

PAPER • OPEN ACCESS

Flying over uneven moving terrain based on optic-flow cues without any need for reference frames or accelerometers

To cite this article: Fabien Expert and Franck Ruffier 2015 *Bioinspir. Biomim.* **10** 026003

View the [article online](#) for updates and enhancements.

You may also like

- [Analysis of a bending-torsion coupled actuator for a smart rotor with active blade tips](#)
Andreas P F Bernhard and Inderjit Chopra
- [Blade-vortex interaction noise reduction with active twist smart rotor technology](#)
Peter C Chen, James D Baeder, Robert A D Evans et al.
- [Nature-inspired in-flight foldable rotorcraft](#)
Hitesh Bhardwaj, Xinyu Cai, Luke Soe Thura Win et al.

Bioinspiration & Biomimetics



PAPER

OPEN ACCESS

RECEIVED
24 July 2014

REVISED
11 December 2014

ACCEPTED FOR PUBLICATION
17 December 2014

PUBLISHED
26 February 2015

Content from this work
may be used under the
terms of the [Creative
Commons Attribution 3.0
licence](#).

Any further distribution of
this work must maintain
attribution to the author
(s) and the title of the
work, journal citation and
DOI.



Flying over uneven moving terrain based on optic-flow cues without any need for reference frames or accelerometers

Fabien Expert and Franck Ruffier

Aix Marseille University, CNRS, ISM UMR 7287, F-13288, Marseille, France

E-mail: fabien.expert@univ-amu.fr and franck.ruffier@univ-amu.fr

Keywords: panoramic optic-flow, non-stationary environment, no referenced states, bio-inspired autopilot

Supplementary material for this article is available [online](#)

Abstract

Two bio-inspired guidance principles involving no reference frame are presented here and were implemented in a rotorcraft, which was equipped with panoramic optic flow (OF) sensors but (as in flying insects) no accelerometer. To test these two guidance principles, we built a tethered tandem rotorcraft called BeeRotor (80 grams), which was tested flying along a high-roofed tunnel. The aerial robot adjusts its pitch and hence its speed, hugs the ground and lands safely without any need for an inertial reference frame. The rotorcraft's altitude and forward speed are adjusted via two OF regulators piloting the lift and the pitch angle on the basis of the common-mode and differential rotor speeds, respectively. The robot equipped with two wide-field OF sensors was tested in order to assess the performances of the following two systems of guidance involving no inertial reference frame: (i) a system with a fixed eye orientation based on the curved artificial compound eye (CurvACE) sensor, and (ii) an active system of reorientation based on a quasi-panoramic eye which constantly realigns its gaze, keeping it parallel to the nearest surface followed. Safe automatic terrain following and landing were obtained with CurvACE under dim light to daylight conditions and the active eye-reorientation system over rugged, changing terrain, without any need for an inertial reference frame.

1. Introduction

Miniature insect-scale robots (Ma *et al* 2013), just like micro aerial vehicles (MAVs), have to be able to make their way autonomously through cluttered, highly moving environments, e.g. foliage moving with the wind, and cope with unpredictable events. These challenging tasks may call for novel sensors and novel control methods that differ from those used in conventional approaches, where all the states of the aerial robot are either measured or estimated in the inertial reference frame (Mellinger *et al* 2012, Shen *et al* 2013).

Ethological findings have shown that complex navigation tasks such as terrain following (Kennedy 1951, Kuenen and Baker 1982) and speed control (Preiss and Kramer 1984, Srinivasan *et al* 1996) are performed by flying insects on the basis of optic flow (OF) cues whereas insects' compound eyes have a very poor spatial resolution in comparison with modern high resolution cameras. In particular, recent studies

on insects have shown that the ventral (Baird *et al* 2006, Barron and Srinivasan 2006, Portelli *et al* 2010a) and dorsal (Portelli *et al* 2011) OFs play an important role in altitude control.

In *Dipterans*, the halteres are known to sense the rotation rates around the animal's cardinal axes (Nalbach 1994, Hengstenberg 1998) by measuring coriolis forces. The halteres are therefore sensory modalities technologically similar to MEMS's rate-gyros that sense the angular speed (MEMS stands for microelectromechanical systems). However, such MEMS's rate-gyros can not be used directly to compute the true angular position (the attitude) because of the well-known drift problem associated with the mathematical integration of the noisy raw measurements. Only complex sensors such as a true angular position gyroscope (e.g. bulky fiber optic gyroscopes) or MEMS-based inertial measurement unit (IMU) based on three degrees-of-freedom (DoF) rate-gyros, three DoF accelerometers and possibly three DoF magnetometers incorporating a complex sensor

fusion algorithm can deliver an attitude angular position without any drift. Insect's halteres are therefore unlikely to enable insects to assess the absolute attitude angles accurately enough by a process that is equivalent to a mathematical integration of this noisy angular velocity. In many species of flying insects, the ocelli are thought to measure attitude changes (Wilson 1978, Berry *et al* 2007), to serve as absolute horizon detectors (Stange 1981) and to be involved along with the compound eye in the dorsal-light response, which compensates for attitude changes by triggering fast head and body movements, as found to occur in *Dipterans* (Schuppe and Hengstenberg 1993, Parsons *et al* 2010). In addition, the dorsal rim area of the flying insects' compound eye is known to serve as a natural compass because it is sensitive to patterns of polarized light in the blue sky (Labhart 1980, Labhart and Meyer 1999). Although ocelli and the dorsal rim may help flying insects to assess their orientation (their attitude) in the inertial frame of reference outdoors, insects are also able to fly in indoor environments where no horizon can be detected and the light polarization is not very likely to be used to determine their direction relative to the Sun. In addition, as far as we know, no organs serving as accelerometers have ever been found to exist in insects which might enable them to sense the gravity vector during flight and to gauge their own body orientations in the inertial reference frame except in dragonflies in which the mass of the head might be sufficiently large to detect angular accelerations (Mittelstaedt 1950). Head compensation around the roll axis has been found to occur on a purely visual basis in wasps (Viollet and Zeil 2013) as well as in flies when the halteres have been removed (Hengstenberg 1993).

Several authors inspired by flying insects recently started to use the OF (i) as a means of landing automatically in the case of rotorcraft (Chahl *et al* 2004, Ruffier and Franceschini 2005, Zufferey *et al* 2010, Kendoul *et al* 2010, Herisse *et al* 2010, Moore *et al* 2010, Herisse *et al* 2012, Expert and Ruffier 2012, de Croon *et al* 2013) and that of fixed-wing unmanned aerial vehicles (Green *et al* 2004, Zufferey *et al* 2010), (ii) avoiding obstacles (Iida 2001, Green *et al* 2004, Ruffier and Franceschini 2005, Griffiths *et al* 2006, Zufferey and Floreano 2006, Ruffier and Franceschini 2008, Beyeler *et al* 2009b, Kendoul *et al* 2009a, Moore *et al* 2010), (iii) hovering (Herisse *et al* 2008, Kendoul *et al* 2009b, Briod *et al* 2013), and (iv) following terrain (Ruffier and Franceschini 2005, Garratt and Chahl 2008, Moore *et al* 2010).

