagram of this scheme is presented in fig. 5.10. The decomposition is performed in the polar frequency domain where the analysing kernel can be implemented in an efficient separable format. At each pyramid level, a high and low pass filter are formulated with a raised cosine filter defined by cutoff frequencies at $\pi/2$ and $\pi/4$

respectively. The highpass band is defined by the Nyquist theorem and the lowpass removes details that will be detected at the next pyramid level. The orientated passband filters are then specified with an angular component in the polar-Fourier space. These filters were developed in (Portilla and Simoncelli, 2000) and are specified in eqs. (5.2–7).

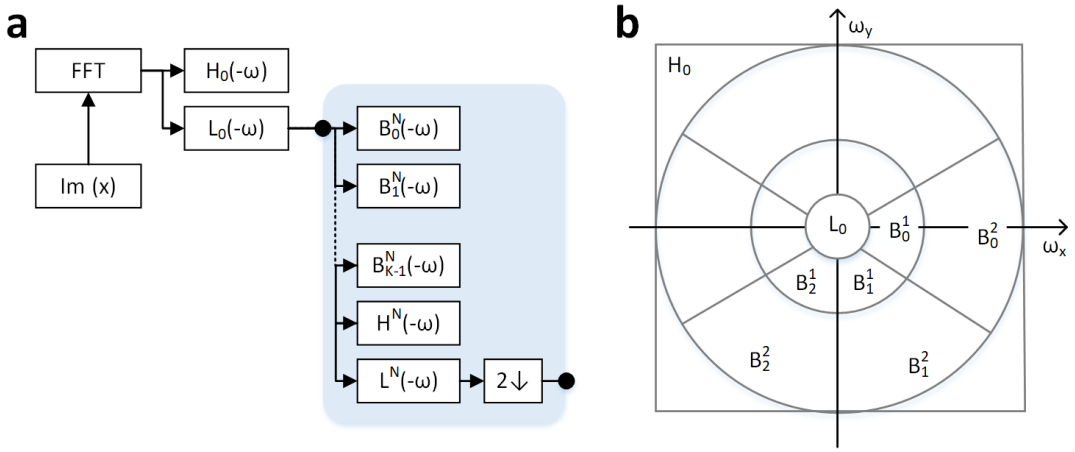


Figure 5.10: Filter bank for the complex steerable pyramid. **a)** The filter bank arrangement, for the specification of each filter see eqs. (5.2–7). The blue shaded box shows the region that is recursively processed n times (current level denoted by N). After each iteration the image is downsampled by a factor of 2. **b)** The idealised representation of an image decomposition in polar-Fourier space for three directions and two scales (K=3, N=2).

$$L(r,\theta) = \begin{cases} 2cos\left(\frac{\pi}{2}log_2(\frac{4r}{\pi})\right) & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 2, & r \leq \frac{\pi}{4} \\ 0, & r \geq \frac{\pi}{2} \end{cases} \tag{5.2}$$

$$H(r) = \begin{cases} cos\left(\frac{\pi}{2}log_2(\frac{2r}{\pi})\right) & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 1, & r \geq \frac{\pi}{2} \\ 0, & r \leq \frac{\pi}{4} \end{cases} \tag{5.3}$$

$$G_k(\theta) = \begin{cases} \alpha_k \left[cos(\theta - \frac{\pi k}{K}\right]^{K-1} & \left|\theta - \frac{\pi k}{K}\right| < \frac{\pi}{2} \\ 0, & , otherwise \end{cases} \tag{5.4}$$

$$B_k(r,\theta) = H(r)G_k(\theta), \qquad k \in [0, K-1] \tag{5.5}$$

where r and θ are the polar steerable coordinates and $\alpha_k = 2^{k-1}\frac{(K11)!}{\sqrt{K(2(K-1))!}}$

$$L_0(r,\theta) = \frac{L\left(\frac{r}{2},\theta\right)}{2} \tag{5.6}$$

$$H_0(r,\theta) = H\left(\frac{r}{2},\theta\right) \tag{5.7}$$

In fig. 5.11 the coefficient space for each level and each subband is plotted for an example image. Note that these images are transformed back into the spatial domain using inverse fast Fourier transform hence the preservation of spatial layout.

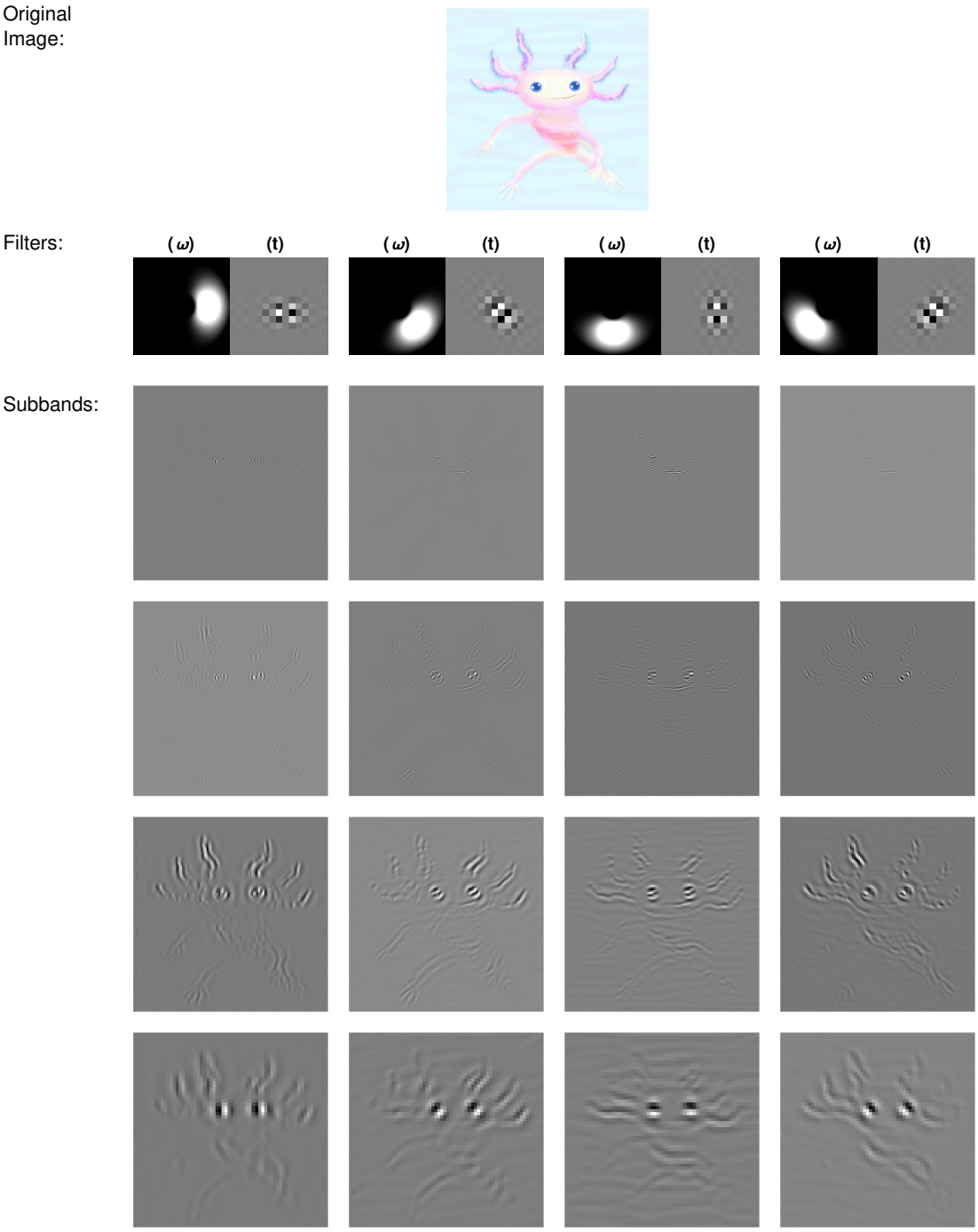


Figure 5.11: Complex wavelet steerable pyramid decomposition with four subbands and four orientations. The corresponding filters are shown in polar frequency and the real part is shown in the spatial domain. The response at different subband levels demonstrates that the pyramid is sensitive to edges at an increasingly coarse scale.

5.2.5 View matching metrics

The view matching procedures used in this work can be classified as being either intensity based (normalised root mean squared error (NRMSE)) or structural based (SSIM and CWSSIM).

5.2.5.1 Normalised root mean squared error

For two images I and I_{ref}, with dimensions n x m, the mean squared error is:

$$MSE = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\left(I(i,j)^2 - I_{ref}(i,j)^2\right)}{m \cdot n} \tag{5.8}$$

In order to compare this with other metrics, the MSE is rooted, normalised in the interval [0 1] and inverted (for convenient comparison with correlation methods) to form the NRMSE.

$$NRMSE = 1 - \left(\frac{\sqrt{MSE}}{255}\right) \tag{5.9}$$

5.2.5.2 Structural similarity

The original structural similarity formulation includes three terms; luminance, contrast and structure (Wang et al., 2004). For performance reasons the luminance and contrast terms were dropped. The remaining structure term correlates local spatial neighbourhoods of each pixel from two images **x** and **y** as outlined in eq. (5.10). The constant term, C, is an arbitrary low value number included to prevent a divide by zero occurrence.

$$SSIM(\mathbf{x}, \mathbf{y}) = corr2(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{xy} + C}{\sigma x \sigma y + C} \tag{5.10}$$

Unless otherwise specified a 7x7 kernel was used for this operation. As specified in the original implementation, an isotropic Gaussian filter with $\sigma = 1.5$ is employed to generate a weighting function centered on the reference pixel. This helps to prevent blocking artefacts.

5.2.5.3 Structured similarity applied to the complex wavelet steerable pyramid

The SSIM algorithm has been previously used to compare complex wavelet steerable pyramids decompositions by (Sampat et al., 2009). Here the formulation for this

approach is adapted and combined with the perfect memory principle outlined in fig. 5.6. The approach is to compute the normalised correlation of coefficients for each subband according to eq. (5.11) at the level(s) of interest. The mean of the set of subband comparisons is then computed (eq. (5.12)). For efficiency, performance and biological plausibility reasons, only the lowest resolution level is included in the metric score for all of the results in section 5.3. The image comparison is conducted on the coarsest scale of a two-orientation, five-level pyramid.

In order to achieve the performance gains necessary for real-time comparison of hundreds of image decompositions per tenth of a second, two key optimisations were made. Firstly, filter masks were precomputed and stored in memory at initialisation time for the filters specified in eqs. (5.2–7) rather than these being recursively generated per comparison. Secondly, a parallel CPU work pool was generated for the processing and comparison of images. A pool size of two was sufficient for real-time operation in this study but in general the procedure is amenable to a greater level of parallelisation. The algorithm is coded in a vectorised fashion with the python *Numpy* package and it is encapsulated with a ROS node for convenient interface with the architecture described in Chapter 3 (fig. 3.11).

$$\tilde{S}(c_x, c_y) = \frac{2\left|\sum_{i=1}^{N} c_{x,i}, c_{y,i}^{*}\right|}{\left|\sum_{i=1}^{N} c_{x,i}\right|^2 + \left|\sum_{i=1}^{N} c_{y,i}\right|^2} \tag{5.11}$$

where * denotes the complex conjugate, N the number of coefficients per subband and $c_{x,i}$ and $c_{y,i}$ are the ith coefficient in subbands of two decomposed images, x and y respectively.

$$CWSSIM = \frac{\sum_{i=1}^{K} \tilde{S}(c_x^i, c_y^i)}{K} \tag{5.12}$$

where K is the number of orientated subbands per image and c_x^i/c_y^i are the set of coefficients in the ith subband for the decomposed images, x and y respectively.

5.2.6 Homing subsystem

A 2D (horizontal) scatter plot of an example trajectory generated by the TORF homing procedure is included in fig. 5.12. Following a direct outbound route, the target heading of the biorobot is rotated by 180°. During this period of rotation, the biorobot is allowed to drift for a few seconds before the homing procedure starts. Thus providing some

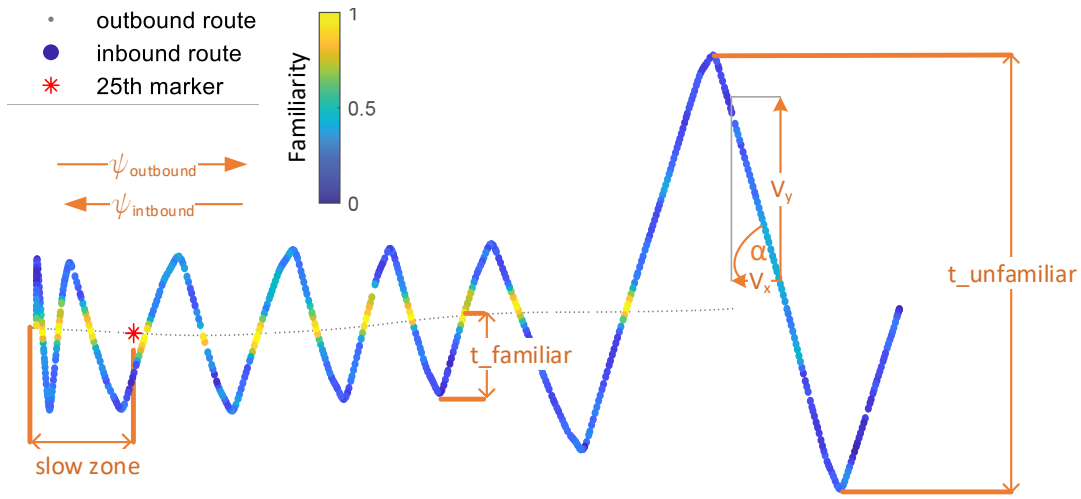


Figure 5.12: Schematic of the homing procedure (top down view). The inbound route is coloured according to an arbitrary familiarity metric. The key parameters that influence the trajectory's shape are labelled in orange.

variance in the starting conditions in a manner that should guarantee that the homing system will always cross the outbound route within a couple of search oscillations.

The biorobot is velocity controlled in the horizontal (XY) plane. The control law directs a longitudinal (V_x) and a transverse (V_y) component of velocity which creates an angle of attack (α) with respect to the biorobot's heading. The procedure is summarised in algorithm 1. Here 70° was used for α unless otherwise stated. The oscillatory motion of the trajectory is produced by inverting the sign of V_y according to one of two events. The period of coherent motion between contiguous switchbacks is referred to as a *sidesweep*. The default means of triggering the next sidesweep is a timer which is reset at the beginning of each switchback. The duration of this timer is referred to as $t\_unfamiliar$ and in this work it is set at 10s, except on the first sidesweep where it is set to 5s so that the biorobot's search is centered roughly on the line of the outbound route. This is a simple means of simulating an effective searching pattern. The on-route means of triggering a switchback is related to the familiarity signal itself. The current view is periodically assessed for familiarity, if at any stage during a sidesweep a view exceeds the familiarity threshold a second timer, t_familiar, is initialised. If this timer exceeds 1 second then a switchback event is triggered. If t_familiar is active and a view that is more familiar than has been previously encountered on this sidesweep, t_familiar is reset so that a full 1 second is again required to elapse before the next switchback is triggered.

```
1  Function TorfHoming(V_y, stored_images):
2      NEW_SIDESWEEP = True
3      AT_HOME = False
4      while not AT_HOME do
5          if is NEW_SIDESWEEP then
6              V_y = −V_y
7              start T1 /* Start catchall timer                    */
8              stop T2 /* Stop/clear time since familiarity        */
9              THRESHOLD_EXCEEDED ← False
10             BEST_SCORE ← 0
11             BEST_SCORE_INDEX ← 100
12             NEW_SIDESWEEP ← False
13         end
14         if len(image_queue > 0) then
15             this_image ← image_queue.pop()
16             [this_score, this_index] ← CWSSIM(this_image, stored_images)
17             if this_score > 0.82 then
18                 if THRESHOLD_EXCEEDED == False then
19                     start T2
20                     THRESHOLD_EXCEEDED ← True
21                 end
22                 if this_score > BEST_SCORE then
23                     BEST_SCORE ← this_score
24                     BEST_SCORE_INDEX ←this_index
25                     reset T2
26                 end
27                 if T1 > t_unfamiliar then
28                     NEW_SIDESWEEP ← True
29                     if BEST_SCORE_INDEX < 5 then
30                         AT_HOME ← True
31                     end
32                 end
33             end
34             if T1 > t_unfamiliar then
35                 NEW_SIDESWEEP ← True
36             end
37         end
38     end
```

Algorithm 1: Torf homing algorithm. Pseudcode for the logic which is used to generate setpoints for the transverse velocity (V_y) during TORF homing behaviour.

In addition to the best score of each sidesweep, the index of the training image that this corresponds to is also logged. This value can be subsequently used as a quasimetric that describes the distance to the nest. When this index falls below a threshold value (25 works well) V_x is proportionally reduced causing α to increase. If the matched image index falls below 6, the final sidesweep is initiated. On this final sidesweep, the behavioural state machine is terminated once the familiarity threshold is exceeded or a secondary timer elapses. While no further homing effort was implemented here, it is assumed that the insect would be in visual contact with the location of interest at this stage causing a separate short range visual homing behaviour to initialise (e.g. (Stürzl et al., 2016)).

5.2.7 Experimental procedures

5.2.7.1 Localisation plausibility assessment

This procedure outlines how the flight data for the analysis in section 5.3.2 was collected.

1. The MAV is directed away from the starting location at $1ms^{-1}$ and images are sampled at 10Hz. Each image is indexed in chronological order and tagged with its location[1].

2. After 30s, the MAV is rotated 180° and directed back towards the starting location. However, on the return journey, there is also a component of velocity directed transversely to the outbound route. After an initial 5s of motion, the sign of the lateral component is flipped, and thereafter is flipped every 10s thus forming a triangle waveform. The default angle of attack of this waveform is 70°. Note that the heading of the MAV remains at a constant 180° offset to the outbound route, rather than facing the direction of travel. The magnitude of the velocity on the inbound route is equal to the outbound route.

3. (Offline). For each familiarity metric of interest, each permutation of inbound and outbound images are evaluated for familiarity and stored in a results matrix.

4. For each inbound location, the familiarity of each metric is taken as the maximum score against all outbound locations and vice versa.

[1] using the local_position topic published by the EKF2 module in the PX4 firmware

5.2.7.2 Translational and rotational image dataset analysis

Three timestamped motion image sequences were extracted from various flight test procedures. Each sequence was captured with the onboard stabilised camera which was aligned to gravity during flight above a level surface. The first two sequences involved the MAV moving forwards and sideways respectively, both at $1ms^{-1}$ at a height of 5m. The third sequence was captured while the MAV yawed on the spot at $90°s^{-1}$, also at a height of 5m. Datasets with the same specification were generated using the Seville virtual world. These were useful for comparison with existing studies. A ground-level yaw rotation dataset was also created for comparison with other terrestrial work. This is the same as the yaw rotating dataset but the height was set to 0.1m and the camera was directed at the horizon rather than at the ground (fig. 5.13).

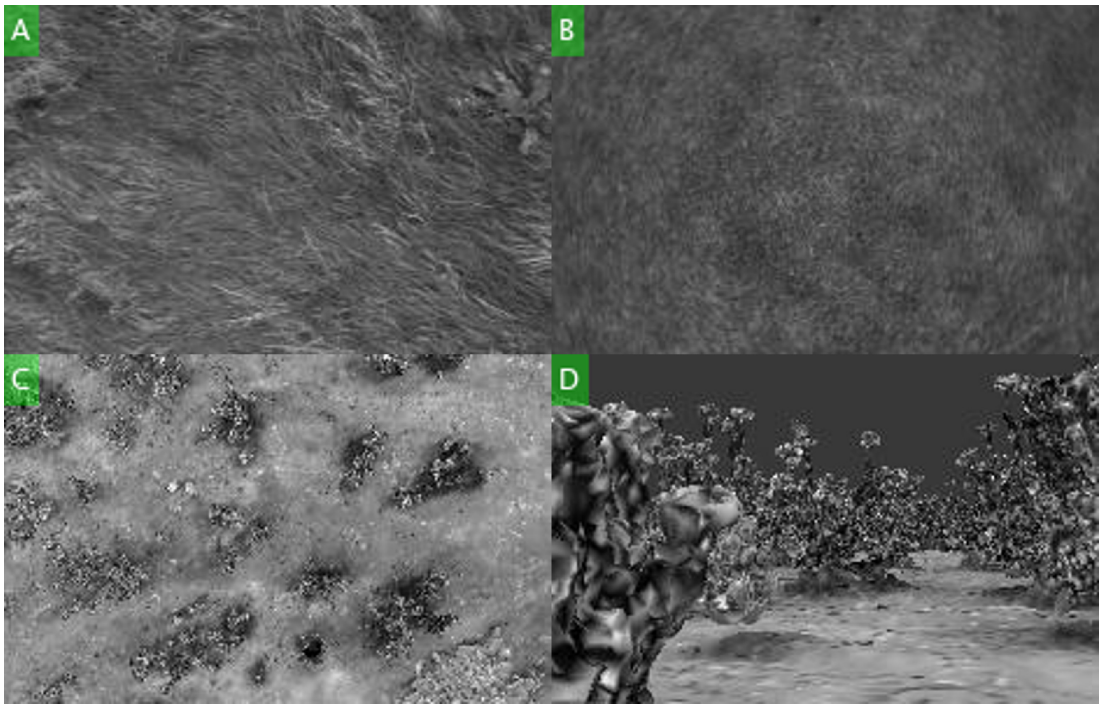


Figure 5.13: Sample images from the test datasets. All are shown at the highest evaluated resolution (235x150). A) Real, translational B) Real, rotational C) Simulated rotational, aerial, looking down D) Simulated rotational, ground-level, facing the horizon.

In order to process the image difference function for a particular dataset, the chronologically (and therefore spatially) central image of each dataset was extracted. All other images in the dataset were evaluated using the image comparison metric under test. The translational or rotational image difference function describes how the metric score varies with displacement from the reference image.

5.2.8 Metrics

5.2.8.1 Catchment area

In a navigational context, it is desirable for the image difference function to return a high-similarity score when the imaging device's location is closely aligned to the image(s) under comparison. This score should monotonically decrease as the imaging device translates and/or rotates away from this. Note that the monotonic region may only be useful while the score exceeds the maximum score outside of this region. Here, the region that satisfies these constraints is referred to as the *catchment area*. It is also desirable for the catchment area to occupy a large span of the metric score (y-axis in fig. 5.14). Another desirable property is that there is a large familiarity score difference between familiar and unfamiliar areas. The distance between the highest scoring unfamiliar location and the lowest scoring familiar location can be thought of as a signal-to-noise ratio.