In these aerial robots, OF processing methods were used in several ways as a means of:

- estimating the usual states of the system along with other more classical sensors such as IMUs, sonars, GPS, airspeed sensors and/or accelerometers (Garratt and Chahl 2008, Kendoul *et al* 2009b, Bristeau *et al* 2011, Garcia-Carrillo *et al* 2012) even though recent studies have shown that attitude

estimation (Shabayek *et al* 2012) or 3D localization and navigation can be performed based on vision only (Soundararaj *et al* 2009);

- extracting relative-state information for navigation purposes using the wide field integration method presented by (Conroy *et al* 2009, Hyslop and Humbert 2010), or
- using the OF directly in a control loop without any need for information about the velocity, acceleration, altitude or even about the characteristics of the terrain. Some OF based aerial robots are able to perform tasks such as taking off, terrain-following and landing safely and efficiently by mimicking insects' behavior (Ruffier and Franceschini 2004, 2005, Hérissé *et al* 2010, Moore *et al* 2010), avoiding frontal obstacles (Griffiths *et al* 2006), as well as hovering and landing on a moving platform (Herisse *et al* 2012, Ruffier and Franceschini 2013).

However, in all these robotic studies involving the use of OF, the inputs used by the autopilots of rotary-winged robots were always referred to the inertial frame provided by either an IMU (Hérissé *et al* 2010, Herisse *et al* 2012, Strydom *et al* 2014), a barometric altimeter (Hérissé *et al* 2010) or an external actuator placed on a tether (Ruffier and Franceschini 2004, 2005, 2013). In some studies, fixed-wing robots did not have to use any inertial frame of reference Barrows and (Neely 2000, Green *et al* 2004, Zufferey and Floreano 2006, Beyeler *et al* 2009a, 2009b) because fixed-wing robots are naturally more stable than rotorcraft: the use of a rate gyro in an inner loop is still compulsory to stabilize the roll and pitch flight dynamics of most rotary-wings based robots. Recently, Srinivasan's lab started to work on freely-flying fixed-wing aircraft based on various visual clues. Thanks to OF, they were able to perform terrain following, obstacle avoidance, and landing (Moore *et al* 2010). Using vision only, they were also able to estimate the UAV's attitude thanks to the visual horizon (Thurrowgood *et al* 2009) and use it to determine the heading direction of the aircraft (Moore *et al* 2011). Recently, they showed that they could also use vision to determine wind strength and direction (Moore *et al* 2012).

In previous studies conducted at our laboratory, (Ruffier and Franceschini 2003, 2004, 2005, Expert and Ruffier 2012, Ruffier and Franceschini 2013), a tethered monorotor helicopter (called OCTAVE) performed various OF-based flight maneuvers using a vertical reference frame: the robot's body pitch relative to the absolute vertical was controlled using an external servomotor. Then, (Serres *et al* 2008 and Roubieu *et al* 2012b), showed that it was possible for a grounded robot to travel along a corridor and automatically adjust its forward speed in proportion to the size of the corridor by regulating the OF generated by the nearest

lateral wall, while simultaneously regulating the sum of the two lateral OFs. Inspired by these studies, we previously reported (Expert and Ruffier 2012) that the sum of the ventral and dorsal OFs can be used to automatically adjust a robot's absolute pitch angle via an external servomotor, and hence its speed in the presence of another feedback loop adjusting the altitude. The aerial robot thus equipped was able to fly automatically and maintain a safe altitude proportional to its forward speed that was itself adjusted with respect to the local cluttering criterion measured by the sum of the ventral and dorsal OFs (Expert and Ruffier 2012).

In the present study, for the first time to our knowledge, a rotorcraft's altitude, pitch and forward speed were adjusted using only visual information, a rate gyro and an airspeed sensor without requiring any referenced state vector describing the absolute speed, position, altitude and attitude or any measurements in the inertial frame of reference relative to the vertical like the ones given by an accelerometer. Although accelerometers are nowadays commonly used on flying robots as they are cheap and small, they do not appear suitable for the lightest of robots such as the flapping-wing robots recently developed by Ma *et al* (2013). Indeed, because of the oscillation due to the flapping wings, the signals of accelerometers would not be usable. Besides, the size of even the smallest three-axis accelerometer package (e.g. 3 mm × 3 mm × 1 for ST-Microelectronics LIS331DLH LGA type package) makes accelerometers not suitable for flying robot with a width of 3.5 mm like the Robobee body (see supplementary data of Ma *et al* 2013) whereas really small OF sensors (0.033 grams) have already been embedded on such small robot (Duhamel *et al* 2013). As the present OF-based guidance principles do not involve any onboard or offboard accelerometers, attitude estimation systems or inertial reference frames, the aerial robot relies solely on the ventral and dorsal OFs to avoid obstacles (all along the manuscript, what we will call obstacles correspond to drastic changes of the distance to the surface the robot overlies), and adjusts its speed depending on the local height of the unknown high-roofed tunnel. Such strategies were indeed observed in honeybees (Portelli *et al* 2010a, 2011). In our last experiments, we have also shown that we could still perform terrain-following and landing even in the absence of the airspeed sensor.

Both of these guidance principles enabled the BeeRotor robot to orient its OF sensors without an inertial reference frame, as follows:

- The BeeRotor I autopilot: the eye has a fixed pitch orientation with respect to the aerial robot body's pitch. The eye's pitch therefore reflects any change in the body's pitch due to the changes in the differential thrust of the rotors induced by changes in the sum of the ventral and dorsal OFs.
 - The BeeRotor II autopilot: the eye is decoupled from the robot's body, like in the head of flies, which is actuated by no less than 21 pairs of muscles (Strausfeld *et al* 1987). This decoupling serves to ensure that the artificial eye is always oriented in parallel with the nearest surface so that the OF is computed in the frame of reference defined by the slope of the local environment which enhances the performances of the BeeRotor robot. Although little is known about the reorientation of the head relative to the body along the pitch axis (van Hateren and Schilstra 1999), various processes (yaw optomotor response, vestibulo-ocular reflexes, saccades...) seem to allow the insects to alternate between straight flight paths where they can extract translational OF and fast rotations during which motion vision may not be used. Here we propose a strategy to control the orientation of the eye along the pitch axis based on OF depending on the local orientation of the environment.
- To test these new guidance principles in a naturally contrasting indoor environment, we built a minimalistic, lightweight (80 g) tethered tandem rotorcraft called the BeeRotor robot, which was equipped successively with two different small optical sensors that we called 'BeeRotor's eye' throughout the paper. These two eyes are designed as follow to assess the OF in a wide field of view (FOV):
- Curvace-based full panoramic eye: two semi-cylindrical curved artificial compound eye (CurvACE) sensors mounted back to back, giving a FOV of $360^\circ \times 60^\circ$ process the 1D OF from dim light to daylight conditions in four specific regions of interest (RoIs). CurvACE is the first curved artificial compound eye which has been specifically designed to process OF at high temporal resolution and automatically adapt to the background light level Floreano *et al* (2013).
 - Minimalistic quasi-panoramic eye: a quasi-panoramic eye composed of four visual motion sensors (VMSs) measuring one component of the OF vector in a total number of 20 specific directions with great accuracy (Expert and Ruffier 2012).

Section 2 describes the aeromechanical and electronic design of the BeeRotor robot and the 12 m long circular experimental set-up in which the flying robot was tested. The common control feedback loops piloting the altitude and the forward speed on the basis of OF measurements are also presented in this section. In section 3, the BeeRotor I's autopilot is presented, along with an account of the successful performances achieved by the rotorcraft equipped with CurvACE even in a moving environment from dim lighting to daylight conditions based only on measurements performed in the frame of reference associated with the

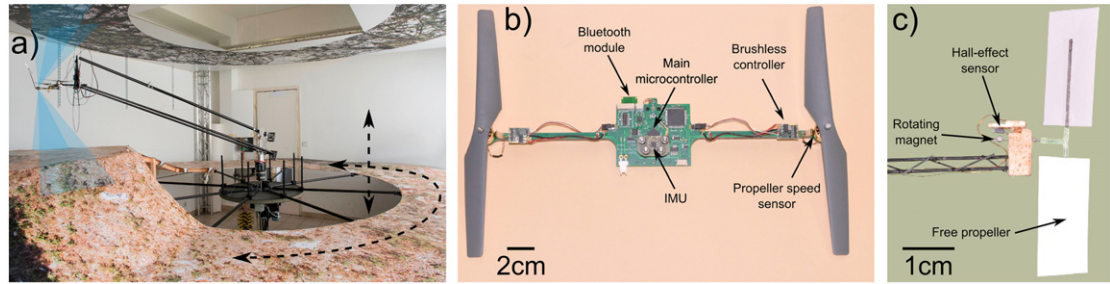


Figure 1. (a) Photograph of the aerial robot in its environment. The robot was made to travel along a 12 m long high-roofed tunnel, the floor and ceiling of which were lined with photographs of natural scenes. The rotorcraft was tethered to the end of a light, counterbalanced whirling pantographic arm, which was driven in elevation and azimuth by the robot's lift and propulsive forces. A 4.5 m long obstacle was added to assess the BeeRotor robot's performances while flying autonomously. This 38 cm high obstacle was composed of a 2.5 m long 9° ascending ramp, a 1 m long flat part and a 1 m long descending ramp with a 21° slope. This artificial ground could be actuated independently via a 30 W brushless dc geared motor and a servomotor which made it rotate in either direction and rise or fall. The fields of view of the four visual motion sensors presented here (blue shaded areas) show that the aerial robot was looking up-forward, up-backward, down-forward and down-backward with a FOV of 24°. The main direction of each visual motion sensor was tilted around the eye's pitch axis at an angle of $\Phi = 23^\circ$. (b) Photograph of the robot's main electronic board (called the body), which is composed of two elongated arms connected to two propellers driven by Brushless outrunner motors. Most of the processing is carried out by the main electronic board, which collects the information originating from the rate gyro of the inertial measurement unit, the custom-made airspeed sensor and the eye. (c) Photograph of a custom-made airspeed sensor fixed to the robot's eye, which weighs less than 0.5 g and measures the airspeed in the $(0.3 \text{ m s}^{-1}; 3 \text{ m s}^{-1})$ range.

robot's body. The performances of the BeeRotor II's autopilot based on the OF measurements performed by its quasi-panoramic eye during autonomous flights are presented in section 4. In particular, section 4 gives further details of the eye-reorientation guidance system aligning the robot's eye with the slope of the nearest neighboring surface. This reorientation strategy improves the performances of the aerial robot, especially in the case of highly variable environments and very steep obstacles (30° slope).

2. The BeeRotor robot: an accelerometer-less aerial robot based mainly on the use of OF cues

Working on the idea that insects' guidance systems may not be based on the knowledge of their body or head attitude, we decided to build a robot devoid of any inertial reference frame, which was capable of navigating on the basis of OF measurements. The rotorcraft had to be tethered to enhance the repeatability of the tests performed and to reduce the risk of crashes while flying without any absolute attitude control. The idea was to equip the BeeRotor robot with a minimal sensor suite (visual sensor, rate gyros and an airspeed sensor) to enable it to fly autonomously on the basis of our innovative guidance principles in an unknown environment without having to estimate its attitude, altitude or ground speed. In particular, neither of the autopilots implemented on the BeeRotor robot refer at any time to the absolute vertical direction (the inertial reference frame), nor are they equipped with accelerometers of any kind.

To navigate autonomously in unknown environments without making any use of the inertial reference frame $(0, x_0, y_0, z_0)$, the BeeRotor robot (see

figure 1(b)) relies mainly on OF measurements computed in four RoIs shown in figure 1(a) (down forward FOV (DF): $270^\circ + \Phi$; down backward FOV (DB): $270^\circ - \Phi$; up forward FOV (UF): $90^\circ - \Phi$ and up backward FOV (UB): $90^\circ + \Phi$), where Φ denotes the specific direction of the FOV perpendicular to the eye's equator: this is the direction in which the OF is measured.

As the BeeRotor robot is equipped with only OF sensors, a rate gyro and an airspeed sensor, thus mimicking respectively the compound eyes, halteres and antennae of flying *Dipterans*, the robot's autopilots receive no information about the robot's absolute pitch angle θ_{pitch} or about the angle of the surfaces α in any frame of reference.

2.1. The BeeRotor robot and its environment

2.1.1. Airframe

The BeeRotor robot is an 80 grams tandem rotorcraft with a 47 cm wingspan which was previously described in Expert and Ruffier (2012), where the pitch angle was controlled via an external servomotor. In the present study, the external servomotor which previously controlled the pitch was removed and the aerial robot was able to rotate freely around its pitch axis thanks to the presence of a low-friction slip-ring assembly. The rotorcraft was composed here of a main electronic board weighing only 13 grams (called the 'body') reinforced by a 25 cm long U-shaped carbon-fiber rod (figure 1(b)). The latest version of the rotorcraft has several DoFs: the aerial robot's altitude depends on the mean speed of the propellers Ω_{Rotors} . The propellers' thrust differential $\Delta\Omega_{\text{Rotors}}$ affects the pitch angle and thus determines the robot's forward speed, and a stepper motor is used to orient the eye with respect to the body in the second version of the BeeRotor robot with a rotating eye.

Details of the procedure used for the identification of the BeeRotor robot and the various transfer functions are given in appendix B and appendix C.

2.1.2. Electronics

The body of the rotorcraft is composed of a custom-designed electronic board, on which all the main sensors and actuators are set around a main dsPIC microcontroller (33FJ128GP804). This main microcontroller includes the autopilot processing the inputs from the VMSs and controlling the robot's actuators. It also communicates with the robot's eye. The main electronic board is also equipped with (see figure 1(b)):

- a tiny Bluetooth module (ALA from Free2move company) mediating information between the robot and a host computer,
- a custom-made positioning sensor based on an A1391 Hall effect sensor placed a few millimeters away from two tiny magnets with opposite polarities measuring the orientation of the eye relative to the body in the $(-20^\circ, 20^\circ)$ range,
- a six axis IMU (MPU 6000 from Invensense) communicating with the main microcontroller via a SPI bus: the IMU was used for monitoring and system identification purposes. During autonomous flight, the aerial robot uses only the angular velocity measured by the rate gyro along the pitch axis. To make the IMU less sensitive to the robot's vibrations during flight, it was placed on top of four rubber bushings (vibration isolators),
- a custom-made airspeed sensor based on a digital Hall effect sensor (A1203) placed in front of a magnet attached to the axis of a freely rotating propeller (this tiny airspeed sensor weighs less than 0.5 grams) (see figure 1(c)),
- two Hall effect sensors placed a few millimeters from the motor shafts containing 16 magnets, measuring the speed of the propellers, which is regulated by a proportional integral (PI) controller, and
- an additional microcontroller for processing the visual data delivered by the CurvACE sensors in order to extract the OF using the interpolation-based time of travel algorithm (Expert *et al* 2012) which is a token-matching scheme (Ullman 1981).