5.2.8.2 Recognition rate and familiarity count

The recognition rate (RR) is a metric that indicates how reliably a familiarity metric will detect intersecting points of overlapping paths whereas the familiarity count (FC) is a metric that indicates the total catchment area of all of these events. Both metrics rely on determining a threshold value for discriminating between familiar and unfamiliar locations. These metrics are therefore not absolute scores, rather they are methods for comparing the relative usefulness of view matching approaches applied to any given dataset. The steps for calculating RR and FC are illustrated in fig. 5.14 and are outlined below (instruction list labels correspond to fig. 5.14 subplot labels):

a The quantity of inbound/outbound route crossover points are identified (instances where the inbound route bisects the outbound route). The use of XY-plane trajectory scatter plots (e.g. fig. 5.19) can be used for this. For each crossover point, the pairing of inbound/outbound locations with the smallest Euclidean inbound→outbound distance is identified by minimising a pairwise distance between observations (e.g. using Matlab's pdist function) (fig. 5.14a).

b The window of 10 points[2] before and after the set of crossover points is classified as a *familiar region*. Note, due to GNSS/stabilisation errors the peak may not

[2] 10 is suitable for the test parameters here but not necessarily applicable to all experimental setups

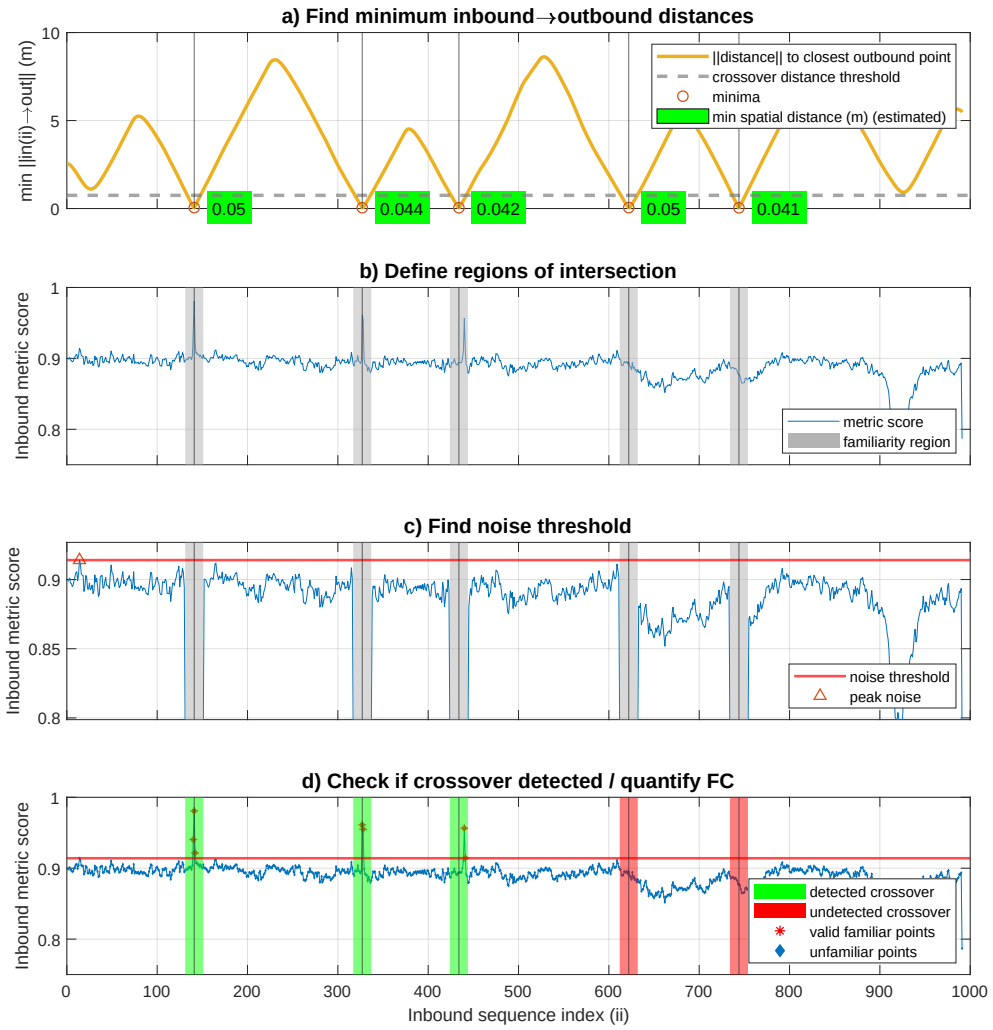


Figure 5.14: Determining FC and RR. Legend entries are consistent for all subplots.

coincide exactly with the closest crossover point (see the central detection spike in fig. 5.14b). In such cases, the reference point is manually shifted to the median peak familiarity value.

c The *noise threshold* is set at the score of the highest metric value outside of the set of familiar regions for the route. Note that all points above this threshold should exist on a monotonic manifold. If there are any local maxima points then the noise threshold value is shifted upwards until the monotonic manifold condition is met.

d The RR is the percentage of detected crossover points (see green and red patches

in fig. 5.14b for detected/undetected locations respectively). The FC is the total quantity of points in the series that exceed the noise threshold value.

For the example in fig. 5.14, the RR is 60% and the FC is 7.

5.2.8.3 Median outlier detection

A median outlier detection (MOD) method was used as a secondary metric for evaluating crossover detection rate and catchment area quantification. This alternative approach was used to ensure that a given method under review isn't over-penalised due to a single bad result as could be the case with the method in section 5.2.8.2. The Matlab® *isoutlier* function was employed to identify any outlier of at least three scaled median absolute deviations away from the median value. The upper outlier threshold is used to identify high familiarity outliers. The crossover points identified in section 5.2.8.2 were then used to categorise outliers. Any high familiarity outlier that is within 10 datapoints of a crossover point is considered a true positive, otherwise it is considered a false negative. Data processed in this way are plotted in fig. 5.22.

5.2.8.4 Homing error

The homing error is the Euclidean distance in the XY plane between the start location of a trial and the position of the biorobot after the homing procedure is completed.

5.2.9 Test locations

A description of the primary test locations is included in fig. 5.15.

Figure 5.15: Test locations. **Left)** Photograph of Dryden Farm. This flat arable land site was where all real flights were performed. Any vertical obstacles are distal and at the horizon. **Right)** Rendering of the Seville model. This virtual world was built with laser scans and features scrub land that is populated with low lying vegetation.

5.3 Results

5.3.1 Comparison of image similarity metrics on motion image sequences

The work in this subsection is largely superseded by work in the remainder of the results section but is included for comparison with other visual navigation publications. The three image comparison metrics that are outlined in section 5.2.5 are applied to the procedure outlined in section 5.2.7.2 for the real and simulated motion image sequences and the results are presented in fig. 5.16 and fig. 5.17 respectively. These figures demonstrate that each view matching procedure generates a catchment area that could be exploited by a prospective navigational algorithm operating in the confines of each spatially limited dataset. In agreement with (Wystrach et al., 2016a), lower resolution views produce the desirable property of broader catchment areas and therefore an extension to the effective range of each reference snapshot. The limits of this trend are exhaustively explored in section 5.3.3.4.

In fig. 5.17, the NRMSE approach appears to produce wider catchment areas than the CWSSIM. This trend is not reflected in subsequent results with multiple crossover points. This discrepancy highlights a deficiency with using a reference image from a video sequence with an automatically exposing camera: neighbouring images are correlated in intensity due to exposure settings as well as scene content. In order to demonstrate this, the procedure in section 5.2.7.2 was repeated with a dataset in which

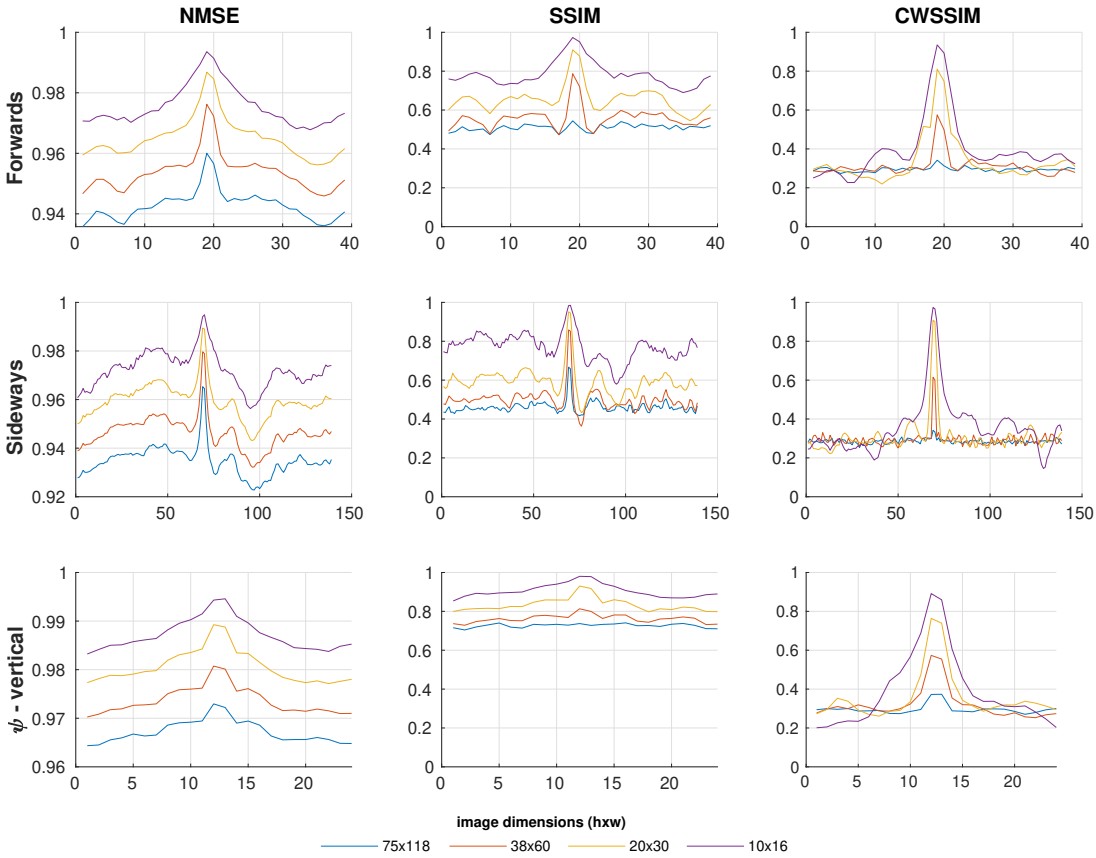


Figure 5.16: Image difference metrics for the real-world datasets as specified by the label on the y axis of each row. The x axis corresponds to the index of each image in the sequence and the y axis is the metric score. Image dimensions correspond to the actual size of the inputted images for MSE and SSIM or the dimensions of the level used for comparison in the CWSSIM image pyramid.

the reference image was captured 30 seconds after the training image set was acquired. The results of this procedure are included in fig. 5.17. In this case, the NRMSE is not able to identify the crossover point at all and the slight peak in familiarity experienced at the crossover point has a narrower catchment area than the CWSSIM. The NRMSE score is correlated with the global image statistics which, for our aerial datasets, vary smoothly over time (see the top panel in fig. 5.17). By contrast CWSSIM is not correlated to the global image statistics and is therefore more robust when used in conjunction with real world vision systems.

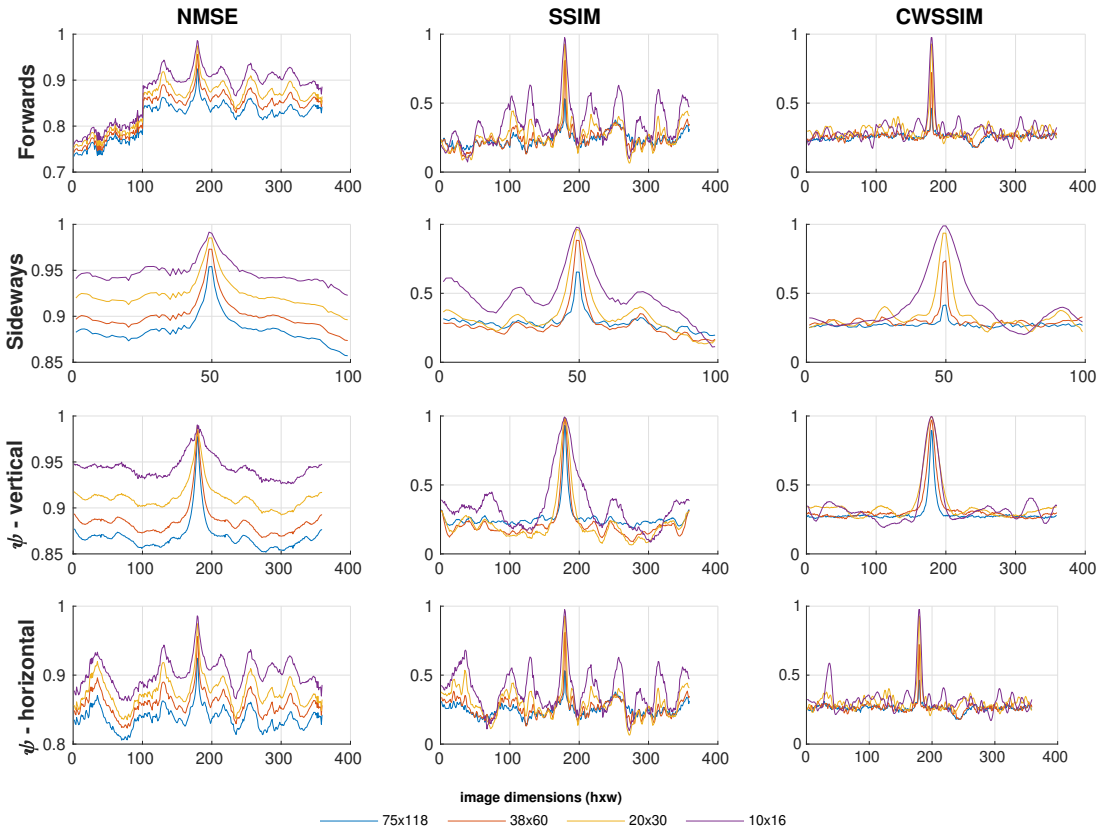


Figure 5.17: As per the caption for fig. 5.16 but this figure contains the simulated datasets described in section 5.2.7.2.

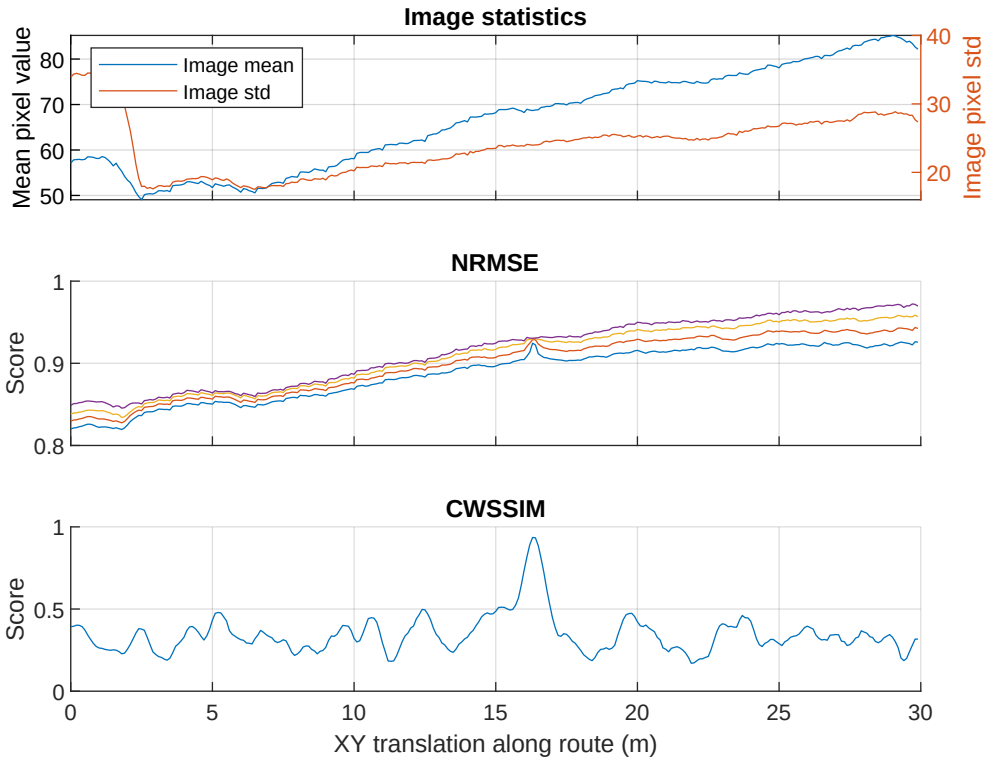


Figure 5.18: CWSSIM is not correlated with image statistics. **Top panel**: Image statistics for the training flight. The reference image is at 17m along the route. It has a mean and standard deviation of 86.1 and 30.1 respectively. **Middle panel** and **Lower panel**: NRMSE and CWSSIM metric responses respectively. As per fig. 5.16 but with a dataset in which the reference image was captured separately (in time) to the test images.

5.3.2 Selection of view comparison metric for routes with multiple crossover points

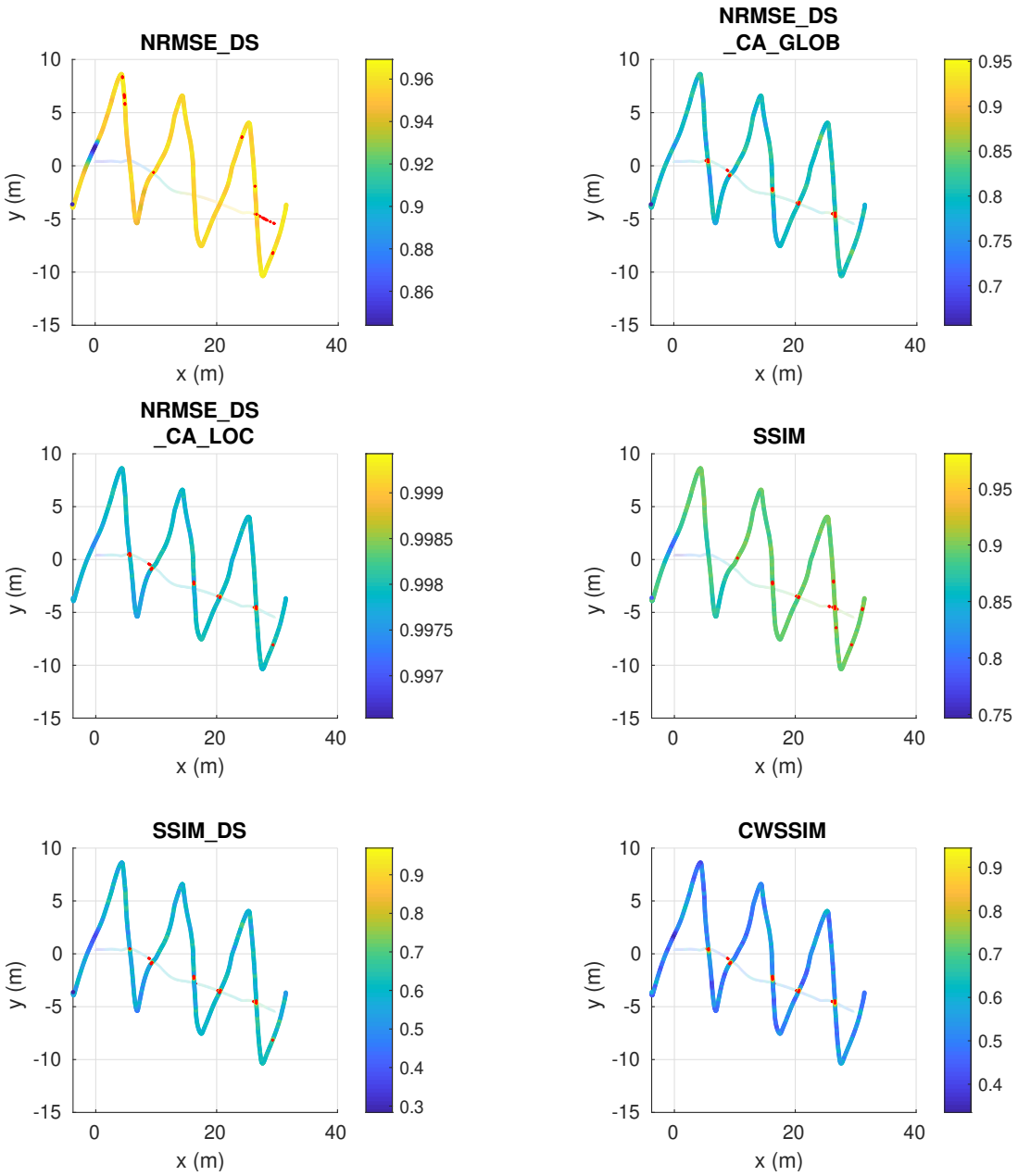


Figure 5.19: Metric score for each location in a sidesweep mission performed at Dryden farm. The outbound route is the straight path with small, translucent markers. The inbound route is the zig-zagging formation with large filled markers. The angle of attack is set at 70°. Each trajectory datum is colour coded according to the corresponding familiarity score under test. Red markers indicate the top 15 scores for each metric on each journey segment

In order to establish whether the image comparison metrics can discriminate between familiar and unfamiliar views across the time course of an entire homing flight (featuring multiple familiar locations), the procedure outlined in section 5.2.7.1 was conducted at Dryden farm. This visually challenging location comprises a relatively flat field of recently cut grass and is therefore characterised as having an apparently self-similar texture and lack of distinct markers or depth variation. The six image comparison pipelines outlined in fig. 5.7 were applied to the resultant dataset offline.

The trajectory of the test procedure output (hereby referred to as T1) is included in fig. 5.19. Here, T1 is shown for each of the view similarity metrics under test. Each trajectory datum is coloured according to the familiarity score generated by the metric under test according to the perfect memory procedure. It is apparent that for some metrics the familiarity score is more correlated with previously experienced views (crossover points) than others. This is emphasised in the figure by the red markers that denote the locations of the top 15 highest familiarity scores for both inbound and outbound routes. For better performing view metrics, these red markers exist exclusively in the vicinity of crossover points and for worse view metrics they are scattered throughout the entire trajectory.

A histogram of the estimated values for MAV height and yaw error for T1 is included in fig. 5.20. This demonstrates that the view metrics evaluated in this section are tolerant to height shifts in the region of +/-20cm (+/-4% of the 5m, the flight height) and +/-2° in yaw[3].

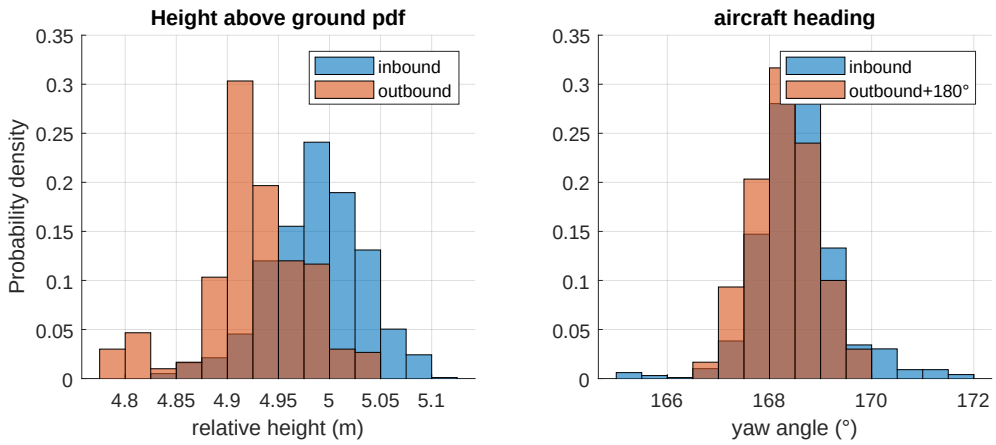


Figure 5.20: Height and yaw distributions for the trajectory illustrated in fig. 5.19

[3]Note that this data is recorded from imperfect sensors. The worst case height error for the height sensor (lidarlite v3) is +/-10cm, no tolerance value can be provided for the heading error which is a fusion of two magnetometers (HMC5983 and LIS3MDL) and three triple axis gyroscopic sensors (LSM303D, L3GD20 and MPU9250)

In order to analyse the relative efficacy of the view metrics for T1, the metric time series data was analysed using the FC, RR and MOD metrics and the time series data is plotted in fig. 5.21.