2.1.3. Test-rig

The BeeRotor robot was travelling along a 12 m long high-roofed circular tunnel, the floor and ceiling of which were covered with giant horizontal printed discs (inner diameter 2.4 m, outer diameter 4.5 m) depicting natural scenes (figure 1(a)). In order to assess the performances of the BeeRotor I and II's autopilots, especially their ability to follow a crooked ground, we

placed a 450 cm long, 38 cm high obstacle on the ground. This obstacle was composed of a 9° ascending ramp, a 1 m long flat part and a 21° descending ramp (see figure 1(a)). In some experiments, an even steeper obstacle was obtained by replacing the 1 m long flat part by a 30 cm high supplementary steep obstacle with an ascending and descending ramp with a 31° slope (corresponding to a slope of more than 60%) (see figure 10). In addition, to simulate movements within the environment, the height of the ground of the tunnel could be actively servoed over a 64 cm range using a servomotor (DBL2 from Kollmorgen) connected to a Servostar 300 varying the height of the tunnel between 140 cm and 200 cm. The ground could also be made to rotate in both directions via a 30 W brushless dc geared motor (80 149 606 from Crouzet), thus affecting the perceived OF. The fields of view chosen for both eyes are given in figure 1(a), which shows that the VMSs were always oriented towards the contrasted patterns regardless of the aerial robot's altitude or attitude. The rotorcraft was tethered to the end of a light, counterbalanced whirling pantographic arm driven in elevation and azimuth by the rotorcraft's main force, which provided the lift and the forward propulsive force. The pantographic arm was equipped on the elevation axis with a servo-potentiometer giving the robot's altitude, and on the travel axis, with an optical encoder giving the robot's azimuth angle, and hence the horizontal distance traveled and the robot's forward speed. These data were collected on the host computer via a RT1104 dSpace board and a C# dedicated program monitoring the data generated on the aerial robot. This program also generates the servomotor's control signals and those of the dc motor controlling the height and rotation of the ground. The azimuthal position of the ground is measured via a magnetoresistive sensor (HMC1052 from Honeywell) which is sensitive to the Earth's magnetic field, placed on the inner disc of the artificial ground. These measurements were sent to the dSpace board via a HF transmitter/receiver purchased from Radiometrix. Although the pantographic arm introduced some inertial forces, it enabled us to reliably and reproducibly test the BeeRotor robot's performances under safe flying conditions, while simplifying the parameter monitoring process.

2.2. OF generated by BeeRotor

Let us take a local motion sensor facing an object located at distance $D(\Phi)$ in the direction $d(\Phi)$ characterized by the elevation angle Φ (Φ is the angle between the direction of the local motion sensor $d(\Phi)$ and the perpendicular with the eye's equator (see figure 6). The movements of the local motion sensor can always be decomposed into a translation vector \vec{T} and a rotation vector \vec{R} . The motion field or OF in that direction $\vec{\omega}(\Phi)$ was defined by Koenderink and Doorn (1987) as follows:

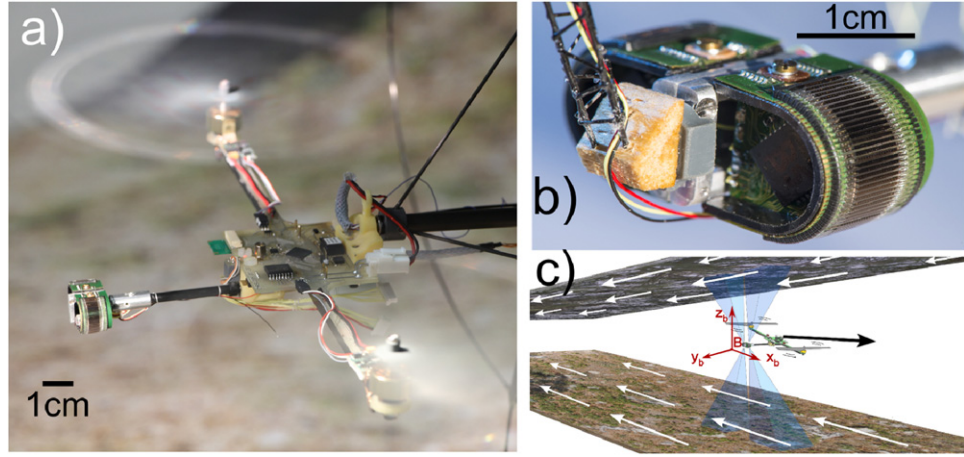


Figure 2. (a) Photograph of the BeeRotor I robot equipped with the full cylindrical CurvACE sensor. The robot's Eye composed of two semi-cylindrical CurvACEs looking forward and backward was placed 7 cm from the body of the BeeRotor I robot to prevent the sensor from being masked by the propellers. (b) Photograph of the full cylindrical CurvACE sensor composed of two semi-cylindrical CurvACEs mounted back to back, giving a field of view of 360°. Among all the pixels, 4 selected RoIs composed of 6×3 pixels sample the visual environment with a FOV of $4 \times 24^\circ$. (c) CAD drawing of the aerial robot equipped with the full cylindrical CurvACE sensor flying along a high-roofed tunnel lined with photographs of natural scenes. The ventral and dorsal optic flows generated by the motion of the aerial robot are sensed by the robot's eye. In this experiment, the robot had no knowledge of any parameters in the inertial frame of reference as all the processing was based on measurements performed in the frame of reference associated with the robot's body (B, x_b, y_b, z_b).

$$\frac{\vec{\omega}(\Phi)}{\omega(\Phi)} = - \frac{\left(\vec{T} - \left(\vec{T} \cdot d(\Phi) \right) d(\Phi) \right)}{D(\Phi)} - \vec{R} \times d(\Phi). \quad (1)$$

If we assume that the BeeRotor robot is moving only in translation ($\vec{R} = 0$), we can obtain from (1):

$$\frac{\vec{\omega}(\Phi)}{\omega(\Phi)} = - \frac{\left(\vec{T} - \left(\vec{T} \cdot d(\Phi) \right) d(\Phi) \right)}{D(\Phi)}. \quad (2)$$

By expressing the translation vector \vec{T} in the orthogonal basis formed by $d(\Phi)$ (the viewing direction) and $\frac{\vec{\omega}(\Phi)}{\|\omega(\Phi)\|}$ (the normalized OF vector), Whiteside and Samuel (1970) established that equation (2) can be expressed in the form:

$$\begin{aligned} \omega(\Phi) &= \frac{\|\vec{V}\|}{D(\Phi)} \sin(90^\circ + \Psi - \Phi) \\ &= \frac{\|\vec{V}\|}{D(\Phi)} \cos(\Phi - \Psi), \end{aligned} \quad (3)$$

where Ψ is the angle between the eye's equator and the direction of the speed vector \vec{V} (see figure 2(c)).

2.3. Dual OF regulation that adjusts altitude and forward speed

The common feedback control loops in BeeRotor I and II can be described as follows:

First OF feedback: the feedback control loop that modifies the altitude was inspired by ethological findings obtained on mosquitoes, desert locusts, moths and honeybees, which showed that flying insects use

OF cues (Kennedy 1951, Kuenen and Baker 1982, Baird *et al* 2006, Portelli *et al* 2010a) to adjust their height by maintaining a preferred retinal velocity.

As we can see from equation (3), the OF measured by a local motion sensor pointing towards the ground depends on the distance between the aerial robot and the ground and on the robot's ground speed. In particular, by fusing the outputs from the two ventral OF sensors, we can deduce from equation (3) that:

$$\begin{aligned} \omega_{\text{MaxOF}} &= \omega(\Phi) + \omega(-\Phi) \\ &= \frac{2 \|\vec{V}\| \cdot \cos(\Phi) \cdot \cos(\Psi)}{D(\Phi)}, \end{aligned} \quad (4)$$

where $\omega(\Phi)$ is the OF in the forward FOV and $\omega(-\Phi)$ is the OF in the backward FOV.

Geometrically, it can be proven that the distance $D(\Phi)$ depends on the slope of the surface α , the distance between the robot and the ground in the direction normal to the slope of the ground $D_{\text{Down/slope}}$, the angle $\theta_{\text{Eye/slope}}$ between the eye's equator and the slope of the nearest surface and the orientation of the local motion sensor Φ

$$D(\Phi) = \frac{D_{\text{Down/slope}}}{\cos(\Phi + \theta_{\text{Eye/slope}})}. \quad (5)$$

In this case, equation (4) can be written:

$$\begin{aligned} \omega_{\text{MaxOF}} &= 2 \cdot \cos(\Phi) \cdot \cos(\Phi + \theta_{\text{Eye/slope}}) \\ &\quad \times \frac{\|\vec{V}\| \cdot \cos(\Psi)}{D_{\text{Down/slope}}}. \end{aligned} \quad (6)$$

We can then see that the sum of the OFs can be used to adjust the ratio between the speed V and the

distance between the robot and the ground in the direction normal to the slope of the ground $D_{\text{Down/slope}}$. We obtain exactly the same result if the robot is following the ceiling, and we then compute the sum of the two dorsal OFs.

To adjust the robot's altitude, it is therefore possible to regulate the maximum OF generated by the nearest surface and thus maintain a safe distance from this surface. During our experiments, we actually used the information from the forward OF sensors to act on the altitude of the rotorcraft as it has proven to improve the ability of the autopilot to avoid forthcoming obstacles.

Second OF feedback: the feedback loop controlling the speed was inspired by experimental findings on honeybees travelling along a tapered corridor, where they were found to adjust their forward speed in order to regulate 'the sum of the two opposite OFs in the horizontal or vertical planes' (Srinivasan *et al* 1996, Portelli *et al* 2011).

In the present case, we can write:

$$\begin{aligned}\omega_{\text{SumOF}} &= \omega(\Phi) + \omega(-\Phi) + \omega(180^\circ + \Phi) \\ &\quad + \omega(180^\circ - \Phi) \\ &= 2 \cdot \cos(\Phi) \cdot \cos(\Psi) \cdot \|\vec{V}\| \\ &\quad \times \left[\frac{\cos(\Phi + \theta_{\text{Eye/slopeDown}})}{D_{\text{Down/slope}}} \right. \\ &\quad \left. + \frac{\cos(\Phi + \theta_{\text{Eye/slopeUp}})}{D_{\text{Up/slope}}} \right], \quad (7)\end{aligned}$$

where $D_{\text{Up/slope}}$ is the distance between the robot and the ceiling in the direction normal to the slope of the ceiling, and $D_{\text{Down/slope}}$ is the distance between the robot and the ground in the direction normal to the slope of the ground.

3. BeeRotor I: OF-based altitude and speed control system adjusting the altitude and speed based on the CurvACE sensor with no inertial reference frame

3.1. Full cylindrical CurvACE sensor and its OF processing scheme

The BeeRotor I robot (see figure 2(a)) was equipped with the full cylindrical CurvACE visual sensor presented in figure 2(b), which is composed of two CurvACEs mounted back to back, each of which has a FOV of $180^\circ \times 60^\circ$ and is composed of three planar layers: a microlens array, a neuromorphic photodetector array, and a flexible printed circuit board that are stacked, cut, and curved to produce a mechanically flexible imager (Floreano *et al* 2013). With its 360° FOV, the rotorcraft can assess the angular speed in several RoIs with great sampling accuracy and cope with any increase in the OF due to the presence of obstacles on its path. It can therefore autonomously

adjust its altitude and forward speed so as to achieve a collision-free trajectory, based solely on the measurements performed in the frame of reference associated with the robot's body (B, x_b, y_b, z_b) .

Figure 2(c) shows the BeeRotor I robot flying in the environment described in section 2.1.3. Due to the robot's motion, the contrasted ceiling and floor generate an OF that is measured by the CurvACE sensor. Among all the pixels in the full cylindrical CurvACE, the four RoIs shown in blue in figure 2(c) composed of 6×3 pixels and covering a FOV of $24^\circ \times 12^\circ$ were selected in order to compute the angular speed between adjacent pixels, using 15 two-pixel local motion sensors (LMSs). In each RoI, we then determined the median value of the OF, which was denoted $\omega_{\text{median}}^{\text{DF}}$, $\omega_{\text{median}}^{\text{DB}}$, $\omega_{\text{median}}^{\text{UF}}$ and $\omega_{\text{median}}^{\text{UB}}$. As we have shown in a previous study (Roubieu *et al* 2012a), by taking the median value of the output of several LMSs, we strongly increase the accuracy of the OF measurements and therefore of the visuomotor control loops as it strongly decreases the noise and the unreliable measurements which are unavoidable in natural environments. We have chosen to take four RoIs as it appears that it was the minimal number in order to implement the control loops presented in section 3.2.

3.2. BeeRotor I's visuomotor control loops

As we saw in the section 2.3, the OF measurements delivered by the full cylindrical CurvACE sensor make it possible to:

- adjust the aerial robot's altitude based on the OF generated by the nearest horizontal surface, and
- adjust its forward speed based on the sum of the ventral and dorsal OFs.

The BeeRotor I's autopilot (see figure 3) is therefore composed of two intertwined OF feedback loops that modify: (i) the mean speed of the propellers Ω_{Rotors} (common-mode) and (ii) the differential thrust of the propellers $\Delta\Omega_{\text{Rotors}}$ (differential-mode).

In order to robustly avoid any forthcoming ground obstacles, the feedback control loop shown in green (see figure 3) adjusts the mean rotor speed on the basis of the maximum forward OF generated by the ground and the ceiling:

$$\omega_{\text{MaxOF}}^{\text{meas}} = \frac{1}{\cos^2(\Phi)} \cdot \max(\omega_{\text{median}}^{\text{DF}}, \omega_{\text{median}}^{\text{UF}}) \quad (8)$$

and this value is compared with the maximum OF setpoint ω_{setMaxOF} in order to set the rotors' mean speed Ω_{Rotors} , which will eventually determine $\frac{D_{\text{Down/slope}} D_{\text{Up/slope}}}{\cos(\Phi + \theta_{\text{Eye/slope}})}$ (see equation (6)), where $D_{\text{Down/slope}}$ is the distance between the aerial robot and the ground in the direction normal to the slope of the ground and $D_{\text{Up/slope}}$ is the distance between the aerial robot and the ceiling in the direction normal to the slope of the ceiling. When the OF generated by the nearest