The RMSE_DS noise level for T1 is at such a level that the metric is completely unable to discriminate between familiar and unfamiliar locations (RR = 0%). The RMSE_DS metric operates directly on absolute pixel intensity values which prove to be a poor basis for real-world image matching schemes due to natural lighting variation and camera exposure settings. For T1, the lighting conditions remain relatively constant but the camera exposure settings are automatically adjusted for each frame resulting in inconsistent encoding of real-world intensities. Note that due to the low dynamic range of modern camera sensors, fixing the exposure settings of a camera would result in highly saturated images. By introducing contrast enhancement into the preprocessing pipeline, RMSE_DS_CA_GLOB is able to achieve the best performance seen by any of the intensity based methods. All five crossover locations are identified (RR=100%) and an FC of 15 indicates that there is some margin for error in terms of route crossing detection[4]. Since the global contrast enhancement does not have a likely neural implementation, a local contrast enhancement method (RMSE_DS_CA_LOC) was also tested and performed well in terms of locating previously experienced views (RR=100%) with only a slight reduction in FC when compared to RMSE_DS_CA_GLOB.

The structural methods (SSIM, SSIM_DS and CWSSIM) were all able to identify familiar locations. SSIM (without resolution downsampling) is included to illustrate the problems with high resolution images for holistic view matching schemes. While it performs well at not delivering false positives, the associated RR is limited to 60% and the corresponding FC is half that of the best scoring intensity methods. The SSIM is therefore a good short range familiarity metric but it is only able to identify familiar locations when the current view is highly coincident with a training view. This is an undesirable characteristic for the proposed homing strategy which depends on the reliable detection of inbound/outbound overlap events. The downsampled variant of the SSIM, (SSIM_DS), overcomes this issue by comparing the less transient, lower frequency image details. With this additional preprocessing step the RR is 100% and the FC is at the same level as the RMSE_DS_CA_LOC. Finally, the CWSSIM metric also achieves a 100% RR and at 32, the corresponding FC is more than double the FC of the next best method. This indicates that the catchment area of each training view

[4]If multiple points are detected per crossover, this implies a degree of redundancy in the image sampling rate but since there are infinite possible crossover points, full redundancy cannot be guaranteed.

is wider for the CWSSIM than for any of the other metrics which equates to reduced search requirements and/or reduced memory capacity requirements for the storage of learning flight images. The respective catchment area sizes can be visualised in the view correspondence matrices in fig. 5.21.

The MOD outlier detection method was also used to test a given view metric's ability to discriminate between noise (unfamiliar residual scoring) and true positive locations. The application of this procedure is plotted in fig. 5.22 and all view matching metric performances are summarised in table 5.1. It is apparent that the rank of each metric is consistent regardless of whether this is governed by MOD or the custom FC and RR metrics.

	Custom metric		**MOD**	
Strategy	**RR**	**FC**	**TP**	**FP**
NRMSE_DS	0	0	0	0
NRMSE_DS_CA_GLOB	5	15	17	1
NRMSE_DS_CA_LOC	5	14	14	2
SSIM	3	7	6	0
SSIM_DS	5	14	14	4
CWSSIM_DS	5	32	27	0

Table 5.1: Summary of view metric RR, FC and MOD scores. A good view comparison strategy exhibits a high RR, FC and TP (true positive), and, a low FP (false positive).

In this section, six view metrics that could be used for the detection of previously experienced views were identified. The intensity based methods (RMSE_DS, RMSE_DS_CA_GLOB and RMSE_DS_CA_LOC) can be implemented as a preprocessing step in the view matching pipeline (see fig. 5.6) with a few lines of code that interface with the OpenCV library. As outlined in table 5.1, CWSSIM is the best performing view matching metric. While the transformation process outlined in section 5.2.4.5 must be added to the view matching pipeline for CWSSIM, this is seen as a worthwhile development step for the reward of a doubled catchment area.

Interestingly, in all metrics that correctly identify crossover locations (all except RMSE_DS), the view correspondence matrix indicates that:

1. There is a high signal-to-noise-ratio between detected views and noisy images

2. All correctly detected views have a low score for other crossover locations (see the relevant columns of the view correspondence matrices in fig. 5.21)

This implies that when a familiar location is detected, the range of this location from the target location can also be inferred. Note that this applies to the perfect memory pipeline but may not be as straightforward with biologically plausible memory schemes.

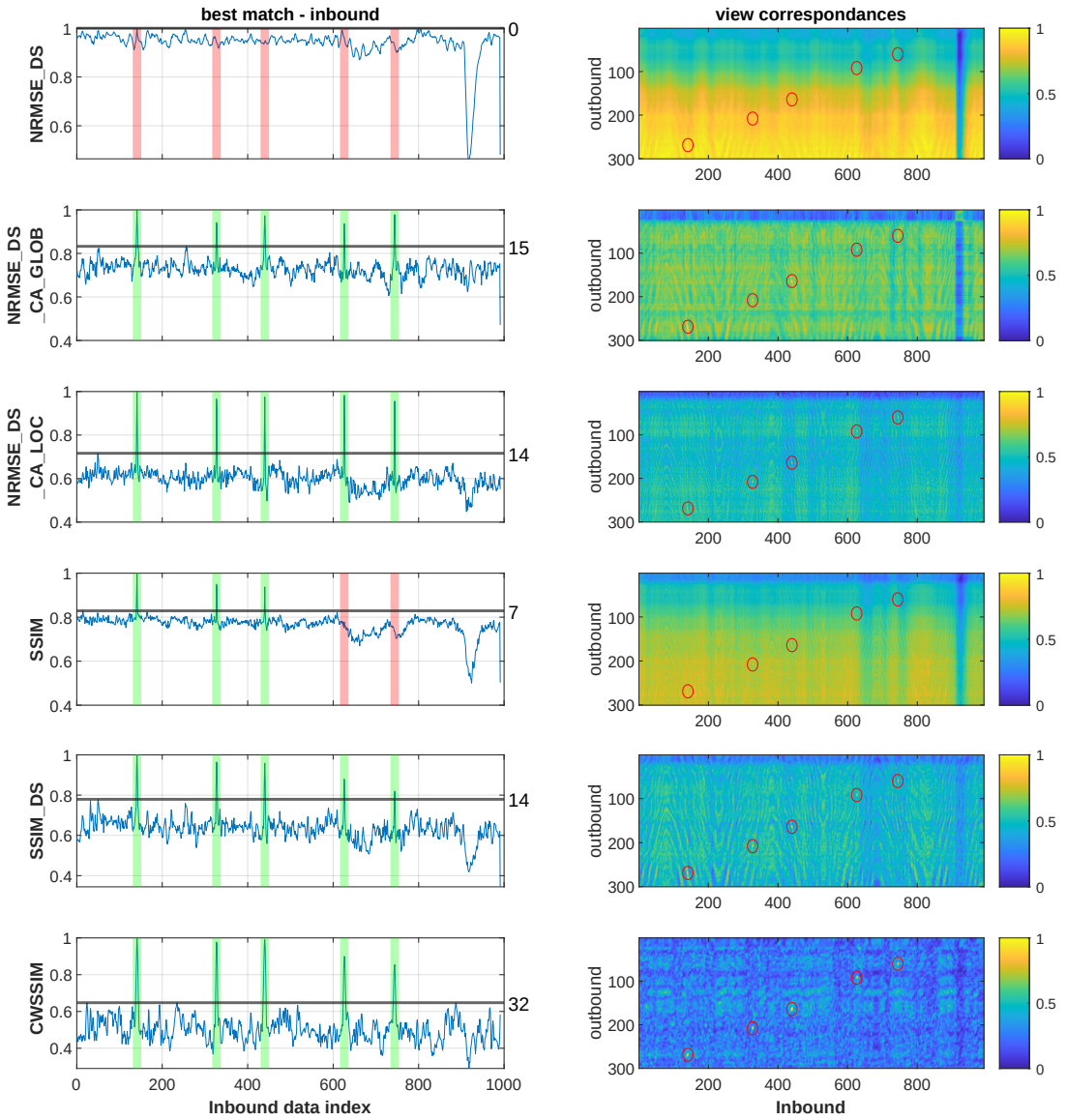


Figure 5.21: Metric matrix. Inbound→outbound view correspondence matrices are shown in the right hand column. The colour relates to the score of a particular view metric (as defined on the left hand y-axis of each row). Red circles indicate the location where spatial crossover points occur. The inbound metric score (maximum of each column of the view correspondence matrix) is plotted as a time series in the figure's left column. This value is the maximum familiarity found in the columns of the associated view correspondence matrix. The horizontal line denotes the maximum noise level and the number to the right of this line is the FC score. Coloured patches indicate the ground truth crossover locations and whether they were detected (green) or not (red).

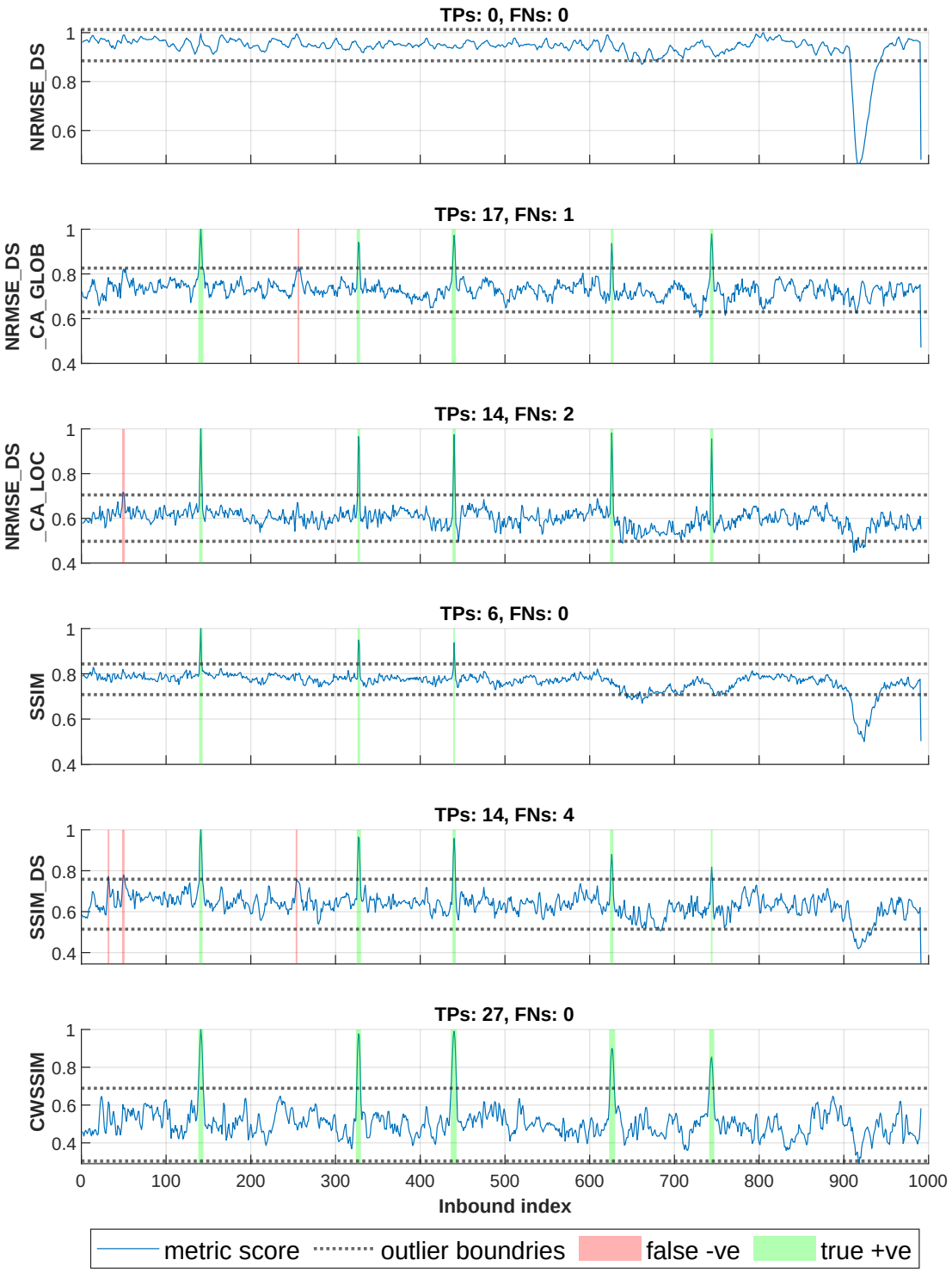


Figure 5.22: MOD applied to the inbound metric score time series for each of the six metrics. True positives and false negatives denoted by colour patches as indicated in the key.

5.3.3 Homing algorithm evaluation

5.3.3.1 Hardware implementation tests

The TORF homing algorithm outlined in section 5.2.6 was evaluated on the biorobot at Dryden farm using the best performing view matching metric, CWSSIM. The procedure converged five times out of five, these routes are included in fig. 5.23. Note that on the day of testing (07/12/2019), there were 16kph wind gusts at the test location. The effect of these gusts can be observed in the outbound routes of subplot **a** and **e** by uncommanded lateral translations. While this is not an in depth examination on environmental robustness, these results go some way towards demonstrating the tolerance of the system to windy conditions. This is is particularly encouraging for the view matching subsystem which could be vulnerable to in scene motion. Robustness to the motion of scenes containing larger vegetation remains untested. Additional trajectories detailing the operation of TORF over some longer outbound routes are included in Appendix A.

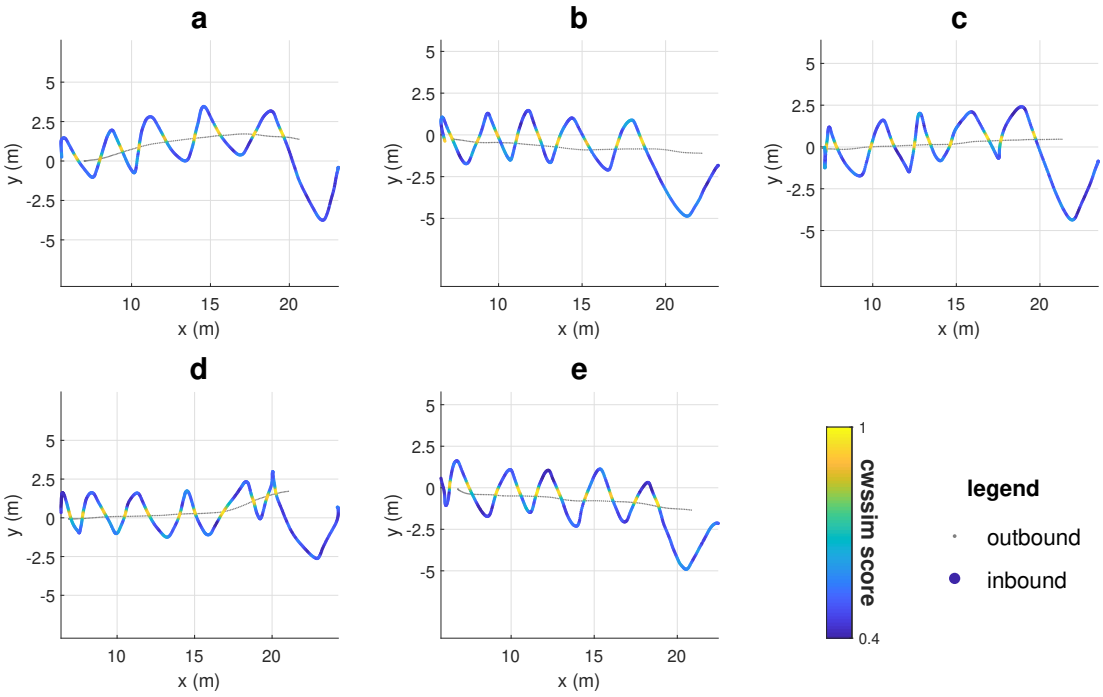


Figure 5.23: Five successful TORF flight trajectories (a-e) performed on the biorobot. The location of each image sample on the outbound route is shown as a small grey dot. The locations of images captured on the inbound route are displayed as a filled circle with the colour indicating the CWSSIM familiarity score at that location.

5.3.3.2 Simulated implementation tests

Performing MAV flights in the physical world is a time consuming process. The simulation environment (outlined in Chapter 3) provided a means of conducting a greater number of trials than would be possible on the physical hardware. The availability of high fidelity natural scene datasets[5] meant that the biorobot could be rapidly tested in multiple relevant landscapes.

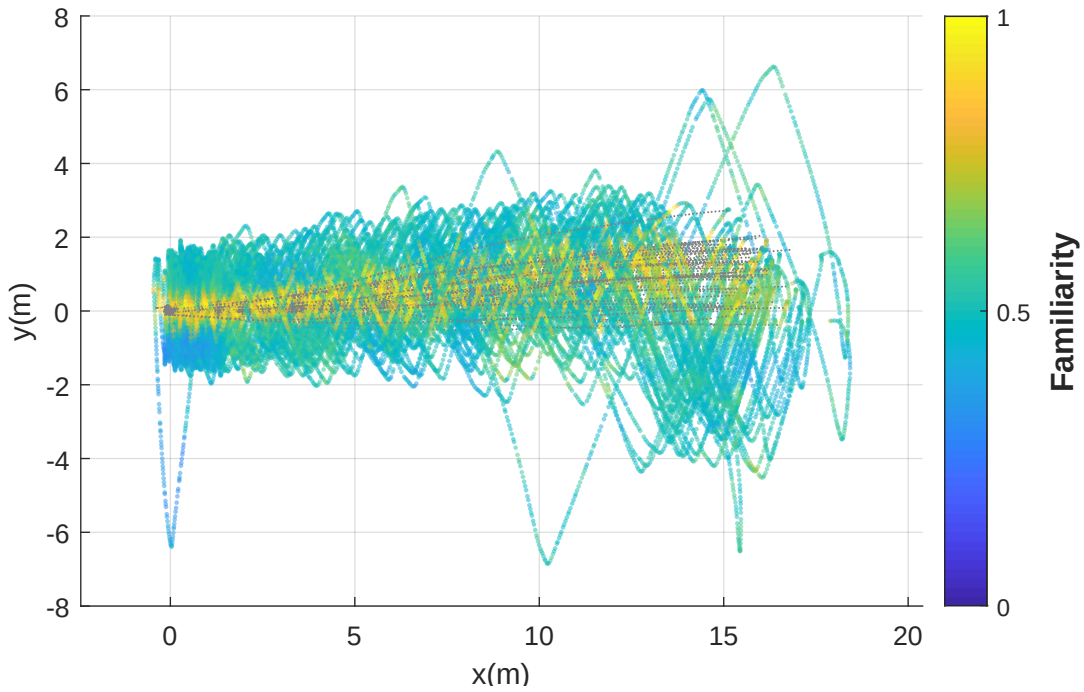


Figure 5.24: The profile of 64 trajectories conducted in the Seville environment.

In order to estimate the worst case performance of TORF with CWSSIM, this homing procedure was simulated 64 times in the Seville environment. The overhead view of each trajectory is included in fig. 5.24, the route variability goes some way to demonstrating that the simulator is a good tool for modelling dynamic behaviours. Note that 100% of trials successfully tracked the ridge of familiarity. This level of robustness was only achieved after a trial and error tuning process which was conducted in the simulation environment.

The final location of each trajectory is included in fig. 5.25 where the worst case final error is 80cm. This demonstrates that the nest search area radius could be reduced from 15m to 0.8m by use of the visual homing algorithm detailed herein. Aggregating inbound and outbound time benefits of this visual navigation behaviour for a given

[5] https://insectvision.dlr.de/3d-reconstruction-tools/habitat3d

foraging trip provides some insight as to why selection pressure would favour foraging animals with visual navigation abilities.

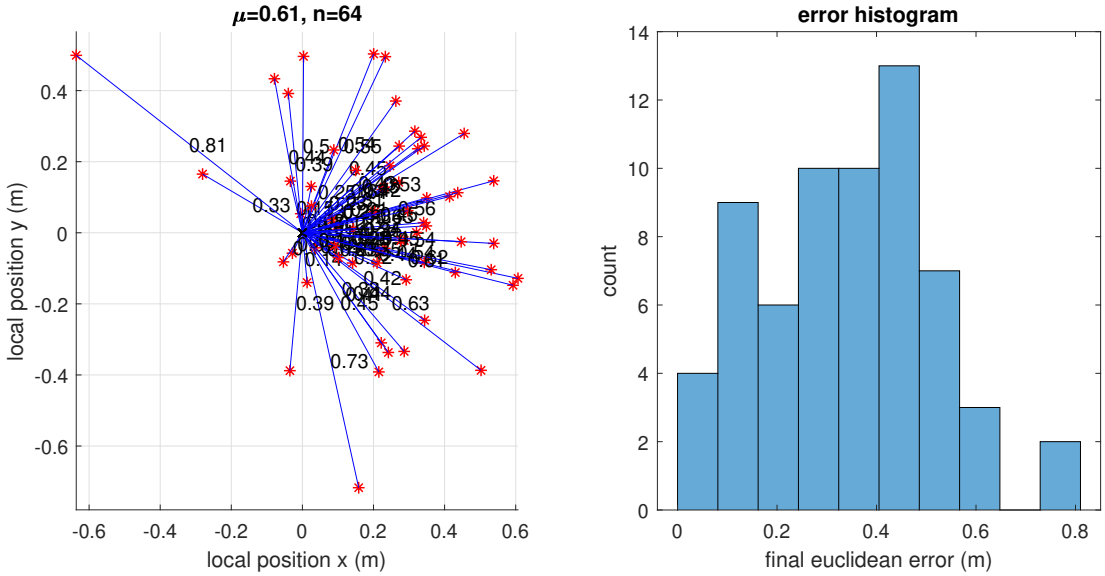


Figure 5.25: Final Euclidean error for 64 trials in the Seville environment. **left** - scatter plot of all landing sites shown as red markers. **right** - histogram of the homing errors of each trial.