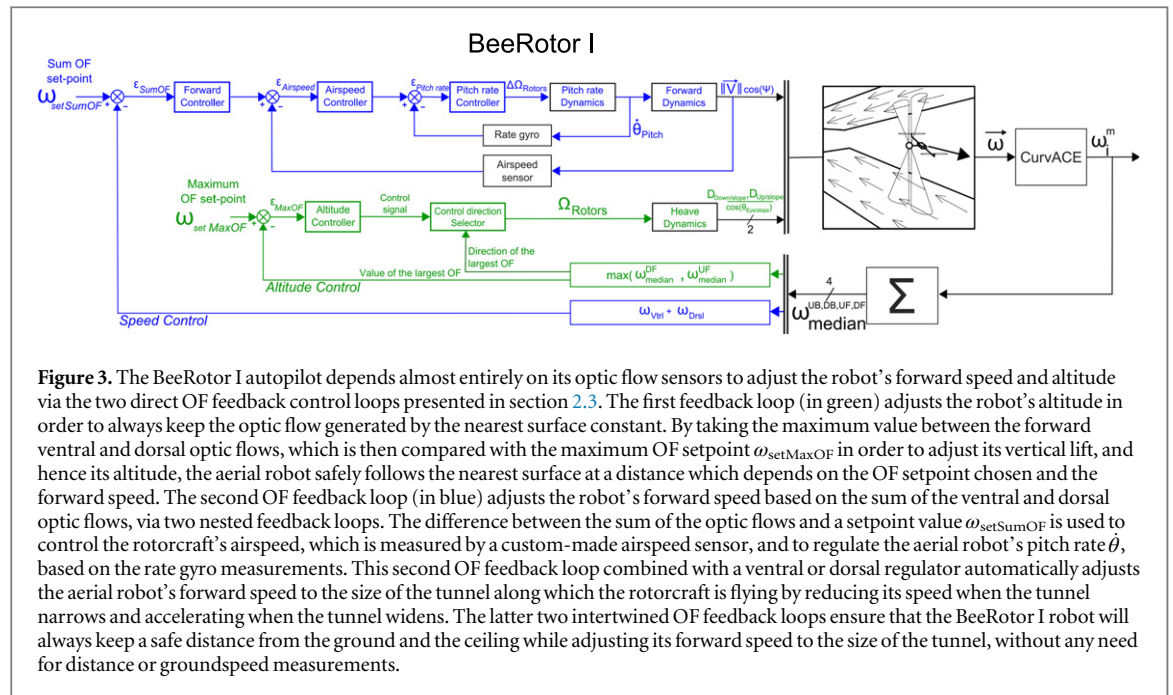


Figure 3. The BeeRotor I autopilot depends almost entirely on its optic flow sensors to adjust the robot's forward speed and altitude via the two direct OF feedback control loops presented in section 2.3. The first feedback loop (in green) adjusts the robot's altitude in order to always keep the optic flow generated by the nearest surface constant. By taking the maximum value between the forward ventral and dorsal optic flows, which is then compared with the maximum OF setpoint ω_{setMaxOF} in order to adjust its vertical lift, and hence its altitude, the aerial robot safely follows the nearest surface at a distance which depends on the OF setpoint chosen and the forward speed. The second OF feedback loop (in blue) adjusts the robot's forward speed based on the sum of the ventral and dorsal optic flows, via two nested feedback loops. The difference between the sum of the optic flows and a setpoint value ω_{setSumOF} is used to control the rotorcraft's airspeed, which is measured by a custom-made airspeed sensor, and to regulate the aerial robot's pitch rate $\dot{\theta}$, based on the rate gyro measurements. This second OF feedback loop combined with a ventral or dorsal regulator automatically adjusts the aerial robot's forward speed to the size of the tunnel along which the rotorcraft is flying by reducing its speed when the tunnel narrows and accelerating when the tunnel widens. The latter two intertwined OF feedback loops ensure that the BeeRotor I robot will always keep a safe distance from the ground and the ceiling while adjusting its forward speed to the size of the tunnel, without any need for distance or groundspeed measurements.

horizontal surface is greater than the maximum OF setpoint ω_{setMaxOF} , depending on which surface is detected, the altitude controller will either increase or decrease the rotor's speed command in order to drive the robot away from this surface.

The median value measured by each VMS was used to compute the ventral and dorsal OFs, which are denoted ω_{VtrI} and ω_{DrsI} :

$$\omega_{\text{Vtrl}}^{\text{meas}} = \frac{1}{2 \cdot \cos^2(\Phi)} \cdot (\omega_{\text{median}}^{\Phi} + \omega_{\text{median}}^{-\Phi}), \quad (9)$$

$$\omega_{\text{Drsl}}^{\text{meas}} = \frac{1}{2 \cdot \cos^2(\Phi)} \cdot \left(\omega_{\text{median}}^{180^\circ + \Phi} + \omega_{\text{median}}^{180^\circ - \Phi} \right), \quad (10)$$

where $\Phi = 23^\circ$ is the angle corresponding to the median direction of each RoI.

The second OF feedback control loop determining the pitch and eventually the forward speed (shown in blue) regulates the following OF sum:

$$\omega_{\text{SumOF}}^{\text{meas}} = \omega_{\text{Vtrl}}^{\text{meas}} + \omega_{\text{Drsl}}^{\text{meas}}. \quad (11)$$

The $\omega_{\text{SumOF}}^{\text{meas}}$ is compared with the sum OF setpoint ω_{setSumOF} in order to adjust the robot's airdspeed.

Two nested feedback loops regulate the pitch rate of the aerial robot $\dot{\theta}$ via a proportional integral derivative (PID) controller based on the rate gyro measurements and the airspeed via a PD controller based on the measurements delivered by the airspeed sensor. The second OF feedback control loop then modifies the differential thrust of the propellers of the aerial robot $\Delta\Omega_{\text{Rotors}}$, which eventually determines the speed $\|\vec{V}\| \cdot \cos(\Psi)$ (see equation (6)) via the forward dynamics. When the sum $\omega_{\text{vtrl}}^{\text{meas}} + \omega_{\text{Dsrfl}}^{\text{meas}}$ is greater than the forward speed OF setpoint ω_{setSumOF} , the forward controller will decrease the airspeed setpoint, which is then compared with the current airspeed measurement. This results in a decrease in the setpoint of the

pitch rate feedback loop and the differential thrust of the propellers $\Delta\Omega_{\text{Rotors}}$. In our experiments, in the absence of wind, the measurements of the airspeed sensor is equal to the groundspeed and provides therefore the robot with metric information in the inertial frame of reference. However, we can predict that, in the case of tailwind, the measured airspeed would be greater than the groundspeed which would result in a transient slowing down of the robot that would be quickly compensated by the *2nd OF feedback loop* adjusting the forward speed of the rotorcraft. In the case of headwind, the robot would first transiently accelerates (inner loop) before reaching back the speed required on steady state thanks to the same *2nd OF feedback loop*. The autopilot therefore does not require any metric information about its speed, altitude or attitude in the inertial frame of reference.

Thanks to the coupling of these feedback loops using respectively the maximum OF generated by the closest surface and the sum of the ventral and dorsal OFs, the forward speed of the robot will be automatically proportionnal to the size of the tunnel (see equation (15) in Serres *et al* 2008).

It is worth noting that this autopilot, which works mainly on the basis of OF measurements, enables an aerial robot to fly autonomously without having to estimate its attitude (yaw, pitch and roll), altitude or groundspeed.

The controllers implemented onboard BeeRotor I are described in detail in appendix D.