The TORF procedure was also simulated in five virtual worlds of differing topographies as summarised in fig. 5.26. These worlds had a range of visual texture and height variation properties. The relative performance of the homing algorithm in each world is outlined in fig. 5.27. TORF performed best in the forest plane, a world that has a flat surface and is highly textured with a varied naturalistic image. TORF performed the next best in Seville, followed by the Earthworks and Forest worlds. This series of worlds has an increasingly complicated depth structure. In the case of the forest world, the height control strategy was adapted so that the biorobot remained at a fixed altitude, rather than tracking the ground height of the terrain. While a flying insect's ability to achieve this is speculative, the robustness of the model to the height variations caused by the tree canopy was seen as an encouraging result. The grass texture world was the only environment in which the biorobot couldn't home. This world comprises a tiled surface of the same patch of grass in a repeating fashion. It is not surprising therefore, that false detections were made due to the aliasing caused by identicle visual surfaces. A process which drastically affected the performance of the homing algorithm (see the bottom row in fig. 5.26).

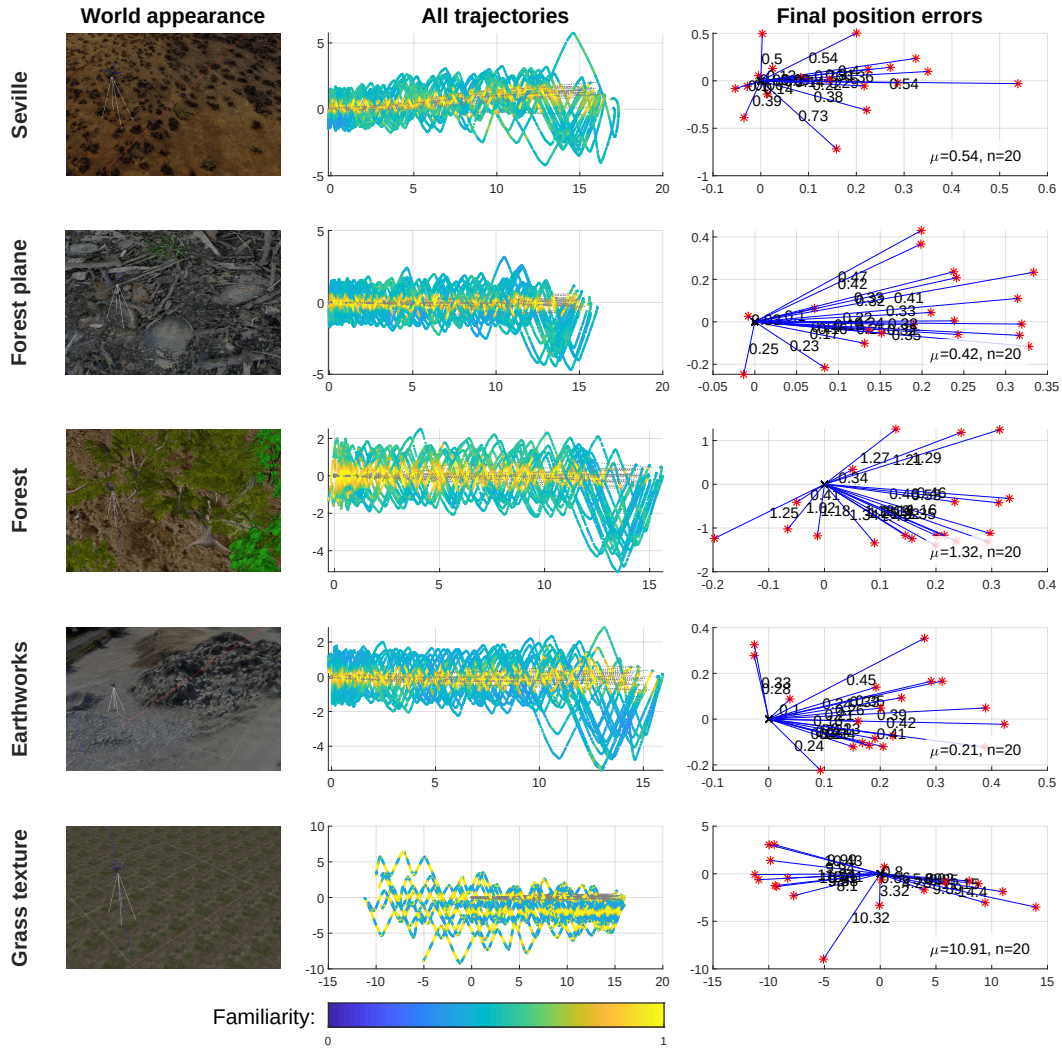


Figure 5.26: Summary of the performance of TORF trialled in different worlds. **Column 1** - screenshots of each test environment. **Column 2** - top down view of 20 trajectories for each route. Each inbound datum is colour coded according to familiarity. **Column 3** - final distance from the nest for each trajectory.

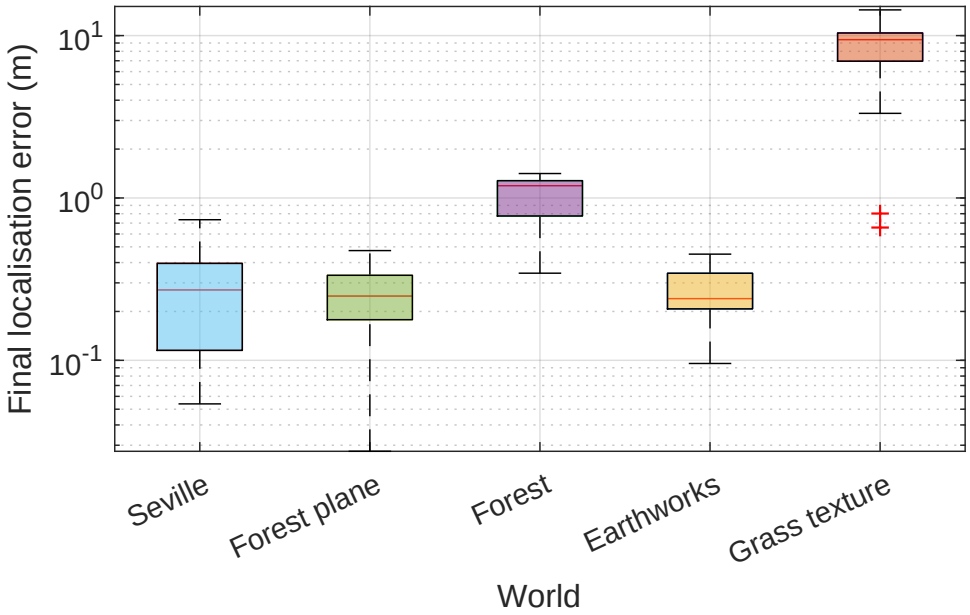


Figure 5.27: TORF homing performance when used with CWSSIM in five different simulation worlds. 10 trials were performed in each world.

5.3.3.3 Immunity to camera pitch angle

In all trials outside of this section, the biorobot's camera is aligned with gravity and directed towards the ground. While this appears to be a good configuration for scene recognition, foraging insect eyes are not believed to have sufficient visual acuity towards the ground to realise this. In order to establish whether TORF can operate successfully with biologically plausible view directions, the pitch angle of the camera was systematically raised in intervals of 10°. The Seville simulation environment was used in this investigation and 10 trials were performed at each pitch angle. Due to the limited size of this simulation environment, the highest pitch elevation trialled was 60°. Note that this represents an area of the eyes of bumble bees and honey bees that has sufficient visual acuity for TORF to operate (see fig. 5.4).

For each trial, the procedure outlined in section 5.2.6 was performed with a minor methodological adaptation. After the outbound route, the biorobot was commanded to return to the start position and the homing procedure was initialised from this location. This meant that the inbound and outbound images were taken from the same heading direction and therefore no mirroring was applied to the outbound snapshots prior to the image comparison phase. Note, this process is likely to be more representative of what a flying insect would do. I.e. it is believed that visual learning would take place under the guidance of path integration on inbound routes. A scatter plot of the results of the camera pitch investigation is included in fig. 5.28. It is apparent that in the Seville simulation environment, TORF with CWSSIM is equally effective across each of the trialled pitch angles.

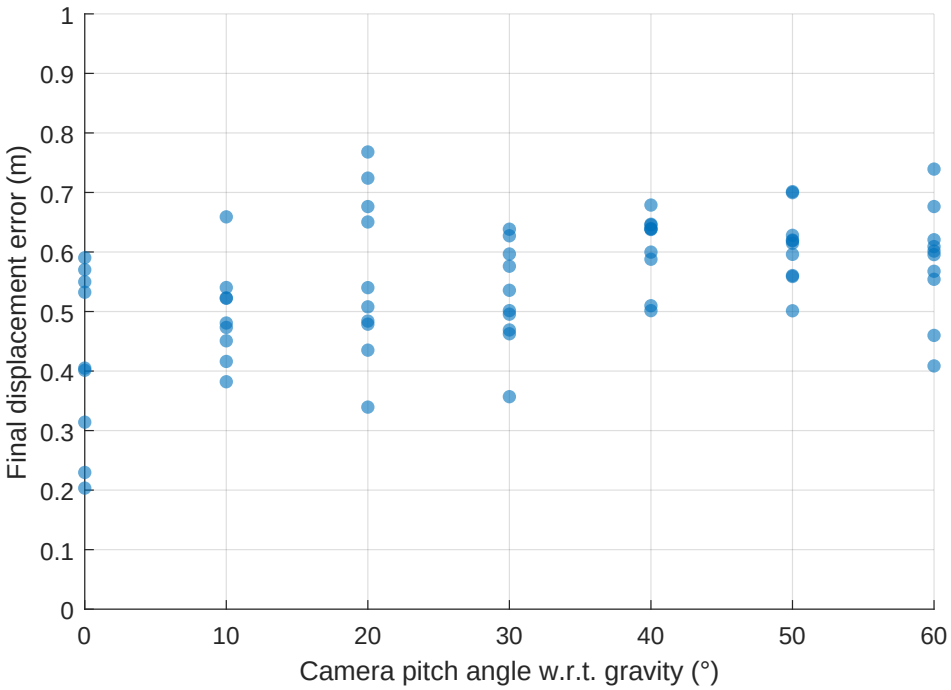


Figure 5.28: Scatter plot of the homing error for a series of simulated trials in which the biorobot's camera pitch angle is systematically raised. 10 trials are performed at each pitch angle.

5.3.3.4 Can CWSSIM operate at biologically relevant resolutions?

In order for CWSSIM to be considered a biologically plausible snapshot mechanism for scene recognition in foraging insects, the metric must be able to discriminate between images captured with the sort of low angular acuities that are typical of compound eyes. This question was first addressed with the real image dataset used in section 5.3.2. These images were resampled to the smallest possible size that could be used with the CWSSIM algorithm, 16x25 and 16x16 for processes that did and did not affect the image aspect-ratio respectively. A scatter plot of the familiarity at each image point is included in fig. 5.29. CWSSIM can identify crossover points in this scenario with zero error (i.e. the RR is 100%) for both sensor sizes. However, as can be seen by the representative portion of time series that is included in fig. 5.29c, the signal-to-noise-ratio is greater for the 16x16 sensor. While the lower resolution images are likely to reduce the system's ability to discriminate between familiar and unfamiliar locations, changing the aspect-ratio of the image series may also contribute to the reduction in performance.

The simulation environment affords more flexibility with the optical properties of

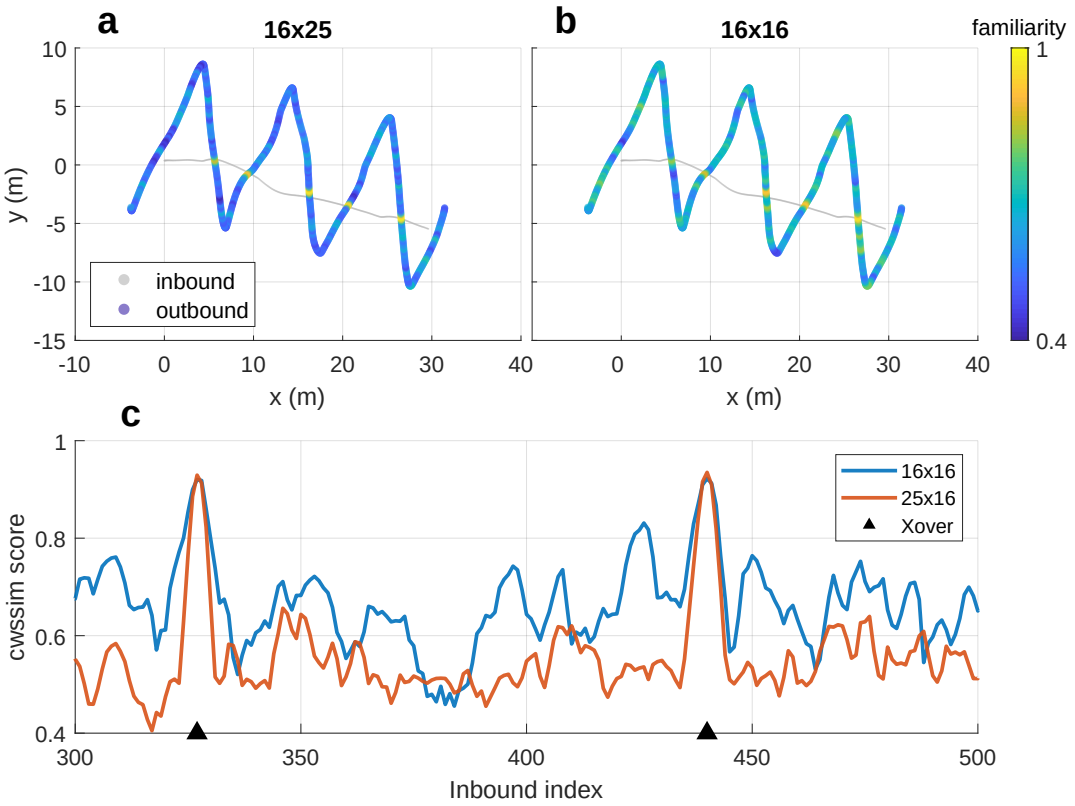


Figure 5.29: Scatter plot of the XY coordinates of the test trajectory. Inbound route is colour coded according to the highest familiarity score when compared to all outbound images. Scatter plot showing the final displacement error of trials performed at the resized images 16x25 (**a**) and 16x16 (**b**). **c** - the time series of the CWSSIM best scores for each inbound image. Actual crossover points are indicated with black triangles.

the camera lens. This includes the ability to systematically modify the angular acuity of the vision system and monitor the effect on homing performance. This was achieved by increasing the field of view of the simulated camera's lens while maintaining the same camera sensor resolution. The results are included in fig. 5.30. It is apparent that as each camera pixel covers a larger field of view, the final displacement error increases. This demonstrates that reducing visual acuity also reduces the precision of the homing procedure when the default parameter settings are used. However, this drop in performance does not effect the overall reliability of the homing procedure. For each trial at all resolutions, the biorobot successfully detected the familiarity ridge and travelled along it until it was close to the trial starting location.

In the previous paragraph a fixed resolution was used as the camera's field-of-view was adjusted. While this test indicated that ultra-low visual acuities can be used in

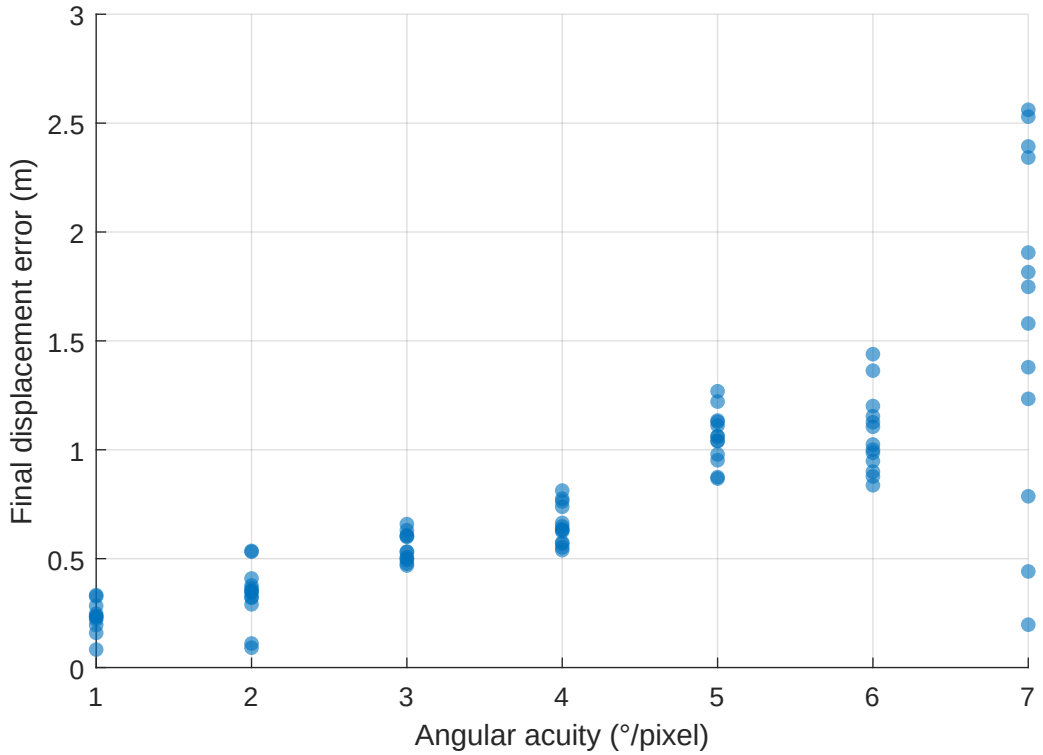


Figure 5.30: Scatter plot showing the final displacement error for trials performed at different angular acuities.

conjunction with TORF, it is not clear what the minimum image size is, and whether this is commensurate with insect vision. To address this, a further batch of experiments was conducted with the camera's field-of-view fixed at 42°x42°. We started with an image resolution of 11x11 pixels and decremented this by 1x1 pixel between trial sets. The results are plotted in fig. 5.31b. False positives are produced by the CWSSIM pipeline when image resolutions below 9x9 pixels are used, this results in the agent failing to home. At 60° below the horizon, the visual acuity is approximately 3° (Taylor et al., 2019). A 9x9 patch in this region would therefore occupy a 27°x27° field of view. We conclude that CWSSIM could be achieved with the resolution and acuity of the fronto-ventral portion of the bee vision system.

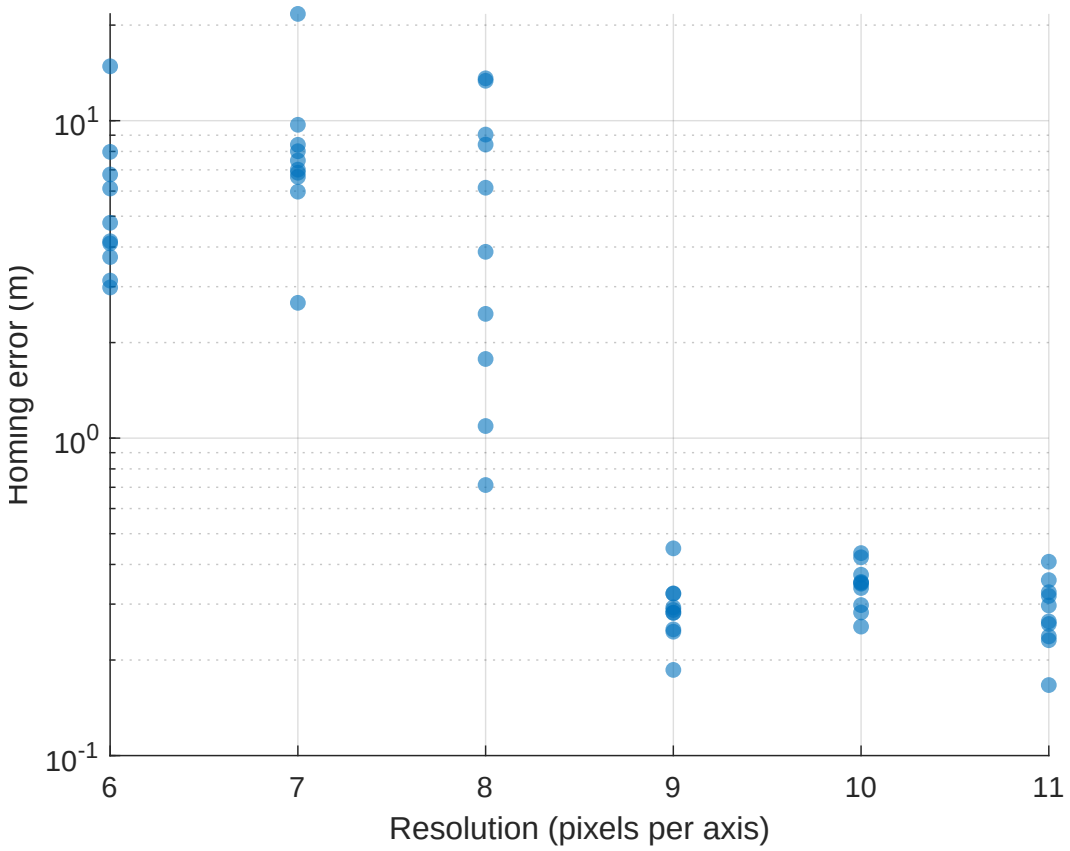


Figure 5.31: Scatter plot showing the homing error for trials performed at different camera resolutions with a constant camera field of view of 42°x42°.

5.3.3.5 Optimal parameters for intensity based comparison metrics

In section section 5.3.2 it was asserted that CWSSIM has a greater translation invariance or catchment area than NRMSE when the number of input coefficients is the same after all preprocessing steps have been applied. In order to verify that the CWSSIM has a larger catchment area than NRMSE with any preprocessing arrangement, a systematic exploration of the effect of blur factor and image resizing was conducted. The search space manifold is plotted in fig. 5.32. The maximal FC score is 22 which is a notable increase in comparison to the score achieved with the CWSSIM equivalent resolution [10x15]. However, this remains substantially lower than the CWSSIM glsfc score (32) which may itself have scope for further improvement. The best image comparison performance was achieved with [6x10] and blur factor of 9. This is analogous to an angular acuity of 7° and an acceptance angle of 21°. It should be noted that this is not necessarily a globally optimal parameter arrangement for scene recognition, but it does give an impression of how very low resolution vision can be applied to the task of scene recognition from an aerial vantage point.

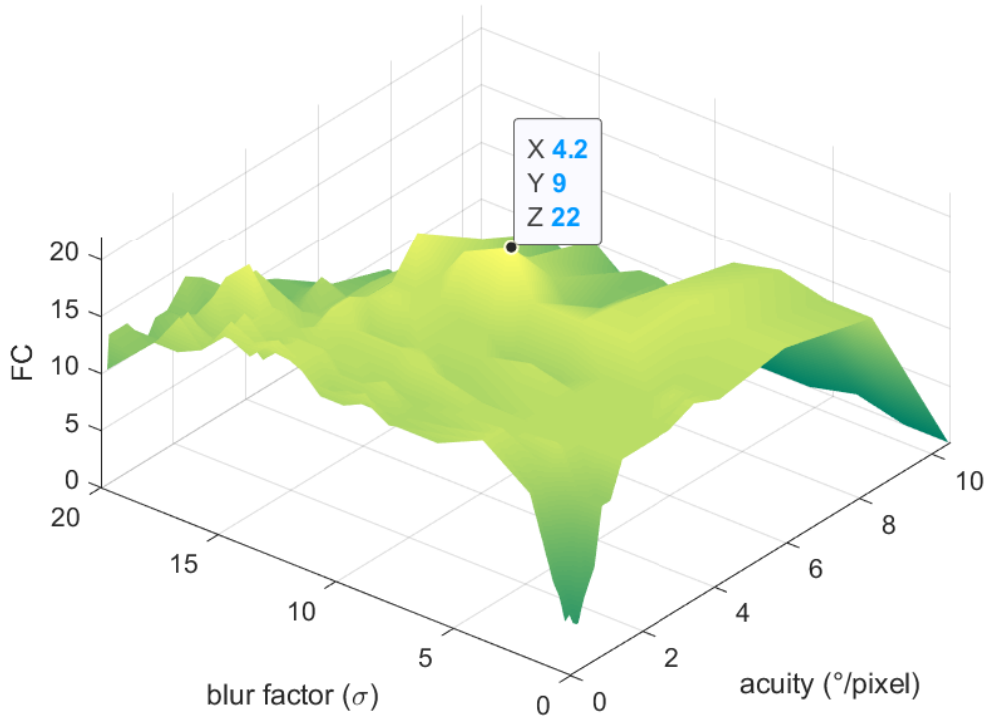


Figure 5.32: FC search space for blur and resized image resolution operations. A datatip is placed at the search space maxima.

5.3.4 Following curved outbound routes

TORF can also be applied to curved outbound routes as could be required in environments with physical obstacles. However, to retain robust performance on outbound curves with tight radii, the angle of attack of the transverse oscillations must be increased as explored in fig. 5.33. Therefore, it is desirable that outbound routes are kept as straight as possible since TORF becomes inefficient on curved routes. It is also possible that either the obstacle avoidance behaviours (such as those enacted by the looming pathways (Muijres et al., 2014)) or a horizontal obstacle range detection ability (e.g. Lingenfelter et al. 2020; Sazbon et al. 2004) would cause the angle of attack to dynamically alter in scenarios where obstacles are approaching. Note that preliminary results indicate that reducing the amplitude of oscillations (set with t_familiar) also improves TORF's route following efficiency. The gradient descent approach outlined in section 5.2.1.1 could provide a more sophisticated control approach by making use of the differentiable characteristic of the familiarity ridge.

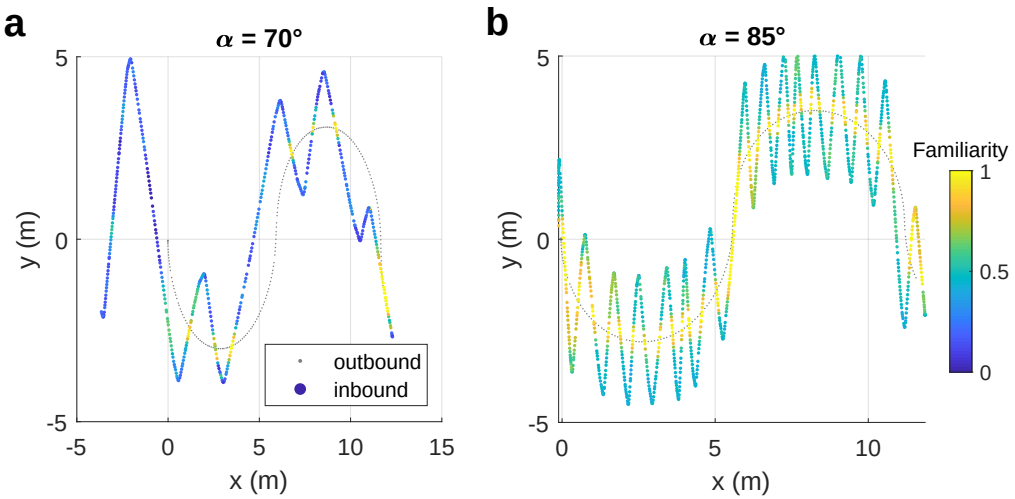


Figure 5.33: Scatter plots of the TORF homing procedure operating on curved outbound routes featuring a tight radii. **a** - using the default transverse angle of attack (α) **b** - using an increased transverse angle of attack.

5.3.5 Effect of height variation on homing performance

In order to quantify the sensitivity of TORF to variation between the height of inbound and outbound routes, a series of simulations was conducted with a fixed constant height

offset between the inbound and outbound routes. Outbound height and CWSSIM familiarity threshold were also found to have an impact on the homing success rate when an inbound height offset was involved and hence these properties were also systematically varied in this investigation. The results are included in fig. 5.34. It is apparent that with the default familiarity threshold, the homing success rate is 100% when the inbound route is within 10% of the outbound route. Reducing the familiarity threshold can increase the homing reliability at greater height offsets but this is at the expense of reliability at smaller height offsets.

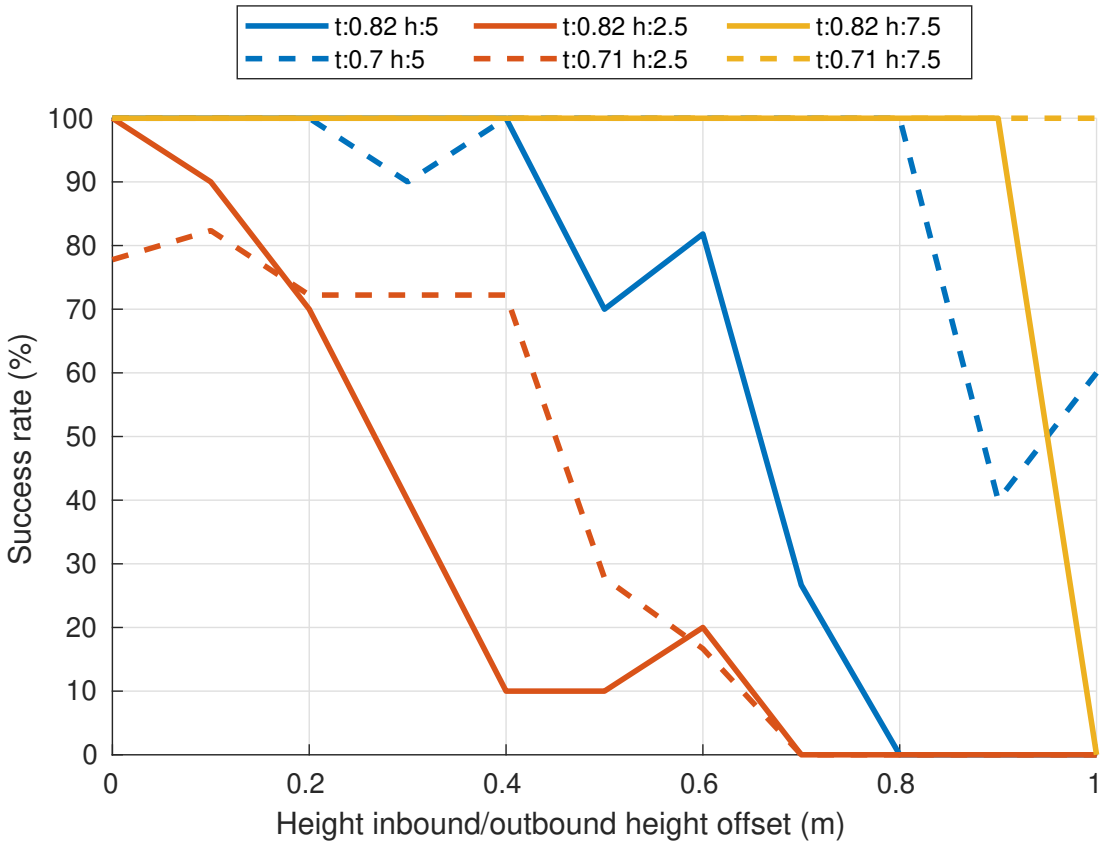


Figure 5.34: Effect of height variation between inbound and outbound routes. For each configuration of CWSSIM threshold (t) and inbound height (h), a fixed offset was made to the height setpoint on the outbound route in increments of +10cm (the x-axis). Each configuration was simulated 10 times at each height offset and the simulation was considered a pass if the biorobot homed to within 1m of the destination.

5.4 Discussion

Since Zeil first demonstrated that translational image difference functions in natural scenes have a bounded monotonic search space (Zeil et al., 2003), this result has been a widely adopted principle for the modelling of terrestrial insect navigation paradigms. For the first time (to the best of the author's knowledge) this philosophy has been successfully deployed on an autonomous MAV forming an aerial analogue of existing terrestrial models. Rather than simply extending state-of-the-art terrestrial navigation paradigms to a three-dimensional problem space, the benefits of an aerial perspective have been identified and incorporated into an evolved visual route following procedure, resulting in surprisingly robust performance that could explain how flying foraging insects are capable of homing in on previously visited sites of interest in a direct fashion.

5.4.1 Discussion on the homing algorithm

The TORF model achieved a 100% success rate in real-world tests and demonstrated the ability to converge in a broad range of simulated locations. This level of robustness exceeded operational expectations. In future an improved implementation and more challenging test schedule will allow the limits of the model to be understood. At the outset, two types of challenging environments were identified; 1) environments which only have vertical features far away in the distance 2) dense cluttered natural environments. In this work TORF has been applied to the former, the lack of an obstacle avoidance system prevented real-world examination of the latter case. It is hoped that the obstacle avoidance functionality will be introduced in future design iterations. Another feature of the main test site was that variations in ground level were gradual. Preliminary trails show that TORF is effective in sites with tall weeds but future work should also probe any challenges introduced by large ground feature height discontinuities. It is hoped that in future longer range three-dimensional trajectories of flying insects will be available such that a better understanding of the height control strategies and capabilities of these insects can be better understood.

Configuring the biorobot to sample a downward view proved a useful strategy, it is difficult to fathom how insects that navigate in the vertically featureless fields that are prevalent in harmonic radar study sites (e.g. (Degen et al., 2015)) could visually localise without preferentially sampling ground-level information. Anatomical studies of bee

eyes demonstrate that 4° acuity is not uncommon at 60° beneath the horizon (Taylor et al., 2019). As demonstrated in section 5.3.3.3, this viewing angle is compatible with the TORF model. Furthermore, behavioural reports indicate that honey bees use distinctive ground based objects, such as rocks to signpost their routes (von Frish K., 1967). However, the strength of the downward facing perspective is also its weakness. Such a view is well localised to a particular place, but has a limited catchment area in the scope of a flying insect's habitat. Therefore, the visual information at this aspect is not effective at providing a long range navigation aid. Conversely, views of the horizon have the potential to be useful over a large range in some habitats, but are often not an effective means for localising on the horizontal XY plane. Behavioural experiments indicate that zero-vector experienced honey bees are able to locate their nest when they are displaced to a novel location (Degen et al., 2015). A downward view would predict that this could only be achieved with extensive searching patterns and this is not observed in harmonic radar data of these experiments. It is therefore hypothesised that rather than a single strategy using holistic views, different visual navigation subfunctions that attend to particular regions of the visual field could exist in tandem. In future studies, the inclusion of a panoramic or wide angled vision system would enable testing of 1) the gradient descent algorithms for comparison with TORF and 2) possible mechanisms for arbitrating between multiple visual cues. It should be noted that while this work focuses on one visual navigation hypothesis, it is entirely possible that multiple strategies co-exist within the scope of a single organism, and/or are distributed across a range of foraging species as dictated by ecological demand.

The implementation of TORF demonstrated that by storing a memory of the direction and an estimated distance towards the familiarity ridge, an agent can follow this visual signal robustly without the need for spatially separated intensity sensors. For this initial implementation of TORF, a relatively large overshoot of the familiarity ridge and a high angle of attack was a required combination for a high confidence in converging. The main driver for this was that the simulated gimbal had a long settling time when its direction was reversed. There is no reason why tighter oscillations would not work in practise and these should be investigated in future studies. The trade-off with this configuration is that more energy is expended on lateral motion than forward progress. An alternative approach would be to minimise path length by inducing fewer, lower amplitude oscillations (specified by switchback interval (t_familiar) and angle of attack (α) in fig. 5.12). One problem with this is that longitudinal familiarity variation could confound the determination of the peak familiarity location. Another issue is that

perturbations due to wind could cause greater problems for strategies with small angles of attack because it would take the agent a longer time to rejoin the path, especially for curved routes. Again, beyond anecdotal reports and low resolution harmonic radar, no quantitative data was found to form a basis for analysis. Interestingly, researchers tracking homing ants have reported subjective oscillations at the micro-scale (Léo et al., July). It is possible that these animals also deliberately pass over the visual corridor in order to keep track of a familiarity ridge.

The reduction of the localisation task space from 6DOF to 2DOF (with some robustness to minor perturbations and offsets in the remaining 4DOF) was key to the success of the homing algorithm. An insect's ability to stabilise its head against the three canonical angular DOFs seems to be well founded (Viollet and Zeil, 2013). However, a flying insect's ability to remain at a constant height above ground is a much greater assumption, at least in free flight. Slow motion video footage of insect flight shows that an insect's instantaneous height control can be very precise (think of a hover fly's ability to level out) but whether the insect's flight control system can remain at an absolute ground height in an arbitrary geographic location is not a given. To probe this and to better understand the role of visual guidance beyond the immediate nest site, would require future studies with the goal of recording three-dimensional extended trajectories of ground nesting wasps.

While not found to be an issue here, the use of a thresholding method to discriminate between familiar and non-familiar locations is rather brittle and there could be scenarios where this is sub-optimal. Another potential source of error that was not fully considered is scene motion. The homing algorithm was tolerant to $16kph$ gusts of wind in the grassy field but it is probable that significant swaying/deformation of larger types of vegetation would cause greater difficulty for the procedure. It is known that flying insects avoid foraging in very windy conditions (Spradbery, 1973), but the risk and extent of wind gusts on TORF remain unspecified for many environments. The motion of flora and fauna in calm conditions and the disturbances caused by this is another consideration, as is the relative shift in the sun's location across the course of the day resulting in the shifting of shadows. In future, some of these problems could be mitigated by the use of sequence information and/or Bayesian treatment of the view matching task. That is, once a familiarity trail has been detected, there exists a means of generating a prior probability distribution of the familiarity peak in the next sidesweep.

Finally, the area that caused the greatest tuning effort for TORF was preventing overshoot at the end of the flight. Ultimately the approach of gradually reducing

forward velocity at the end of the flight was successful but the requirement of range inference could be avoided in alternative strategies. In recent work studying terrestrial learning walks, the notion of negative training views has been explored (Jayatilaka et al., 2018). If the biorobot continued the learning flight beyond the stop location, views could be tagged with an alternative instruction. If these views are experienced then the biorobot has moved too far and should turn about. Conversely the notion of nest range could provide a useful source of information for the initialisation of other guidance requirements, for example, when to initialise the descent procedure or the direction of an impending course change due to the presence of impending obstacles.

5.4.2 Discussion on the view matching pipeline

A critical component of the TORF behavioural model is the view matching pipeline. Inspired by the image quality assessment work for the human vision system, image comparison metrics that have a basis in image structure were implemented and compared with the intensity based view matching procedures that are commonly adopted for bio-mimetic view matching paradigms. The best intensity (NRMSE_DS_CA_GLOB) and structural (CWSSIM) methods had 100% RR rate in the test datasets (fig. 5.21) thus it is not possible to separate the metrics on this measure. In future studies it would be useful to probe this ability further by identifying problem areas for both methods.

A desirable property for a reference snapshot is a large catchment area. On this measure, the CWSSIM had a 100% increase in catchment area for a given reference image when compared to intensity based metrics with equivalent image resolution. In order to understand why complex wavelets have wide catchment areas it is useful to consider the processing steps that have a bearing on purely spatial image difference comparison; blurring and image downsampling. Both of these operations have a low-pass filtering effect. If either are applied too vigorously, the loss of spatial localisation and image discriminability renders the method useless as explored in figure fig. 5.32. The wavelet transform, which can also benefit from blurring and downsampling, has an additional means of increasing the catchment area of a view which is encoded in the wavelet analysing function. As Gabor intimated (Gabor, 1946), the Heisenberg uncertainty principle dictates that as the minimum passband of a spatio-frequency filter is reduced, the less precisely its response is localised in space. For view matching, this *fuzziness* in spatial precision turns out to be a desirable characteristic, a notion explored further in fig. 5.35.

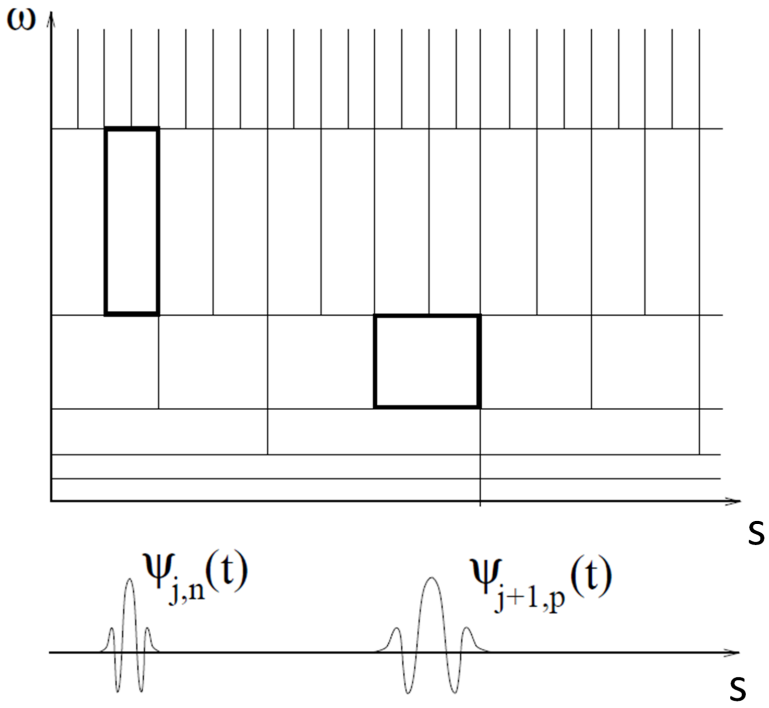


Figure 5.35: Modified from (Mallat, 2008) with permission from Elsevier. Spatio-frequency boxes demonstrate there relationship between the spatio-frequency domain and Heisenberg's uncertainty principle. Morlet wavelet functions in the jth and j+1th layer are shown. In the jth layer the wavelet coefficients contain have double the spatial granularity of the j+1th layer but the j+1th layer can detect lower frequency signals than on the jth layer.

Conceptually, the response of the CWSSIM is predominantly driven by a constellation of edge primitives with a fuzzy sensitivity to spatial configuration (where the edge primitive is in image space) and a fuzzy sensitivity to edge orientation and magnitude. The degree to which either of these properties are fuzzy is contingent on the wavelet analysing function. In this work, a pre-existing wavelet transform was employed for the analysis. The steerable pyramid with a second order Gaussian wavelet basis function was selected due to its reported translational invariance and because of the feasibility of running it onboard the MAV. The wavelet coefficients were analysed at a single spatial-frequency level. While this strategy worked well as a proof of concept, it is hypothesised that generating a wavelet basis function which is optimised for the decomposition of natural scenes may yield even more fruitful results. There is already a precedent in the early layers of the mammalian vision system of isotropic bandpass filtering cells which have a spatio-frequency basis in the primary and secondary optic

lobes. These so called V1 neurons (Bredfeldt and Ringach, 2002; Hubel and Wiesel, 1959) are widely cited as being the inspiration for the convolution layers of the now ubiquitous convolutional neural network. The identification of such cells in the optic lobes of foraging insects may provide clues on useful properties of such filters for the task of localising in natural scenes.

Ultimately, the multi-resolution aspect of wavelets was not exploited in this work, this is largely because the MAV's imaging device had a relatively narrow field of view and the inclusion of higher resolution wavelet subbands resulted in a narrowing of the image catchment area. For wider fields of view, multi-resolution may have more applicability, for example, as a means of pruning views before a more detailed comparison at a higher subband level. The complex wavelet steerable pyramid's ability to decompose an image into a parameterised set of subbands and resolutions proved a useful method for investigating biologically plausible image structural comparison techniques. In future studies that don't require multiresolution analysis, a Gabor type filter arrangement may prove just as useful as a basis function. For more efficient methods that are less focused on biological plausibility, the use of orientated Sobel filters in conjunction with *histogram of gradient* descriptors have been proven to work well in an object detection context.

Previous work on insect vision has been dominated by the EMD circuit which already provides a mechanism for the detection of orientated edges. The EMD could also be naturally extended to a multi-resolution format by correlating detector units between non-contiguous ommatidia. However, there are two challenges with the EMD circuit 1) Egomotion is required for reliable edge detection and this egomotion must presumably be consistent with the biorobot's motion when the reference view memory was created 2) The EMD output confounds intensity gradient and spatial layout. For these reasons the DC component of signal that is known to propagate into the medulla (Juusola et al., 1995) may also be a fruitful source of exploration for this kind of behaviour.

Given the apparently *featureless* natural texture of the test environment (a grassy field), the discriminability of reference views was of particular surprise. Not a single false positive was identified by the CWSSIM procedure in the test environment, yet ask a human to identify a particular patch of grass in a field from a downward facing image and this would be a near impossible task. The unexpectedly large hamming distance between views in this type of scene permits the consideration that flying insects could geometrically localise in the 2D task space. While this is an avenue

for deeper consideration, here speculation was limited to the notion that views could be associated with a range from the target. This helped to increase the robustness of homing flights, for example, by reducing the longitudinal component of velocity as the biorobot approached its destination.

Chapter 6

Conclusion

The purpose of this thesis was to extend current insect navigation behavioural models (inspired by desert ant behaviour) to the aerial context of foraging bees and wasps. In particular, two complementary behavioural models were investigated; path integration and visual navigation. Path integration is believed to be a key mechanism in invertebrate local navigation, in some instances operating over several kilometres. However, cumulative errors are a feature of path integration which, as evidenced in Chapter 4, means that animals navigating with this method may be out of visual contact with their destination when the path integration circuit is discharged. This homing error could be in the order of dozens of meters requiring the insect to search for its destination in a large area. By contrast, visual navigation has a limited range but as demonstrated in Chapter 5, provides a potential means of bridging the navigational gap left by path integration, thereby reducing the overall amount of time a foraging insect expends when searching for its destination.

In order to deploy each of the target behavioural models, this thesis reports the development of a quadcopter biorobot as detailed in (Chapter 4). The uncompromising stance on deploying models onboard a physical robotic platform in the real world has necessitated innovation. In particular, models that were developed using visual input from simulation environments have not translated readily to the biorobot without additional visual processing and behavioural adaptations.

6.1 Outcomes

The outcomes of this work are summarised by chapter below:

6.1.1 Chapter 3: An aerial platform for testing insect inspired navigation algorithms

Robotic platforms have proved a useful substrate for evaluating models of animal behaviour. In order to deploy these models in a biologically relevant fashion, an interface between the model logic and an appropriate dynamical system is required. For flying insect navigation, a candidate biorobot should be able to move around in 3D space and conduct similar manoeuvres as the biological analogue. The suitability of a quadcopter as a platform for embodying flying insect navigation models is reviewed in Chapter 3. 2-axis view stabilisation is deemed to be a necessary feature for achieving biologically relevant system output controllability. An open source MAV hardware platform and corresponding software stack is developed and described. A combination of COTS components and bespoke parts is detailed and the resultant platform is evaluated in order to ensure that all control interfaces operate as intended. A typical University engineering workshop should be able to reproduce the open-source system and readily adapt it to their needs.