3.3. Flight tests on BeeRotor I over uneven terrain

As described in section 2.1.3, the floor of the experimental tunnel could be either rotated or moved up and down by means of two actuators in order to test the BeeRotor robot's performances in a highly variable

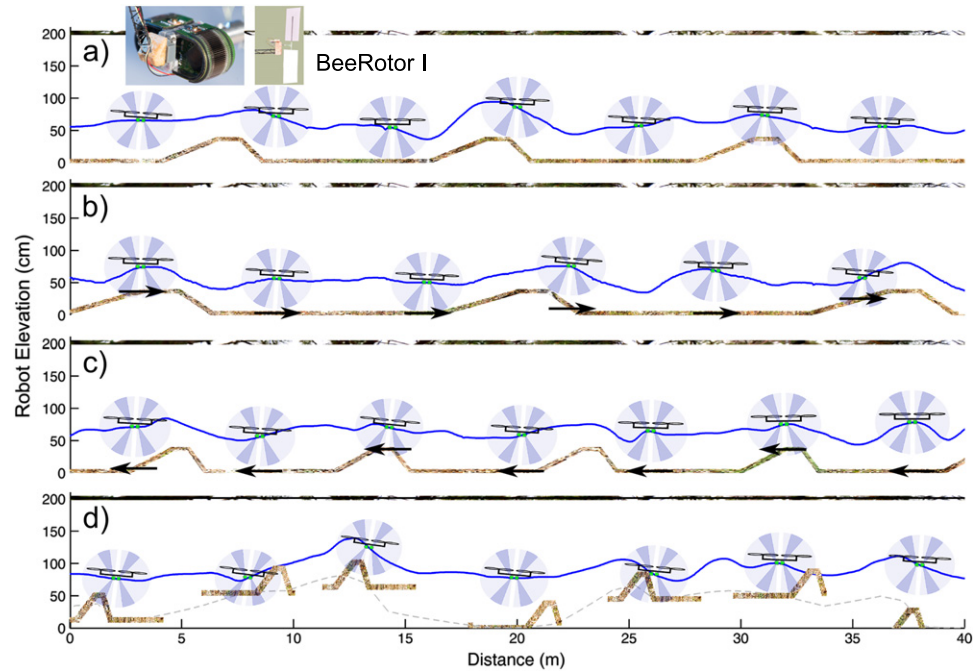


Figure 4. BeeRotor I automatic ground following based on the OF with a moving floor. The aerial robot oriented depending on its pitch angle is shown here every 6 s with the fields of view of the four RoIs superimposed on the field of view of the full cylindrical CurvACE. (a)–(c)–(d) Each curve shows the aerial robot's altitude with $\omega_{\text{setMaxOF}} = 200^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^\circ \text{ s}^{-1}$ above a fixed, a backward- moving and an up-and-down oscillating floor. Despite the disturbances imposed on the ventral optic flow due to the movement of the artificial ground, the aerial robot always robustly avoided colliding with the obstacle. (b) Altitude of the aerial robot travelling over an artificial ground moving at 30 cm s^{-1} in the same direction as the robot with $\omega_{\text{setMaxOF}} = 180^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^\circ \text{ s}^{-1}$. Due to the motion of the ground, the perceived optic flow was lower than previously and the maximum OF setpoint was therefore reduced to prevent the aerial robot from crashing into the ground. (c) Altitude of the aerial robot travelling over an artificial ground moving at 30 cm s^{-1} in the opposite direction to the robot. This time, the perceived optic flow was larger than previously due to the motion of the floor, and the aerial robot automatically flew at a higher altitude. (d) Altitude of the aerial robot flying above the ground oscillating up and down to heights ranging between 0 and 64 cm with a frequency of 0.05 Hz. Despite the very strong perturbations imposed on the measured optic flow due to the obstacle of the ground and the oscillations, the aerial robot was still able to fly autonomously and avoid the obstacle while adjusting its forward speed to the constantly changing size of the tunnel. The dotted line gives the altitude of the ground at each of the aerial robot's positions, taking the oscillations of the ground and the obstacle into account.

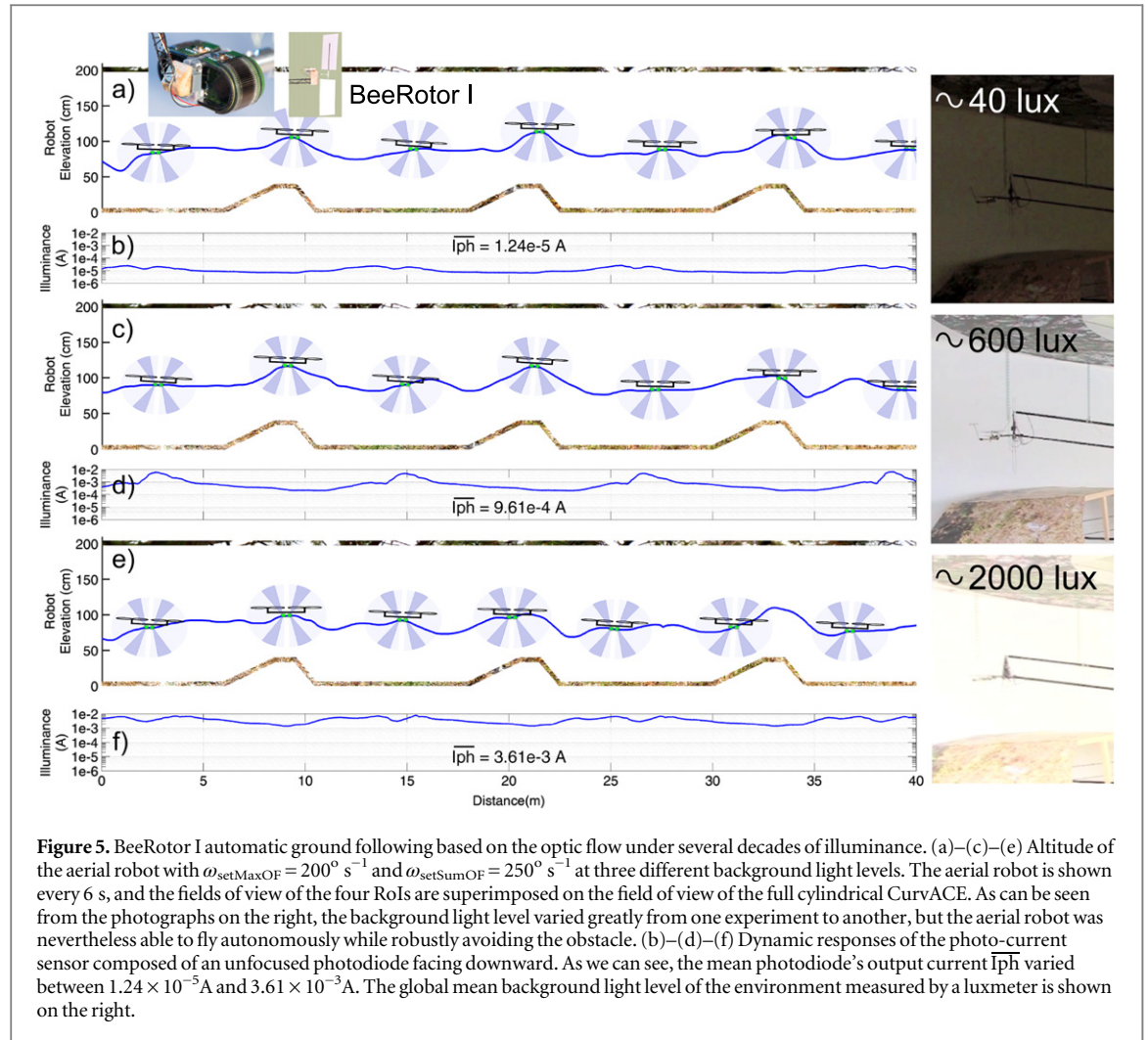
environment. Figures 4(a)–(d) show the BeeRotor I's altitude when flying over a stationary floor, a floor moving at 30 cm s^{-1} in the same direction as the robot, a floor moving at 30 cm s^{-1} in the opposite direction and a floor oscillating up and down (see extension 1). Each curve corresponds to the altitude of the aerial robot during a 40 m long trajectory with $\omega_{\text{setMaxOF}} = 200^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^\circ \text{ s}^{-1}$ except when the aerial robot was rotating in the same direction as the aerial robot (figure 4(b)) where $\omega_{\text{setMaxOF}} = 180^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^\circ \text{ s}^{-1}$. In the latter case, the perceived OF decreased due to the motion of the ground and the altitude feedback loop decreased the mean thrust of the propellers accordingly, and we had to decrease the maximum OF setpoint value to prevent the robot from crashing onto the ground. On the other hand, when the floor was rotating in the opposite direction (figure 4(c)), the perceived OF increased and the aerial robot's altitude increased accordingly so that the OF was kept at the setpoint value similarly to what have been observed on honeybees with lateral moving walls Srinivasan *et al* (1991). Due to the changes in the perceived OF generated by the movement of the ground, the sum of

the ventral and dorsal OFs also varied. This resulted in an increase in the robot's forward speed when the ground was moving in the same direction because the sum of the OFs decreased, and vice versa.