6.1.2 Chapter 4: Deploying the central complex path integration circuit on an MAV

An anatomically plausible candidate neural circuit for path integration in flying foraging insects was recently developed in (Stone et al., 2017). However, in this work the model was not validated on a platform that accurately portrays the fully holonomic, 3D task space of the target species. Furthermore, an untested core assumption of this model is that flying insects can accurately estimate their egomotion. While motion detection circuits are well studied in insects, their ability to accurately judge translational distance is not yet established. By extension, the efficacy of a central complex path integration circuit, and the extent to which flying insects can rely on this mechanism is unspecified.

In order to quantify the performance of the central complex model developed in (Stone et al., 2017), it was deployed on an aerial platform and tested in the real world with outbound distances of up to 150m. An optic flow speed estimation circuit was developed using a matched filter approach. Biologically plausible image resolutions (3° acuity) were used as the input signal. From this circuit it was demonstrated that active mechanical stabilisation of the vision sensor improves the performance of the speed estimation circuit. By evaluating different permutations of view direction and visual

environment, it is concluded that a view towards the ground provides the most reliable speed estimation information. This is in agreement with experimental observations (Linander et al., 2017) but as yet there is no direct evidence that flying bees or wasps have sufficient visual acuity towards the ground to support this. The work here however shows that a small array of pixels is sufficient for estimating optic flow in natural scenes. In a flat environment the homing error, which is a function of outbound distance, is found to be lower than 10m at the outbound distance of 150m. The model is also tested in a range of simulated worlds. Performance degrades in scenes with strong height discontinuities but the mean homing error remains below 10m for these worlds provided that there is sufficient ground texture.

6.1.3 Chapter 5: Complex wavelet structural similarity for view based image matching

Flying insects are known to use visual navigation in order to overcome the path integration circuit limitations that were quantified in Chapter 4. In the last decade, a popular hypothesis has been that insects sample multiple view memories at regular intervals along important routes and these memories can be used to guide future journeys along these routes (Ardin et al., 2016; Baddeley et al., 2012). However, successful deployment of this strategy has to date been restricted to simulation environments, or ground based robotic platforms travelling over short distances (<10m). Flying insects by contrast, rely on visual navigation to find their way over modest distances (15m+) in a 3D task space.

In order to verify that visual route following could provide a robust navigation cue for flying insects, an adapted visual route following method was developed and deployed on the aerial platform presented in Chapter 3. The method exploits the fact that flying insects can use transverse translational motion as a means of performing image difference functions rather than the rotational image difference procedure thought to occur in terrestrial insects. As with existing methods, a prerequisite for visual navigation is a learning activity whereby visual snapshots are sampled from salient locations and stored in memory. Here it is proposed that the direct outbound and inbound flights of flying insect learning flights (shown earlier in fig. 5.1) provide an opportunity for insects to periodically record visual snapshots along a straight route radiating from the site of importance. On return journeys, an agent that intercepts this chain of snapshots is able to follow the sequence of stored images to the nest as though it were a landing

strip. This concept performs robustly in outdoor natural scenes when deployed on the biorobot.

Using the CWSSIM procedure, an improvement in place recognition and a doubling of the catchment area of unprocessed visual snapshots is achieved. The bandpass operation that is involved in complex wavelet decomposition was inspired by studies in the human vision system. Thus this appears to be a more biologically plausible snapshot mechanism than previously suggested spatial or frequency domains which are both somewhat contingent on uniform sampling. It is shown across multiple trials that the complex wavelet structured similarity metric provides enough discriminability to effectively localise in a large area around a site of importance.

6.2 Discussion and future work

Understanding how flying insects can navigate their habitat with limited information processing resources has been an area of considerable active research for at least a century. Even in terrestrial platforms where sensor resolution and computational power are relatively unconstrained in size and weight, man-made systems are not yet able to navigate in certain environments as competently as insects, if at all. The boom in MAV technology makes the desire to unlock the secrets of the insect navigational toolkit as relevant as ever.

This thesis made a first pass at applying the base insect navigation model (Webb, 2019) to an aerial biorobotic platform, supporting the contention that the base model provides an end-to-end description of how flying insects can navigate large areas without getting lost. In order to iterate this work further, additional behavioural datasets are first required to provide a ground truth for the evaluation of the models developed herein. In the remainder of this section, the predictions of this thesis are outlined and suggested future work streams are presented.

6.2.1 Limitations of existing datasets

Harmonic radar trajectories (Capaldi et al., 2000) provide a phenomenal resource in terms of describing the capabilities and long-range foraging strategies of the species-of-interest. However, the low accuracy, low sample rate, 2D tracks produced by this method do not yield much information about a particular insect's flight dynamics and how the course it plots relates to its visual experience. There are fundamental un-

answered questions relating to the motion of flying foraging insects in natural outdoor conditions. How high do they fly above the ground? What profile does their flight path have? How do these properties vary (spontaneously and with environmental and topographical variation)? What happens when an obstacle is encountered? How repeatable are these actions?

Tests conducted in flight tunnels have revealed a great deal about the visually mediated flight control system of flying insects. Spatially precise and high frequency datasets have been recorded with stereo camera systems mounted on the tunnels. However, because of the unnatural conditions provided by flight tunnels, it is not possible to answer the questions raised in the previous paragraph with this data. Stereo cameras have also been used to precisely map out the 3D tracks of insects in confined volumes in the field (Stürzl et al., 2016). However, to date the size of these tracks has also precluded answers to the questions raised in the previous paragraph.

Recording entire foraging trajectories with the desirable fidelity is perhaps beyond the scope of modern research efforts but documenting carefully selected sectors of the trajectories of the species-of-interest could help to close some of the existing knowledge gaps and permit the development of more biologically relevant models and test schedules.

It should be possible to use an array of high resolution cameras and/or time-of-flight sensors to accurately track insect trajectories over larger sectors than has been previously been achieved. For example, the pair of cameras used in (Stürzl et al., 2016) had a resolution of 1280x1024 and the wasp was large enough in the frame that its heading could be inferred but the setup had an operating volume of just $50cm^3$. Using a combination of higher resolution cameras, a greater number of cameras in the rig and relaxing the required frame rate and requirement to track the insect's heading it is reasonable to assume that a considerably larger volume could be monitored. Another previously adopted approach for tracking flies via a standard vision system was to track the animals with a human operated, instrumented pan and tilt system (Fry et al., 2000) and to specify the tracked animal's position using inverse kinematics. It is beyond the scope of this work to recommend an ideal setup, but for the purpose of the discussion an assumption is made that with a tractable research effort, a honey bee sized animal could be tracked in an outdoor volume of space with the approximate dimensions; 10x5x5m.

6.2.2 Combining path integration and visual navigation

An obvious next step towards modelling the base insect navigation model would be to combine the path integration and visual navigation algorithms into a unified model, for example using the approach outlined in (Sun et al., 2019). The path integration circuit already exhibits an emergent systematic search behaviour[1] when it is in a discharged state. The simplest combined approach would be for the agent to engage in this search behaviour until a view with high familiarity is detected by the visual navigation circuit at which point the mechanisms described in Chapter 5 take over. However, the trajectories that would be generated by this approach have not been observed in live animals. In general, it will be difficult to validate the biological relevance of a combined path integration/visual navigation model without a reference biological dataset.

By recording the trajectories of flying insects approaching their nests (with the approach discussed in section 6.2.1), a change in flight course that may signify a transition between path integration and visual navigation could be detected. A solitary ground nesting wasp would be a good study candidate for this because these insects forage over shorter distances than honey bees and are less likely to use odour as a navigational cue. Ground nesting wasps are also thought to approach the nest from a consistent direction (Tinbergen, 1972). If this turns out to be the case this would reduce the total area that is required to be monitored. According to the work herein, such a transition would occur within 15m of the nest. The model presented in Chapter 5 depended on the insect embarking on a flight with an oscillatory profile, it may also therefore also be a good opportunity to test this hypothesis and indeed an increased tendency to travel with a sinuous profile may indicate a behavioural transition from path integration to visual navigation.

6.2.3 Extended learning flights

In section 6.2.2, the measurement of inbound flights at the site of a solitary wasp's nest is proposed. It would also be worth extending the work of (Stürzl et al., 2016) as a concurrent activity. Specifically, the trajectories of an individual insect's return flight are compared to those of its outbound flights. Two intermediate learning flight behaviours are anecdotally reported in (Spradbery, 1973) (p. 133). The first being a looping flight directly above the nest and the second is a turn back and look arcing process that occurs approximately 10m radially from the animal's nest. The documentation

[1] This was suppressed in Chapter 4 but is evidenced in (Stone et al., 2017))

of this process in conjunction with the trajectories of subsequent return flights and a laser scan of the habitat (as per (Stürzl et al., 2015)) would provide an invaluable visual navigation modelling resource.

The route following procedure outlined in Chapter 5 predicts that all homing flights should have a close spatial alignment with an animal's learning flights because ventral views have limited catchment areas. This is not necessarily the case with visual navigation strategies that are driven by horizontal views, especially in open ground. Therefore, the collection of learning flights at the same site as the homing flights could be a useful resource for determining what visual information is attended by a particular animal in its habitat.

6.2.4 Height estimation in flying insects

A significant assumption in both the path integration and visual navigation models of this thesis is the ability of the agent to cruise at a fixed and/or known height above the ground. In the literature, an optic flow regulation strategy is frequently cited as the sole means of governing height control in flying insects (Serres and Ruffier, 2017). In the natural world, optic flow estimates are modulated by scene depth variation and environmental conditions. According to the results in Chapter 4, modest disturbances in height across a foraging route result in a significant reduction in the performance of an optic flow based path integration system. It seems unlikely that the primary long-range navigation system would be so vulnerable to corruption. Conversely, the availability of good quality height estimates in the insect brain, as assumed in this thesis, is equally controversial. Other mechanisms, such as sensor fusion, speed filtering and the rejection of implausible estimates may also have a part to play. The approach outlined in section 6.2.1 could be used to interrogate possible models of height estimation and control of flying insects in open outdoor environments.

A *relatively* straightforward procedure would be to setup an observation volume between a hive and a feeder site in flat open ground. By recording the paths of honey bees flying through this sector, the variation between topography, insect speed and insect height could be quantified for individual bee flights. The local wind speed and direction would also need to be monitored (with time stamping) as this is already known to have a bearing on the flight path of foraging bees. The work in this thesis predicts that there could be considerably more variation in the ratio of an insect's speed and height than would be predicted by the optic flow regulator. This could be visible in the scope

of an individual flight trajectory given a large enough recording volume. It would also be interesting to monitor the variation of speed/height ratios in a population of animals to establish what extent this value is influenced by environmental and physiological factors. Another prediction is that some active manoeuvres may be involved in the process of estimating height as discussed below. If the trajectory recording procedure worked in flat terrain, a comparison with sites above undulating and/or ramping surfaces could be even more insightful.

Monitoring successive return journeys of a solitary ground nesting wasp as proposed in section 6.2.2 would also yield a useful dataset for specifying the required tolerance to height variation in the visual homing task. The visual homing model presented in Chapter 5 is dependent on the animal searching at a relatively consistent height above the ground ($\pm 10\%$ according to fig. 5.34). By contrast, the optic flow regulator theory predicts that the animal would fly with a fixed speed/height ratio (as has already been demonstrated in the landing manoeuvres of honey bees (Srinivasan, 2011), but the same mechanism does not manifest in a species of wasp (Stürzl et al., 2016)). If either of these hypotheses are valid they should be evidenced by the method outlined in this paragraph. In order to minimise confounding factors, the data collection should occur in a flat landscape.

While a biological dataset would clearly be a useful asset for an enriched understanding of the height control mechanisms involved in central place foraging flying insects, a pure modelling approach could also provide useful insight here. In this work a lidar range sensor was used as the primary control signal for the height control system. If accurate height measurements could be inferred with a biologically constrained vision system this would strengthen the case that the species-of-interest make use of such a mechanism. This is not a straightforward task but one approach would be to use a sensor fusion algorithm to combine optic flow, airspeed and the three visual height estimation processes outlined in (Straw et al., 2010). An active state estimation approach could also be investigated. For example, by deliberately dropping altitude at a known rate, an agent may be able to infer height by looking at the expansion patterns using the techniques outlined in (Sazbon et al., 2004).

6.2.5 Obstacle avoidance

A significant limitation of the biorobot developed in this thesis is the lack of any obstacle avoidance capabilities. This means that the biorobot can only operate in

open environments. There are three criteria that an obstacle avoidance system can be evaluated against in the context of this work:

1. ***Efficacy*** - How well does the obstacle avoidance system work? An effective system will be able to operate in many different environments.

2. ***Biological relevance*** - Does the system avoid obstacles in a way that produces similar trajectories to flying insects? A biologically relevant obstacle avoidance system would recreate the views experienced by the insect of study and is therefore also likely to be effective.

3. ***Biological plausibility*** - Does the system achieve obstacle avoidance in a method that could be achieved with the neuroanatomy of a flying insect?

These criteria are somewhat ordered in implementation feasibility and could be developed independently depending on the needs of a study. It should be relatively straight forward to add a range sensor to the front of the biorobot and integrate the output of a robot obstacle avoidance system with the behavioural model. The pixhawk flight controller already has an inbuilt obstacle avoidance feature which operates on sparse point cloud data. Skirting obstacles in a manner that resembles an insect's approach would require the collection of some data about how insects interact with obstacles in outdoor spaces. Creating a biologically plausible obstacle avoidance system would require adaptations to the vision system. Ideally it would also be effective and biologically relevant.

Once again, an investigation into the way that insects negotiate obstacles of different size and stature in the real world would provide a useful asset. Since substantial physical barriers would be required for this task the experiment would ideally be undertaken at sites where obstacles between a hive and a feeder naturally occur. For some obstacles, such as long masonry walls, the animal is forced to go over the top of the barrier. In cases where the obstacle is tall and narrow, such as an isolated tree, the animal may just skirt around it. It would be interesting to compare properties of the trajectories of animals in open ground with those in flight tunnels. For example, how far in advance of an obstacle does an insect change its course.

As this thesis predicts that flying insects compute an online estimate of their height above the ground, a hypothesis that insects travelling over obstacles do not necessarily maintain a fixed speed/height ratio can be made. Frontal vision areas are likely to play a key part in the detection of oncoming obstacles, this advanced warning of an obstacle

could be used to improve odometry estimates or increase the weight of alternative speed sensing cues in a multi-modal system. Visual navigation may also play a part in dealing with obstacles via the triggering of local vectors (Collett, 2012; Collett and Collett, 2002).

In Chapter 4, odometry derived from ventral views was shown to be relatively tolerant to obstacles whereas odometry derived from horizontal views was intolerant to the presence of obstacles. There are some habitats where lateral cues may be more salient than ventral cues, a rainforest tree canopy for example. Development of an effective biologically relevant obstacle avoidance system would enable a comparison between optic flow based odometry with different view directions.

6.2.6 Salient view directions

This project has predominantly addressed navigation in open environments, primarily because this is the type of environment that harmonic radar datasets have already been collected in, but also because this meant that a solution for obstacle avoidance was not required. Despite this apparent simplification, the prospect of navigating to a precise geographic location in an environment of seemingly featureless terrain seemed like a daunting prospect at the outset of this project. Success at this task for both path integration and visual homing has depended on focusing on visual information with a ventral aspect. Whether the insect actually adopts this policy is not yet known.

To some extent the prediction that flying insects preferentially use ventral optic flow information is already behaviourally evidenced by the flight control experiments outlined in (Linander et al., 2017). It would be useful to conduct a survey of the downward facing region of the eyes of honey bees that are know to navigate in arable land. This would also require knowledge of the orientation of the animal's eyes (or head) during flight. Another way to approach this problem would be to investigate the full receptive field of the path integration speed sensitive (TN) neurons in a variety of central place foraging insects. At the time of writing, the receptive field of TN neurons is only known for a horizontal motion-field. It should also be possible to probe the relative contribution of horizontal and ground based information for visual homing by placing these stimuli in conflict in an electropysiology rig. Ventral optic flow was shown to be the most dependable in the environments tested in Chapter 5. It is therefore predicted that TN cell output will be correlated with ventral optic flow rates when this is in conflict with optical flow presented in other regions of a given test animal's eye.

Horizontal versus ground facing cue conflicts for visual navigation could be investigated at a behavioural level. A potential scheme for this is presented in fig. 6.1. According to the findings of this thesis, it is hypothesised that the insect's search will be correlated with ground information rather than the orientation of the walls, provided that the radius of the arena is sufficiently large.

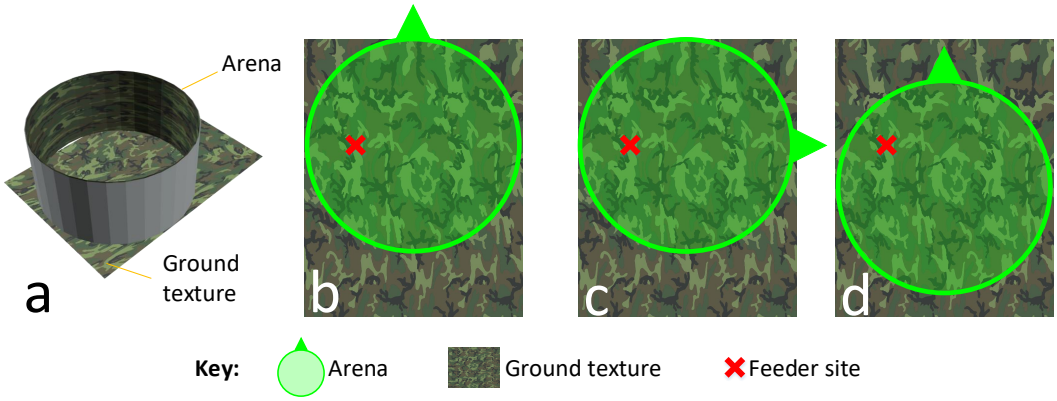


Figure 6.1: Potential experiment to ascertain which part of a flying insect's visual feed has the greatest weight when locating a feeder. The animal is trained to a discrete feeder in the ground of an arena (setup shown in **a**). The feeder is then removed and the returning animal's search pattern is monitored (**b**). The experiment is repeated for other animals in one of two test conditions; (**c**), the arena horizontal wall is rotated by a predefined amount. (**d**) the ground texture is shifted by a predefined amount.

6.2.7 Visual processing

While the use of holistic images is ubiquitous in insect navigation modelling paradigms, it is unlikely that an insect's brain propagates and stores a time synchronised chunk of data representing photon intensity at the retina at a particular instance in time. A more likely hypothesis is that a chain of preprocessing operations filter out redundant information and the remaining salient information is distilled into a compact representation which can be readily stored in the mushroom bodies. Bandpass filters are thought to be prominent in the early stages of the mammalian vision system (e.g V1 cells (Bredfeldt and Ringach, 2002)), while there are considerable differences between mammalian and insect vision systems, it would not be surprising for there to be similarities in the early stages of vision processing systems of both animal groups.

The CWSSIM procedure, which is based on the analysing functions of the human

vision system, was selected over other wavelet basis functions in part because of the possibility to run this procedure on digital images in real time. The complex wavelet transform acts as multi-orientation, multi-scale bandpass filter. In future it would be interesting to examine which bandpass filters could be implemented on data sampled from an insect's eye. Insect eye facets are often hexagonal (Kim et al., 2016), a sixfold orientation basis function could have more discriminability than the two orientations used herein. It would also be interesting to investigate how snapshot like representations can be achieved with more biologically equivalent, asynchronous hardware, such as the output of an event based camera Gallego et al. (2020).

6.2.8 Robotic applications

Given the robustness of the CWSSIM procedure in Chapter 5, it is hypothesised that this kind of approach could be explored further in a purely robotic context. The use of low resolution holistic descriptors could be exploited by SLAM algorithms that operate in natural environments. While not suitable for all paradigms, the ability to perform loop closure and/or prune keyframe information would be a useful augmentation to many contemporary SLAM approaches. The approach used here could also provide an independent means of travelling between key locations for platforms where the overhead of a full SLAM algorithm is not desirable. This could provide a redundant system for GNSS denied segments of a route or in case of GNSS hardware failures. Finally, the ability to hover over a key point is often achieved in MAV applications by the placement of a fiducial marker. The procedure used here could be used to formulate such markers in natural scenes which can be allocated arbitrarily and on the fly.

Dead reckoning odometry systems that are functionally equivalent to path integration circuits already exist for MAV systems (Honegger et al., 2013). However, these systems typically require a range sensor which can be expensive and/or unreliable in certain scenarios. If the mechanisms that flying insects use to estimate their ground height could be identified and engineered into a vision only system, this could have considerable application in the MAV industry. The matched filter approach pursued in Chapter 4 offers one route towards this problem. Matched filters may provide vision only solutions to other problems that engineers are currently relying on expensive RGB-D sensors. For example, a matched filter could perform the basis of a landing aid system for VTOL aircraft which ensures that an aircraft's landing terrain is sufficiently flat and obstacle free before it is permitted to land.

Glossary

allothetic	A navigation method using an external fixed frame of reference as apposed to an egocentric reference
biorobot	Biorobots are described as an intersection between robotics and biology. While this general definition includes robotic applications that have some general inspiration from nature, here the definition of a biorobot is a robot that has been designed to investigate an ethological hypothesis (unless otherwise specified).
central place forager	Animals that transport resources back to a particular site. A trait common in Hymenoptera which bring food back to a nest for the benefit of their young and/or social group.
holonomic	A holonomic system can be controlled in each of its degrees of freedom. A quadcopter can be controlled about all of its axes of motion and is holonominc, whereas a car is unable to translate sideways and is therefore non-holonomic.
idiothetic	A navigation method using self motion cues rather than allothetic cues

middleware	A software layer that enables convenient standardised interfacing and data exchange between different software modules.
quadcopter	A VTOL MAV with 4 motors.
Reynolds number	The ratio of inertial and viscous forces.
thrust margin	The ratio of an aircraft's weight and the level of thrust its propulsion system can generate.
visual homing	A visual navigation mechanism associated with inferring a goal direction from a single snapshot with potentially a large spatial displacement from the reference snapshot.
visual navigation	A navigation mechanism that operates by comparing a current view with a stored view or views and inferring a desired motion based on this process
visual route following	A visual navigation mechanism associated with the following of a chain of visual homing memories formed along a route, thus allowing visual navigation to occur well beyond the catchment area of a single view memory.

Appendix A

Extended visual homing routes

A set of visual navigation experiments with extended 30s outbound routes was conducted and the results are included in fig. A.1. While it is apparent the inbound agent successfully traverses the outbound route there were some issues with the agent stopping at the mission start location on some of the flights (see flight a,b & d)due to some incorrect parameter settings. The results are however included in this appendix because this is evidence that the TORF procedure can operate on routes of up to 30m and with up to 300 image memories describing the outbound route. The failure to stop at the departure point highlights the brittle nature of using parameters to shape the TORF homing procedure.

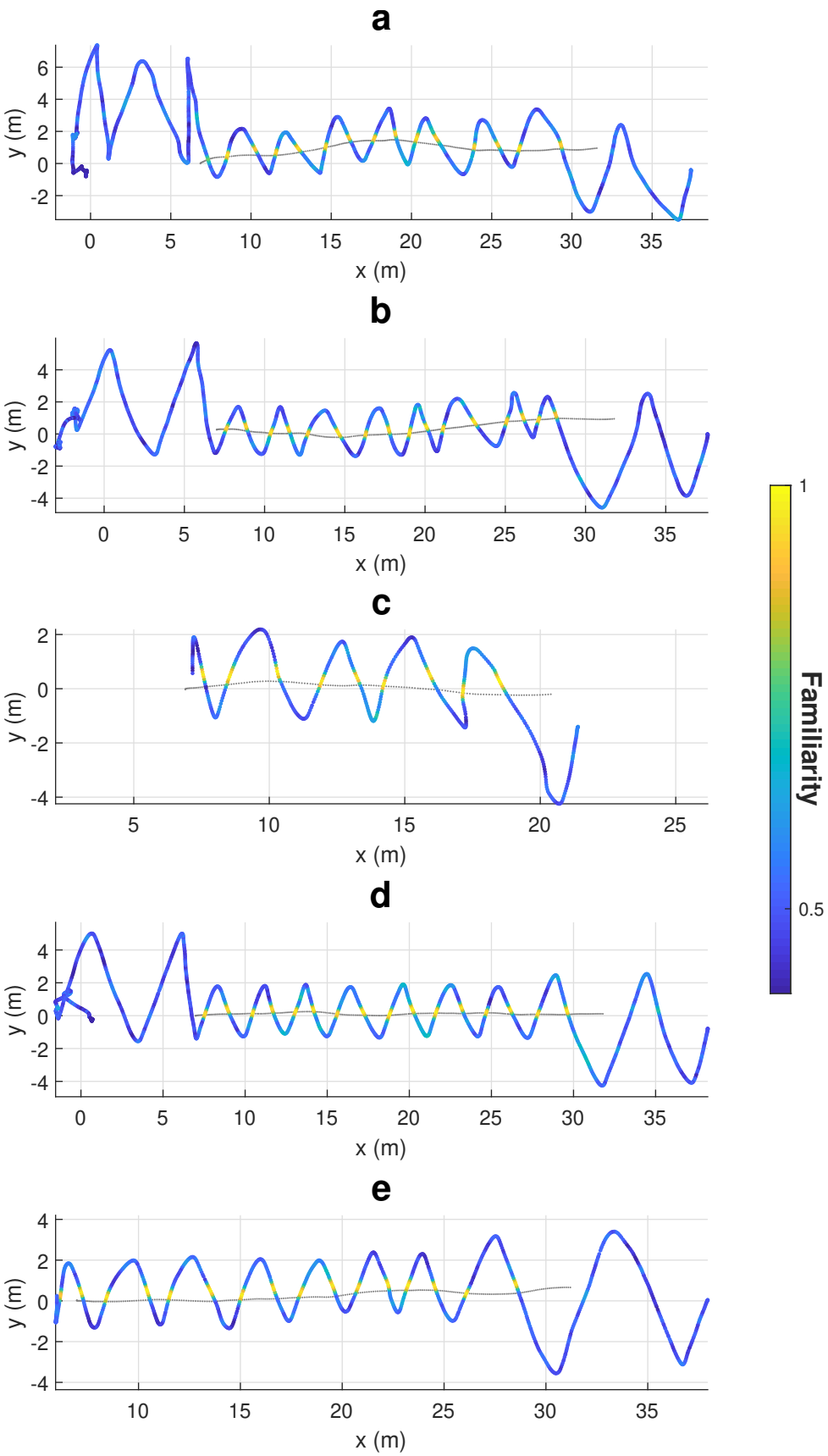


Figure A.1: Scatter plot of 5 outbound (small grey circles) and corresponding inbound (filled coloured circles, coloured according to the familiarity level) trajectories generated by operation of the TORF procedure outlined in chapter 5.

Bibliography

Ai, H., Okada, R., Sakura, M., Wachtler, T., and Ikeno, H. (2019). Neuroethology of the waggle dance: How followers interact with the waggle dancer and detect spatial information. *Insects*, 10.

Ai, H., Rybak, J., Menzel, R., and Itoh, T. (2009). Response characteristics of vibration-sensitive interneurons related to johnston's organ in the honeybee, apis mellifera. *Journal of Comparative Neurology*, 515(2):145–160.

Alsayed, Z., Bresson, G., Verroust-Blondet, A., and Nashashibi, F. (2017). Failure detection for laser-based slam in urban and peri-urban environments. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–7.

Ardin, P., Peng, F., Mangan, M., Lagogiannis, K., and Webb, B. (2016). Using an Insect Mushroom Body Circuit to Encode Route Memory in Complex Natural Environments. *PLoS Computational Biology*, 12(2).

Baddeley, B., Graham, P., Husbands, P., and Philippides, A. (2012). A model of visual route navigation in ants without waypoints. *Behavioral Neuroscience Archive*, 6(261).

Badrya, C., Govindarajan, B., and Baeder, J. (2017). Analysis of flapping wings from a rotorcraft perspective.

Baird, E., Srinivasan, M. V., Zhang, S., and Cowling, A. (2005). Visual control of flight speed in honeybees. *Journal of Experimental Biology*, 208(20):3895–3905.

Barron, A. B. and Plath, J. A. (2017). The evolution of honey bee dance communication: a mechanistic perspective. *Journal of Experimental Biology*, 220(23):4339–4346.

Basten, K. and Mallot, H. (2010). Simulated visual homing in desert ant natural environments: Efficiency of skyline cues. *Biological cybernetics*, 102:413–25.

Bomphrey, R., Taylor, G., and Thomas, A. (2010). Smoke visualization of free-flying bumblebees indicates independent leading-edge vortices on each wing pair. *Animal Locomotion*, 46:249–259.

Borst, A., Haag, J., and Mauss, A. (2019). How fly neurons compute the direction of visual motion. *Journal of Comparative Physiology A*, 206.

Bredfeldt, C. and Ringach, D. (2002). Dynamics of spatial frequency tuning in macaque v1. *The Journal of Neuroscience*, 22:1976–1984.

Briscoe, A. D. and Chittka, L. (2001). The evolution of color vision in insects. *Annual Review of Entomology*, 46(1):471–510.

Buatois, A. and Lihoreau, M. (2016). Evidence of trapline foraging in honeybees. *Journal of Experimental Biology*, 219(16):2426–2429.

Buehlmann, C., Mangan, M., and Graham, P. (2020). Multimodal interactions in insect navigation. *Animal Cognition*.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. (2016). Simultaneous localization and mapping: Present, future, and the robust-perception age. *IEEE Transactions on Robotics*, 32.

Capaldi, E., Smith, A., Osborne, J., Fahrbach, S., Farris, S., Reynolds, D., Edwards, A., Martin, A., Robinson, G., Poppy, G., and Riley, J. (2000). Ontogeny of orientation flight in the honeybee revealed by harmonic radar. *Nature*, 403:537–40.

Cartwright, B. A. and Collett, T. (1987). Landmark maps for honeybees. *Biological Cybernetics*, 57:85–93.

Cartwright, B. A. and Collett, T. S. (1983). Landmark learning in bees. *Journal of comparative physiology*, 151:521–543.

Chahl, J. (2015). Unmanned aerial systems (uas) research opportunities. *Aerospace*, 2:189–202.

Chatterjee, A., George, E., Prabhudev, M., Basu, P., and Brockmann, A. (2019). Honey bees flexibly use two navigational memories when updating dance distance information. *The Journal of Experimental Biology*, 222:jeb.195099.

Cheeseman, J. F., Millar, C. D., Greggers, U., Lehmann, K., Pawley, M. D. M., Gallistel, C. R., Warman, G. R., and Menzel, R. (2014). Way-finding in displaced clock-shifted bees proves bees use a cognitive map. *Proceedings of the National Academy of Sciences*, 111(24):8949–8954.

Cheung, A., Collett, M., Collett, T. S., Dewar, A., Dyer, F., Graham, P., Mangan, M., Narendra, A., Philippides, A., Stürzl, W., Webb, B., Wystrach, A., and Zeil, J. (2014). Still no convincing evidence for cognitive map use by honeybees. *Proceedings of the National Academy of Sciences*, 111(42):E4396–E4397.

Chikasha, P. N. and Dube, C. (2017). Adaptive model predictive control of a quadrotor. *IFAC-PapersOnLine*, 50(2):157 – 162. Control Conference Africa CCA 2017.

Collett, M. (2010). How desert ants use a visual landmark for guidance along a habitual route. *Proceedings of the National Academy of Sciences*, 107(25):11638–11643.

Collett, M. (2012). How navigational guidance systems are combined in a desert ant. *Current biology : CB*, 22:927–32.

Collett, M., Chittka, L., and Collett, T. (2013). Spatial memory in insect navigation. *Current Biology*, 23(17):R789 – R800.

Collett, M. and Collett, T. S. (2009). Local and global navigational coordinate systems in desert ants. *Journal of Experimental Biology*, 212(7):901–905.

Collett, M., Harland, D., and Collett, T. S. (2002). The use of landmarks and panoramic context in the performance of local vectors by navigating honeybees. *Journal of Experimental Biology*, 205(6):807–814.

Collett, T. S. (2019). Path integration: how details of the honeybee waggle dance and the foraging strategies of desert ants might help in understanding its mechanisms. *Journal of Experimental Biology*, 222(11).

Collett, T. S. and Collett, M. (2002). Memory use in insect visual navigation. *Nature Reviews Neuroscience*, 3:542–552.

Conroy, J., Gremillion, G., Ranganathan, B., and Humbert, J. (2009). Implementation of wide-field integration of optic flow for autonomous quadrotor navigation. *Autonomous Robots*, 27:189–198.

Coombs, D. and Roberts, K. (1992). 'Bee-bot': using peripheral optical flow to avoid obstacles. In Casasent, D. P., editor, *Intelligent Robots and Computer Vision XI: Algorithms, Techniques, and Active Vision*, volume 1825, pages 714 – 721. International Society for Optics and Photonics, SPIE.

Cope, A. J., Ahmed, A., Isa, F., and Marshall, J. A. R. (2019). Minibee: A minature mav for the biomimetic embodiment of insect brain models. In *Biomimetic and Biohybrid Systems*, pages 76–87, Cham. Springer International Publishing.

Cope, A. J., Sabo, C., Gurney, K., Vasilaki, E., and Marshall, J. A. R. (2016). A model for an angular velocity-tuned motion detector accounting for deviations in the corridor-centering response of the bee. *PLOS Computational Biology*, 12(5):1–22.

Cruse, H. and Wehner, R. (2011). No need for a cognitive map: Decentralized memory for insect navigation. *PLOS Computational Biology*, 7(3):1–10.

Dacke, M. and Srinivasan, M. V. (2008). Two odometers in honeybees? *Journal of Experimental Biology*, 211(20):3281–3286.

Degen, J., Hovestadt, T., Storms, M., and Menzel, R. (2018). Exploratory behavior of re-orienting foragers differs from other flight patterns of honeybees. *PLoS ONE*, 13.

Degen, J., Kirbach, A., Reiter, L., Lehmann, K., Norton, P., Storms, M., Koblofsky, M., Winter, S., Georgieva, P., Nguyen, H., Chamkhi, H., Greggers, U., and Menzel, R. (2015). Exploratory behaviour of honeybees during orientation flights. *Animal Behaviour*, 102:45–57. cited By 38.

Degen, J., Kirbach, A., Reiter, L., Lehmann, K., Norton, P., Storms, M., Koblofsky, M., Winter, S., Georgieva, P., Nguyen, H., Chamkhi, H., Meyer, H., Singh, P., Manz, G., Greggers, U., and Menzel, R. (2016). Honeybees learn landscape features during exploratory orientation flights. *Current Biology*, 26:2800–2804.

Denuelle, A. and Srinivasan, M. V. (2016). A sparse snapshot-based navigation strategy for uas guidance in natural environments. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3455–3462.

Denuelle, A., Thurrowgood, S., Kendoul, F., and Srinivasan, M. (2015). A view-based method for local homing of unmanned rotorcraft. *ICARA 2015 - Proceedings of the 2015 6th International Conference on Automation, Robotics and Applications*, pages 443–449.

Dickinson, M., Lehmann, F.-O., and Sane, S. (1999). Wing rotation and the aerodynamic basis of insect flight. *Science (New York, N.Y.)*, 284:1954–60.

Dickinson, M. H. and Muijres, F. T. (2016). The aerodynamics and control of free flight manoeuvres in drosophila. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 371(1704).

Differt, D. and Stürzl, W. (2020). A generalized multi-snapshot model for 3d homing and route following. *Adaptive Behavior*, 0(0):1059712320911217.

Dittmar, L., Stürzl, W., Baird, E., Boeddeker, N., and Egelhaaf, M. (2010). Goal seeking in honeybees: matching of optic flow snapshots? *Journal of Experimental Biology*, 213(17):2913–2923.

Douglass, J. K. and Strausfeld, N. J. (2001). *Pathways in Dipteran Insects for Early Visual Motion Processing*, pages 68–82. Springer Berlin Heidelberg, Berlin, Heidelberg.

Duckett, T. and Nehmzow, U. (1998). Mobile robot self-localisation and measurement of performance in middle-scale environments. *Robotics and Autonomous Systems*, 24(1):57 – 69. Scientific Methods in Mobile Robotics.

Dyer, F. C. and Dickinson, J. A. (1994). Development of sun compensation by honeybees: how partially experienced bees estimate the sun's course. *Proceedings of the National Academy of Sciences of the United States of America*, 91 10:4471–4.

Esch, H. and Burns, J. (1996). Distance estimation by foraging honeybees. *Journal of Experimental Biology*, 199(1):155–162.

Evans, H. E. (1966). The behavior patterns of solitary wasps. *Annual Review of Entomology*, 11(1):123–154.

Expert, F. and Ruffier, F. (2015). Flying over uneven moving terrain based on optic-flow cues without any need for reference frames or accelerometers. *Bioinspiration & Biomimetics*, 10(2):026003.

Farnebäck, G. (2003). Two-frame motion estimation based on polynomial expansion. *Image analysis*, 2749:363–370.

Franceschini, N., Ruffier, F., and Serres, J. (2007). A bio-inspired flying robot sheds light on insect piloting abilities. *Current Biology*, 17(4):329 – 335.

Franz, M., Mallot, H., Egelhaaf, M., and Dahmen, H. (2000). Wide-field, motion-sensitive neurons and optimal matched filters for optic flow. *Biological Cybernetics*, 83:185–197.

Franz, M. O. and Krapp, H. G. (2000). Wide-field, motion-sensitive neurons and matched filters for optic flow fields. *Biological cybernetics*, 83(3):185–197.

Fry, S., Bichsel, M., Müller, P., and Robert, D. (2000). Tracking of flying insects using pan-tilt cameras. *Journal of Neuroscience Methods*, 101(1):59 – 67.

Fuller, S. B., Karpelson, M., Censi, A., Ma, K. Y., and Wood, R. J. (2014). Controlling free flight of a robotic fly using an onboard vision sensor inspired by insect ocelli. *Journal of The Royal Society Interface*, 11(97):20140281.

Furrer, F., Burri, M., Achtelik, M., and Siegwart, R. (2016). *RotorS – A Modular Gazebo MAV Simulator Framework*, volume 625, pages 595–625.

Gabor, D. (1946). Theory of communication. part 1: The analysis of information. *Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering*, 93:429–441(12).

Gaffin, D. D., Dewar, A., Graham, P., and Philippides, A. (2015). Insect-inspired navigation algorithm for an aerial agent using satellite imagery. *PLoS ONE*, 10(4):e0122077.

Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A., Conradt, J., Daniilidis, K., and Scaramuzza, D. (2020). Event-based vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, PP.

Gkanias, E., Risse, B., Mangan, M., and Webb, B. (2019). From skylight input to behavioural output: A computational model of the insect polarised light compass. *PLOS Computational Biology*, 15(7):1–30.

Greggers, U., Koch, G., Schmidt, V., Dürr, A., Floriou-Servou, A., Piepenbrock, D., Göpfert, M. C., and Menzel, R. (2013). Reception and learning of electric fields in bees. *Proceedings of the Royal Society B: Biological Sciences*, 280(1759):20130528.

Gudmundsson, V., Kristinsson, H., Petersen, S., and Hasan, A. (2018). Robust uav attitude estimation using a cascade of nonlinear observer and linearized kalman filter.

In *ASME 2018 Dynamic Systems and Control Conference, DSCC 2018; Conference date: 30-09-2018 Through 03-10-2018*, volume 3. American Society of Mechanical Engineers. ASME 2018 Dynamic Systems and Control Conference, DSCC 2018 ; Conference date: 30-09-2018 Through 03-10-2018.

Hagen, M., Wikelski, M., and Kissling, W. (2011). Space use of bumblebees (bombus spp.) revealed by radio-tracking. *PloS one*, 6:e19997.

Hassenstein, B. and Reichardt, W. (1956). Systemtheoretische analyse der zeit, reihenfolgen und vorzeichenauswertung bei der bewegungsperzeption des rüsselkäfers chlorophanus. *Zeitschrift für Naturforschung B*, 11(9-10):513 – 524.

Hausen, K. (1984). *The Lobula-Complex of the Fly: Structure, Function and Significance in Visual Behaviour*, pages 523–559. Springer US, Boston, MA.

Heinze, S. and Homberg, U. (2007). Maplike representation of celestial e-vector orientations in the brain of an insect. *Science*, 315(5814):995–997.

Heinze, S. and Homberg, U. (2009). Linking the input to the output: New sets of neurons complement the polarization vision network in the locust central complex. *Journal of Neuroscience*, 29(15):4911–4921.

Heinze, S., Narendra, A., and Cheung, A. (2018). Principles of insect path integration. *Current Biology*, 28(17):R1043 – R1058.

Heinze, S. and Reppert, S. M. (2011). Sun compass integration of skylight cues in migratory monarch butterflies. *Neuron*, 69(2):345 – 358.

Homberg, U., Heinze, S., Pfeiffer, K., Kinoshita, M., and el Jundi, B. (2011). Central neural coding of sky polarization in insects. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 366:680–7.

Honegger, D., Meier, L., Tanskanen, P., and Pollefeys, M. (2013). An open source and open hardware embedded metric optical flow cmos camera for indoor and outdoor applications. In *2013 IEEE International Conference on Robotics and Automation*, pages 1736–1741.

Honkanen, A., Adden, A., da Silva Freitas, J., and Heinze, S. (2019). The insect central complex and the neural basis of navigational strategies. *Journal of Experimental Biology*, 222(Suppl 1).

Horn, B. K. P., editor (1986). *Robot Vision*. MIT Press, Cambridge, MA, USA.

Howard, J. G., Dubs, A., and Payne, R. J. (2005). The dynamics of phototransduction in insects. *Journal of Comparative Physiology A*, 154:707–718.

Hubel, D. H. and Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology*, 148:574–91.

Hung, K.-L. J., Kingston, J. M., Albrecht, M., Holway, D. A., and Kohn, J. R. (2018). The worldwide importance of honey bees as pollinators in natural habitats. *Proceedings of the Royal Society B: Biological Sciences*, 285(1870):20172140.

J., G. J. (1950). *The Perception of the Visual World,*. ed. Carmichael, L. Boston, MA: Houghton Mifflin.

Jafferis, N., Helbling, E., Karpelson, M., and Wood, R. (2019). Untethered flight of an insect-sized flapping-wing microscale aerial vehicle. *Nature*, 570:491–495.

Jakobi, T., Kolomenskiy, D., Ikeda, T., Watkins, S., Fisher, A., Liu, H., and Ravi, S. (2018). Bees with attitude: the effects of directed gusts on flight trajectories. *Biology Open*, 7(10).

Jayatilaka, P., Murray, T., Narendra, A., and Zeil, J. (2018). The choreography of learning walks in the australian jack jumper ant myrmecia croslandi. *Journal of Experimental Biology*, 221(20).

Jo, K., Gupta, M., and Nayar, S. K. (2015). Spedo: 6 dof ego-motion sensor using speckle defocus imaging. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 4319–4327.

Jouir, T., Strydom, R., Stace, T. M., and Srinivasan, M. V. (2018). Vision-only egomotion estimation in 6dof using a sky compass. *Robotica*, 36(10):1571–1589.

Juusola, M., Uusitalo, R. O., and Weckström, M. (1995). Transfer of graded potentials at the photoreceptor-interneuron synapse. *Journal of General Physiology*, 105(1):117–148.

Keshavan, J., Gremillion, G., Alvarez-Escobar, H., and Humbert, J. S. (2015). Autonomous vision-based navigation of a quadrotor in corridor-like environments. *International Journal of Micro Air Vehicles*, 7(2):111–123.

Keshavan, J., Gremillion, G., Escobar-Alvarez, H., and Humbert, J. (2014). A μ analysis-based, controller-synthesis framework for robust bioinspired visual navigation in less-structured environments. *Bioinspiration & Biomimetics*, 9:025011.

Kim, A., Fitzgerald, J., and Maimon, G. (2015). Cellular evidence for efference copy in drosophila visuomotor processing. *Nature neuroscience*, 18.

Kim, I. and Dickinson, M. (2017). Idiothetic path integration in the fruit fly drosophila melanogaster. *Current Biology*, 27.

Kim, S., Cassidy, J. J., Yang, B., Carthew, R. W., and Hilgenfeldt, S. (2016). Hexagonal patterning of the insect compound eye: Facet area variation, defects, and disorder. *Biophysical Journal*, 111(12):2735 – 2746.

Knight, J., Sakhapov, D., Domcsek, N., Dewar, A., Graham, P., Nowotny, T., and Philippides, A. (2019). Insect-inspired visual navigation on-board an autonomous robot: Real-world routes encoded in a single layer network. pages 60–67.

Kodzhabashev, A. and Mangan, M. (2015). Route following without scanning. In Wilson, S. P., Verschure, P. F., Mura, A., and Prescott, T. J., editors, *Biomimetic and Biohybrid Systems*, pages 199–210, Cham. Springer International Publishing.

Koenderink, J. J. and van Doom, A. J. (1987). Facts on optic flow. *Biol. Cybern.*, 56(4):247–254.

Kohn, J. R., Heath, S. L., and Behnia, R. (2018). Eyes matched to the prize: The state of matched filters in insect visual circuits. *Frontiers in Neural Circuits*, 12:26.

Krapp, H. G. and Hengstenberg, R. (1996). Estimation of self-motion by optic flow processing in single visual interneurons. *nature*, 384(December):463–466.

Kukko, A., Kaijaluoto, R., Kaartinen, H., Lehtola, V. V., Jaakkola, A., and Hyyppä, J. (2017). Graph slam correction for single scanner mls forest data under boreal forest canopy. *ISPRS Journal of Photogrammetry and Remote Sensing*, 132:199 – 209.

Labrosse, F. (2006). The visual compass: Performance and limitations of an appearance-based method. *Journal of Field Robotics*, 23(10):913–941.

Lambrinos, D., Möller, R., Labhart, T., Pfeifer, R., and Wehner, R. (2000). A mobile robot employing insect strategies for navigation. *Robotics and Autonomous Systems*, 30(1):39 – 64.

Lambrinos, D., Roggendorf, T., and Pfeifer, R. (1999). Insect strategies of visual homing in mobile robots. *Biorobotics - Methods and Applications*.

Land, M. (1997). Visual acuity in insects. *Annual review of entomology*, 42:147–77.

Land, M., Land, P., Nilsson, D., and Nilsson, P. (2002). *Animal Eyes*. Oxford animal biology series. Oxford University Press.

Le Möel, F. and Wystrach, A. (2020). Opponent processes in visual memories: A model of attraction and repulsion in navigating insects' mushroom bodies. *PLOS Computational Biology*, 16(2):1–24.

Lecoeur, J., Dacke, M., Floreano, D., and Baird, E. (2019). The role of optic flow pooling in insect flight control in cluttered environments. *Scientific Reports*, 9:7707.

Legge, E. L. G., Wystrach, A., Spetch, M. L., and Cheng, K. (2014). Combining sky and earth: desert ants (melophorus bagoti) show weighted integration of celestial and terrestrial cues. *Journal of Experimental Biology*, 217(23):4159–4166.

Linander, N., Baird, E., and Dacke, M. (2015a). Bumblebee flight performance in environments of different proximity. *Journal of Comparative Physiology A*, 202:97–103.

Linander, N., Baird, E., and Dacke, M. (2017). How bumblebees use lateral and ventral optic flow cues for position control in environments of different proximity. *Journal of Comparative Physiology A*, 203.

Linander, N., Dacke, M., and Baird, E. (2015b). Bumblebees measure optic flow for position and speed control flexibly within the frontal visual field. *Journal of Experimental Biology*, 218(7):1051–1059.

Lindauer, M. A. (1960). Time-compensated sun orientation in bees. *Cold Spring Harbor symposia on quantitative biology*, 25:371–7.

Lindsay, T., Sustar, A., and Dickinson, M. (2017). The function and organization of the motor system controlling flight maneuvers in flies. *Current Biology*, 27(3):345 – 358.

Lingenfelter, B., Nag, A., and van Breugel, F. (2020). Flivver: Fly lobula inspired visual velocity estimation & ranging.

Lipp, H.-P., Vyssotski, A., Wolfer, D., Renaudineau, S., Savini, M., Tröster, G., and Dell'Omo, G. (2004a). Pigeon homing along highways and exits. *Current biology : CB*, 14:1239–49.

Lipp, H.-P., Vyssotski, A. L., Wolfer, D. P., Renaudineau, S., Savini, M., Tröster, G., and Dell'Omo, G. (2004b). Pigeon homing along highways and exits. *Current Biology*, 14(14):1239 – 1249.

Loianno, G., Scaramuzza, D., and Kumar, V. (2018). Special issue on high-speed vision-based autonomous navigation of uavs. *Journal of Field Robotics*, 35(1):3–4.

Longden, K. D., Wicklein, M., Hardcastle, B. J., Huston, S. J., and Krapp, H. G. (2017). Spike Burst Coding of Translatory Optic Flow and Depth from Motion in the Fly Visual System. *Current Biology*, 27(21):3225–3236.e3.

Léo, C., Schwarz, S., and Wystrach, A. (2020, July). How oscillations are controlled in navigating ants (poster). Insect Navigation Workshop (Online).

Mallat, S. (2008). *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, Inc., USA, 3rd edition.

Mangan, M. and Webb, B. (2012). Spontaneous formation of multiple routes in individual desert ants (Cataglyphis velox). *Behavioral Ecology*, 23(5):944–954.

Mao, J., Hu, X., Zhang, L., He, X., and Milford, M. (2020). A bio-inspired goal-directed visual navigation model for aerial mobile robots. *Journal of Intelligent & Robotic Systems*.

Menzel, R. (2019). The waggle dance as an intended flight: A cognitive perspective. *Insects*, 10:424.

Menzel, R., Brandt, R., Gumbert, A., Komischke, B., and Kunze, J. (2000). Two spatial memories for honeybee navigation. *Proceedings of the Royal Society B: Biological Sciences*, 267(1447):961–968.

Menzel, R., Fuchs, J., Nadler, L., Weiß, B., Kumbischinski, N., Adebiyi, D., Hartfil, S., and Greggers, U. (2010). Dominance of the odometer over serial landmark learning in honeybee navigation. *Naturwissenschaften*, 97:763–767.

Menzel, R. and Greggers, U. (2013). Guidance by odors in honeybee navigation. *Journal of comparative physiology. A, Neuroethology, sensory, neural, and behavioral physiology*, 199.

Menzel, R., Greggers, U., Smith, A., Berger, S., Brandt, R., Brunke, S., Bundrock, G., Hülse, S., Plümpe, T., Schaupp, F., Schüttler, E., Stach, S., Stindt, J., Stollhoff, N., and Watzl, S. (2005). Honey bees navigate according to a map-like spatial memory. *Proceedings of the National Academy of Sciences*, 102(8):3040–3045.

Menzel, R., Kirbach, A., Haass, W.-D., Fischer, B., Degen, J., Koblofsky, M., Lehmann, K., Reiter, L., Meyer, H., Nguyen, H., Jones, S., Norton, P., and Greggers, U. (2011). A common frame of reference for learned and communicated vectors in honeybee navigation. *Current biology : CB*, 21:645–50.

Menzel, R., Tison, L., Fischer-Nakai, J., Cheeseman, J. F., Balbuena, M. S., Chen, X., Landgraf, T., Petrasch, J., Polster, J., and Greggers, U. (2019). Guidance of navigating honeybees by learned elongated ground structures. *Frontiers in Behavioral Neuroscience*, 12.

Mischiati, M., Lin, H.-T., Herold, P., Imler, E., Olberg, R., and Leonardo, A. (2014). Internal models direct dragonfly interception steering. *Nature*, 517.

Mota, T., Yamagata, N., Giurfa, M., Gronenberg, W., and Sandoz, J.-C. (2011). Neural organization and visual processing in the anterior optic tubercle of the honeybee brain. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 31:11443–56.

Muijres, F. T., Elzinga, M. J., Melis, J. M., and Dickinson, M. H. (2014). Flies evade looming targets by executing rapid visually directed banked turns. *Science*, 344(6180):172–177.

Müller, J., Nawrot, M. P., Menzel, R., and Landgraf, T. (2017). A neural network model for familiarity and context learning during honeybee foraging flights. *Biological Cybernetics*, 112:113–126.

Möller, R. and Vardy, A. (2006). Vardy, a.: Local visual homing by matched-filter descent in image distances. biological cybernetics 95, 413-430. *Biological cybernetics*, 95:413–30.

Narendra, A. (2007). Homing strategies of the australian desert ant melophorus bagoti ii. interaction of the path integrator with visual cue information. *Journal of Experimental Biology*, 210(10):1804–1812.

Olberg, R. (1983). Pheromone-triggered flip-flop interneurons in the ventral nerve cord of the silkworm moth bombyx mori. *Journal of Comparative Physiology*, 152:297–307.

Osborne, J. L., Smith, A., Clark, S. J., Reynolds, D. R., Barron, M. C., Lim, K. S., and Reynolds, A. M. (2013). The ontogeny of bumblebee flight trajectories: From naïve explorers to experienced foragers. *PLOS ONE*, 8(11):1–11.

Pahl, M., Zhu, H., Tautz, J., and Zhang, S. (2011). Large scale homing in honeybees. *PLOS ONE*, 6(5):1–7.

Parsons, M. M., Krapp, H. G., and Laughlin, S. B. (2010). Sensor fusion in identified visual interneurons. *Current Biology*, 20(7):624 – 628.

Paulk, A., Dacks, A., Phillips-Portillo, J., Fellous, J.-M., and Gronenberg, W. (2009). Visual processing in the central bee brain. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 29:9987–99.

Peters, R. S., Krogmann, L., Mayer, C., Donath, A., Gunkel, S., Meusemann, K., Kozlov, A., Podsiadlowski, L., Petersen, M., Lanfear, R., Diez, P. A., Heraty, J., Kjer, K. M., Klopfstein, S., Meier, R., Polidori, C., Schmitt, T., Liu, S., Zhou, X., Wappler, T., Rust, J., Misof, B., and Niehuis, O. (2017). Evolutionary History of the Hymenoptera. *Current Biology*, 27(7):1013–1018.

Philippides, A., Baddeley, B., Cheng, K., and Graham, P. (2011). How might ants use panoramic views for route navigation? *Journal of Experimental Biology*, 214(3):445–451.

Philippides, A., de Ibarra, N. H., Riabinina, O., and Collett, T. S. (2013). Bumblebee calligraphy: the design and control of flight motifs in the learning and return flights of bombus terrestris. *Journal of Experimental Biology*, 216(6):1093–1104.

Philippides, A., Steadman, N., Dewar, A., Walker, C., and Graham, P. (2016). Insect-inspired visual navigation for flying robots. In Lepora, N. F., Mura, A., Mangan, M., Verschure, P. F., Desmulliez, M., and Prescott, T. J., editors, *Biomimetic and Biohybrid Systems*, pages 263–274, Cham. Springer International Publishing.

Pomozi, I., Horváth, G., and Wehner, R. (2001). How the clear-sky angle of polarization pattern continues underneath clouds: full-sky measurements and implications for animal orientation. *Journal of Experimental Biology*, 204(17):2933–2942.

Portelli, G., Ruffier, F., and Franceschini, N. (2010). Honeybees change their height to restore their optic flow. *Journal of comparative physiology. A, Neuroethology, sensory, neural, and behavioral physiology*, 196:307–13.

Portelli, G., Ruffier, F., Roubieu, F. L., and Franceschini, N. (2011a). Honeybees' speed depends on dorsal as well as lateral, ventral and frontal optic flows. *PLOS ONE*, 6(5):1–10.

Portelli, G., Ruffier, F., Roubieu, F. L., and Franceschini, N. (2011b). Honeybees' speed depends on dorsal as well as lateral, ventral and frontal optic flows. *PLOS ONE*, 6(5):1–10.

Portilla, J. and Simoncelli, E. P. (2000). Parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40(1):49–71.

Preece, K. and Beekman, M. (2014). Honeybee waggle dance error: adaption or constraint? unravelling the complex dance language of honeybees. *Animal Behaviour*, 94:19 – 26.

Reid, S. F., Narendra, A., Hemmi, J. M., and Zeil, J. (2011). Polarised skylight and the landmark panorama provide night-active bull ants with compass information during route following. *Journal of Experimental Biology*, 214(3):363–370.

Reinhard, J. and Srinivasan, M. (2009). The role of scents in honey bee foraging and recruitment. *Food Exploitation by Social Insects. Ecological, Behavioral, and Theoretical Approaches*, pages 165–182.

Reynolds, A. M. and Rhodes, C. J. (2009). The lévy flight paradigm: Random search patterns and mechanisms. *Ecology*, 90(4):877–887.

Riley, J. R., Greggers, U., Smith, A. D., Stach, S., Reynolds, D. R., Stollhoff, N., Brandt, R., Schaupp, F., and Menzel, R. (2003). The automatic pilot of honeybees. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 270(1532):2421–2424.

Riley, J. R., Reynolds, D. R., Smith, A. D., Edwards, A. S., Osborne, J. L., Williams, I. H., and Mccartney, H. A. (1999). Compensation for wind drift by bumble-bees. *Nature*, 400:126–126.

Roubieu, F. L., Serres, J., Franceschini, N., Ruffier, F., and Viollet, S. (2012). A fully-autonomous hovercraft inspired by bees: wall following and speed control in straight and tapered corridors. In *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO 2012)*, pages 1311–1318, Guangzhou, China.

Roy Khurana, T. and Sane, S. P. (2016). Airflow and optic flow mediate antennal positioning in flying honeybees. *eLife*, 5:e14449.

Ruffier, F. and Franceschini, N. (2005). Optic flow regulation: the key to aircraft automatic guidance. *Robotics and Autonomous Systems*, 50(4):177 – 194. Biomimetic Robotics.

Sabo, C., Chisholm, R., Petterson, A., and Cope, A. (2017). A lightweight, inexpensive robotic system for insect vision. *Arthropod Structure & Development*, 46(5):689 – 702. From Insects to Robots.

Sabo, C., Cope, A., Gurny, K., Vasilaki, E., and Marshall, J. (2016). Bio-inspired visual navigation for a quadcopter using optic flow. *AIAA SciTech 2016*.

Sampat, M. P., Wang, Z., Gupta, S., Bovik, A. C., and Markey, M. K. (2009). Complex wavelet structural similarity: A new image similarity index. *IEEE Transactions on Image Processing*, 18(11):2385–2401.

Sane, S. P., Dieudonné, A., Willis, M. A., and Daniel, T. L. (2007). Antennal mechanosensors mediate flight control in moths. *Science*, 315(5813):863–866.

Sazbon, D., Rotstein, H., and Rivlin, E. (2004). Finding the focus of expansion and estimating range using optical flow images and a matched filter. *Mach. Vis. Appl.*, 15:229–236.

Seelig, J. D. and Jayaraman, V. (2015). Neural dynamics for landmark orientation and angular path integration. *Nature*, 521(7551):186–191.

Serres, J., Dray, D., Ruffier, F., and Franceschini, N. (2008). A vision-based autopilot for a miniature air vehicle: Joint speed control and lateral obstacle avoidance. *Autonomous Robots*, 25:103–122.

Serres, J. R. and Ruffier, F. (2017). Optic flow-based collision-free strategies: From insects to robots. *Arthropod Structure & Development*, 46(5):703 – 717. From Insects to Robots.

Shyy, W., Aono, H., Kang, C.-k., and Liu, H. (2013). *Introduction*, page 1–41. Cambridge Aerospace Series. Cambridge University Press.

Silies, M., Gohl, D. M., and Clandinin, T. R. (2014). Motion-detecting circuits in flies: Coming into view. *Annual Review of Neuroscience*, 37(1):307–327.

Skelton, P. S., Finn, A., and Brinkworth, R. S. (2019). Consistent estimation of rotational optical flow in real environments using a biologically-inspired vision algorithm on embedded hardware. *Image and Vision Computing*, 92:103814.

Spradbery, J. (1973). *Wasps : an account of the biology and natural history of solitary and social wasps*. Seattle : University of Washington Press.

Srinivasan, M., Lehrer, M., Kirchner, W., and Zhang, S. (1991). Range perception through apparent image speed in freely flying honeybees. *Visual neuroscience*, 6:519–35.

Srinivasan, M., Zhang, S., and Bidwell, N. (1997). Visually mediated odometry in honeybees. *Journal of Experimental Biology*, 200(19):2513–2522.

Srinivasan, M., Zhang, S., Lehrer, M., and Collett, T. (1996). Honeybee navigation en route to the goal: visual flight control and odometry. *Journal of Experimental Biology*, 199(1):237–244.

Srinivasan, M. V. (2011). Visual control of navigation in insects and its relevance for robotics. *Current Opinion in Neurobiology*, 21(4):535 – 543. Sensory and motor systems.

Srinivasan, M. V. (2015). Where paths meet and cross: navigation by path integration in the desert ant and the honeybee. *Journal of comparative physiology. A, Neuroethology, sensory, neural, and behavioral physiology*, 201(6):533—546.

Srinivasan, M. V., Zhang, S., Altwein, M., and Tautz, J. (2000). Honeybee navigation: Nature and calibration of the "odometer". *Science*, 287(5454):851–853.

Stankiewicz, J. (2020). *CAD files for an insect inspired biorobot*.

Stankiewicz, J. and Webb, B. (2020). Using the neural circuit of the insect central complex for path integration on a micro aerial vehicle. In Vouloutsi, V., Mura, A., Tauber, F., Speck, T., Prescott, T. J., and Verschure, P. F. M. J., editors, *Biomimetic and Biohybrid Systems*, pages 325–337, Cham. Springer International Publishing.

Steidle, F., Stürzl, W., and Triebel, R. (2019). Visual-inertial sensor fusion with a bio-inspired polarization compass for navigation of mavs. *11th International Micro Air Vehicle Competition and Conference*.

Stone, T. R., Webb, B., Adden, A., Weddig, N. B., Honkanen, A., Templin, R. M., Wcislo, W. T., Scimeca, L., Warrant, E. J., and Heinze, S. (2017). An anatomically constrained model for path integration in the bee brain. *Current Biology*, 27:3069–3085.e11.

Straw, A. D., Lee, S., and Dickinson, M. H. (2010). Visual control of altitude in flying drosophila. *Current Biology*, 20(17):1550 – 1556.

Strydom, R., Denuelle, A., and Srinivasan, M. (2016). Bio-inspired principles applied to the guidance, navigation and control of uas. *Aerospace*, 3:21.

Stürzl, W., Grixa, I., Mair, E., Narendra, A., and Zeil, J. (2015). Three-dimensional models of natural environments and the mapping of navigational information. *Journal of comparative physiology. A, Neuroethology, sensory, neural, and behavioral physiology*, 201(6):563–84.

Stürzl, W. and Zeil, J. (2007). Depth, contrast and view-based homing in outdoor scenes. *Biological Cybernetics*, 96(5):519–531.

Stürzl, W., Zeil, J., Boeddeker, N., and Hemmi, J. (2016). How wasps acquire and use views for homing. *Current Biology*, 26:470 – 482.

Sun, X., Yue, S., and Mangan, M. (2019). Modelling the insect navigation toolkit: How the mushroom bodies and central complex coordinate guidance strategies. *bioRxiv*.

Tal, E. and Karaman, S. (2018). Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness. *2018 IEEE Conference on Decision and Control (CDC)*, pages 4282–4288.

Tammero, L. F. and Dickinson, M. H. (2002). The influence of visual landscape on the free flight behavior of the fruit fly drosophila melanogaster. *Journal of Experimental Biology*, 205(3):327–343.

Tanner, D. and Visscher, P. (2008). Do honey bees average directions in the waggle dance to determine a flight direction? *Behavioral Ecology and Sociobiology*, 62:1891–1898.

Taylor, G. and Krapp, H. (2007). Sensory systems and flight stability: What do insects measure and why? *Advances in Insect Physiology*, 34:231–316.

Taylor, G. J., Luu, T., Ball, D., and Srinivasan, M. V. (2013). Vision and air flow combine to streamline flying honeybees. *Nature Scientific Reports*, 3.

Taylor, G. J., Tichit, P., Schmidt, M. D., Bodey, A. J., Rau, C., and Baird, E. (2019). Bumblebee visual allometry results in locally improved resolution and globally improved sensitivity. *eLife*, 8:1–32.

Thurrowgood, S., Moore, R. J. D., Soccol, D., Knight, M., and Srinivasan, M. V. (2014). A biologically inspired, vision-based guidance system for automatic landing of a fixed-wing aircraft. *Journal of Field Robotics*, 31(4):699–727.

Tinbergen, N. (1972). *The animal in its world (explorations of an Ethologist)*. Cambridge, Mass. : Harvard University Press, 1972-73.

Towne, W. F., Ritrovato, A. E., Esposto, A., and Brown, D. F. (2017). Honeybees use the skyline in orientation. *Journal of Experimental Biology*, 220(13):2476–2485.

Vanhoutte, E., Ruffier, F., and Serres, J. R. (2017). A honeybee's navigational toolkit on board a bio-inspired micro flying robot.

Vautherin, J., Rutishauser, S., Schneider-Zapp, K., Choi, H., Chovancova, V., Glass, A., and Strecha, C. (2016). Photogrammetric accuracy and modeling of rolling shutter cameras. *ISPRS Journal of Photogrammetry and Remote Sensing*, III-3:139–146.

Viollet, S. and Zeil, J. (2013). Feed-forward and visual feedback control of head roll orientation in wasps (polistes humilis, vespidae, hymenoptera). *Journal of Experimental Biology*, 216(7):1280–1291.

von Frish K. (1967). *The dance language and orientation of bees*. London: Oxford University Press.

Wang, Z. and Bovik, A. C. (2009). Mean Squared Error : Love It or Leave It ? *IEEE Signal Processing Magazine*, 26(January):98–117.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.

Webb, B. (2002a). Can robots make good models of biological behavior? *The Behavioral and brain sciences*, 24:1033–50; discussion 1050.

Webb, B. (2002b). Robots in invertebrate neuroscience. *Nature*, 417:359–63.

Webb, B. (2019). The internal maps of insects. *Journal of Experimental Biology*, 222(Suppl 1).

Webb, B. and Wystrach, A. (2016). Neural mechanisms of insect navigation. *Current Opinion in Insect Science*, 15:27 – 39. Pests and resistance * Behavioural ecology.

Wehner, R. (2009). The architecture of the desert ant's navigational toolkit (hymenoptera: Formicidae). *Myrmecological News*, 12:85–96.

Wehner, R., Michel, B., and Antonsen, P. (1996). Visual navigation in insects: coupling of egocentric and geocentric information. *Journal of Experimental Biology*, 199(1):129–140.

Wehner, R. and Müller, M. (2006). The significance of direct sunlight and polarized skylight in the ant's celestial system of navigation. *Proceedings of the National Academy of Sciences*, 103(33):12575–12579.

Werner, A., Stürzl, W., and Zanker, J. (2016). Object recognition in flight: How do bees distinguish between 3d shapes? *PLOS ONE*, 11(2):1–13.

Williams, B., Klein, G., and Reid, I. (2007). Real-time slam relocalisation. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8.

Woodgate, J., Makinson, J., and Lim, K. (2017). Continuous radar tracking illustrates the development of multi-destination routes of bumblebees. *Scientific Reports*.

Woodgate, J. L., Makinson, J. C., Lim, K. S., Reynolds, A. M., and Chittka, L. (2016). Life-long radar tracking of bumblebees. *PLOS ONE*, 11(8):1–22.

Wystrach, A., Beugnon, G., and Cheng, K. (2011). Landmarks or panoramas: what do navigating ants attend to for guidance? *Frontiers in Zoology*, 8(1):21.

Wystrach, A., Dewar, A., Philippides, A., and Graham, P. (2016a). How do field of view and resolution affect the information content of panoramic scenes for visual navigation? a computational investigation. *Journal of Comparative Physiology A*, 202(2):87–95.

Wystrach, A., Lagogiannis, K., and Webb, B. (2016b). Continuous lateral oscillations as a core mechanism for taxis in drosophila larvae. *eLife*, 5(OCTOBER2016).

Wystrach, A., Mangan, M., and Webb, B. (2015). Optimal cue integration in ants. *Proceedings of the Royal Society B: Biological Sciences*, 282(1816):20151484.

Zeil, J. (1993). Orientation flights of solitary wasps (cerceris; sphecidae; hymenoptera):ii. similarities between orientation and return flights and the use of motion parallax. *Journal of Comparative Physiology*, 172:207–222.

Zeil, J. (2012). Visual homing: an insect perspective. *Current Opinion in Neurobiology*, 22(2):285 – 293. Neuroethology.

Zeil, J., Hofmann, M. I., and Chahl, J. S. (2003). Catchment areas of panoramic snapshots in outdoor scenes. *Journal of the Optical Society of America A*, 20(3):450.

Zeil, J., Narendra, A., and Stürzl, W. (2014). Looking and homing: How displaced ants decide where to go. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1636).