Figure 4(d) gives the altitude of the aerial robot while it was flying over a floor oscillating up and down, reaching heights ranging between 0 and 64 cm with a frequency of 0.05 Hz. Despite this severe disturbance, the aerial robot consistently followed the ground and avoided colliding with the sloping ground. In addition, the robot's forward speed varied constantly due to the changes in the height of the tunnel. In particular, it can be seen that the robot flew at a higher speed when the altitude of the ground was minimal, which explains why the rotorcraft's altitude did not decrease as much as the ground's altitude because the altitude feedback loop adjusted the propellers' speed in order to maintain the ventral OF equal to the maximum OF setpoint value.

3.4. Robustness to illuminance variations

Thanks to its auto-adaptive pixels automatically adapting to background light level, the CurvACE sensor proved to be able to robustly compute the optic



flow under several decades of illuminance Floreano *et al* (2013). Figure 5 shows the trajectories of the BeeRotor I's autopilot while flying autonomously over more than two decades of background light level (photo-current of the photodiode ranging from $1.24 \times 10^{-5} \text{ A}$ to $3.61 \times 10^{-3} \text{ A}$). To measure the effective illuminance of the scene scanned by the CurvACE sensors, a custom-made illuminance sensor based on an OSRAM BPX65 photodiode facing downward was connected to an analog amplifier circuit operating in the photovoltaic mode. The instantaneous photocurrent I_{ph} of this illuminance sensor was determined as follows: $I_{\text{ph}} = (e^{V_{\text{out}}/0.125} - 1) \cdot I_{\text{dark}}$, where the dark current I_{dark} is equal to 0.1 nA and V_{out} is the amplifier's output voltage.

Despite the considerable differences existing from one experiment to another in terms of the background light level, as can be seen from the photographs on the right of the figure 5 and the last experiment in extension 1, the aerial robot was consistently able to automatically adjust its altitude and its forward speed based on the OF and to robustly avoid the obstacle of the ground.

3.5. BeeRotor I summary

The BeeRotor I aerial robot equipped with a full cylindrical CurvACE proved to be able to automatically adjust its altitude and its forward speed based on OF measurements. The performances of the robot were assessed under several decades of background light level in an environment oscillating up and down or moving along the direction of the robot flight or its opposite. The BeeRotor I's autopilot, which uses only measurements performed on the robot's own body frame of reference, manages to navigate robustly in a rugged environment, avoiding all the obstacles encountered.

However, as we can see from equations (6) and (7) and figure 3, this autopilot will adjust not only the speed V and the distance between the aerial robot and the nearest surface $D_{\text{Down,Up/slope}}$, but also other parameters depending on the direction of the speed vector Ψ and the angle between the eye and the slope of the nearest surface $\theta_{\text{Eye/slope}}$. This obviously affects the performances of BeeRotor I's autopilot, which was designed to automatically reach a 'safe height and safe forward speed' based on OF measurements. These quantities that depend on the structure of the

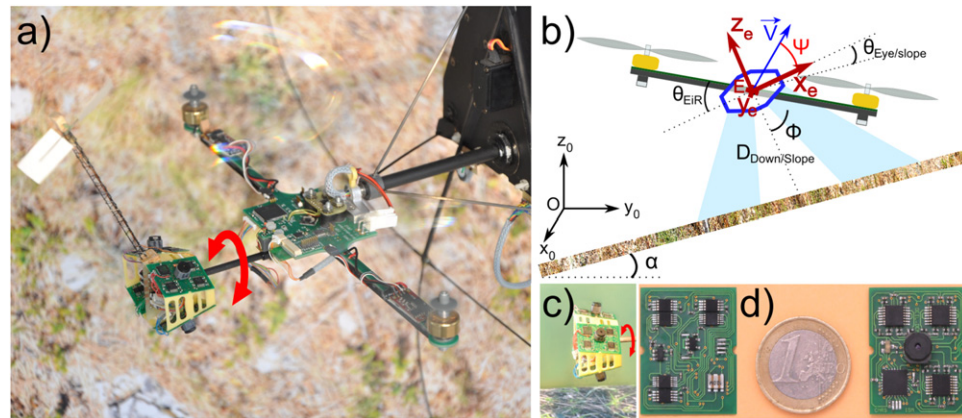


Figure 6. (a) Photograph of the 80 gram BeeRotor II robot. The BeeRotor robot is equipped here with a quasi-panoramic eye decoupled from the body, which is composed of 4 visual motion sensors sampling the visual environment with a $4 \times 24^\circ$ FOV. (b) Drawing of the BeeRotor robot flying over a terrain slanting at an angle α . The angle of the eye relative to the body θ_{EIR} is measured via a magnetic sensor and the angle $\theta_{Eye/Slope}$ between the eye's equator and the slope of the nearest surface is estimated on the basis of the optic flow (see section 4.1 for details of the method), and the result is used to align the eye, keeping it parallel with the terrain. The aerial robot is assumed to be flying at a velocity \vec{V} in the direction defined by the angle Ψ (Ψ is the angle between the direction of the speed vector and the eye's equator). (c) Photograph of the quasi-panoramic eye mounted on the BeeRotor II robot, which constantly realigned itself with respect to the slope of the nearest surface. The orientation of the eye relative to the body can be finely adjusted via a lightweight stepper motor combined with a $\frac{1}{120}$ gear-reducer. (d) Top and bottom view of the electronic board (size: 33×40 mm) of one VMS with its lens mounted on the LSC photosensor array.

unknown environment can be regarded as additional nonlinear disturbances affecting the performances of the BeeRotor I's autopilot. Indeed, when flying over very irregular terrain, the flying robot proved to be unable to avoid very steep obstacles. To improve the proficiency of the aerial robot without adding any sensors giving measurements defined in the inertial frame of reference, we decided to use the OF measurements to visually realign the eye so as to keep it parallel to the slope of the surface followed and thus obtain a more straightforward OF defined in a new frame of reference, i.e. the local slope of the surface followed. One would say that thanks to the panoramic FOV of the full cylindrical CurVACE sensor in the vertical plane, it would have been possible to virtually realign the eye by changing the chosen RoIs. But, due to an inter-ommatidial angle greater than 4° (Floreato *et al* 2013), the artificial compound eye (CURVACE) spatial sampling was not high enough to process the OF in the relevant RoIs. This is the reason why a second quasi-panoramic eye composed of four VMSs oriented precisely thanks to a stepper motor towards specific RoIs was used. By coupling the stepper motor with a $1/120$ reductor, we were able to adjust the orientation of the eye with a $0.02^\circ/\text{steps}$ resolution.

4. BeeRotor II: three direct OF feedback control loops referred to the local slope of the environment

Since the BeeRotor I was not able to avoid obstacles when flying over particularly steep terrain in the absence of an airspeed sensor, or over very irregular terrain, a new eye-reorientation principle was

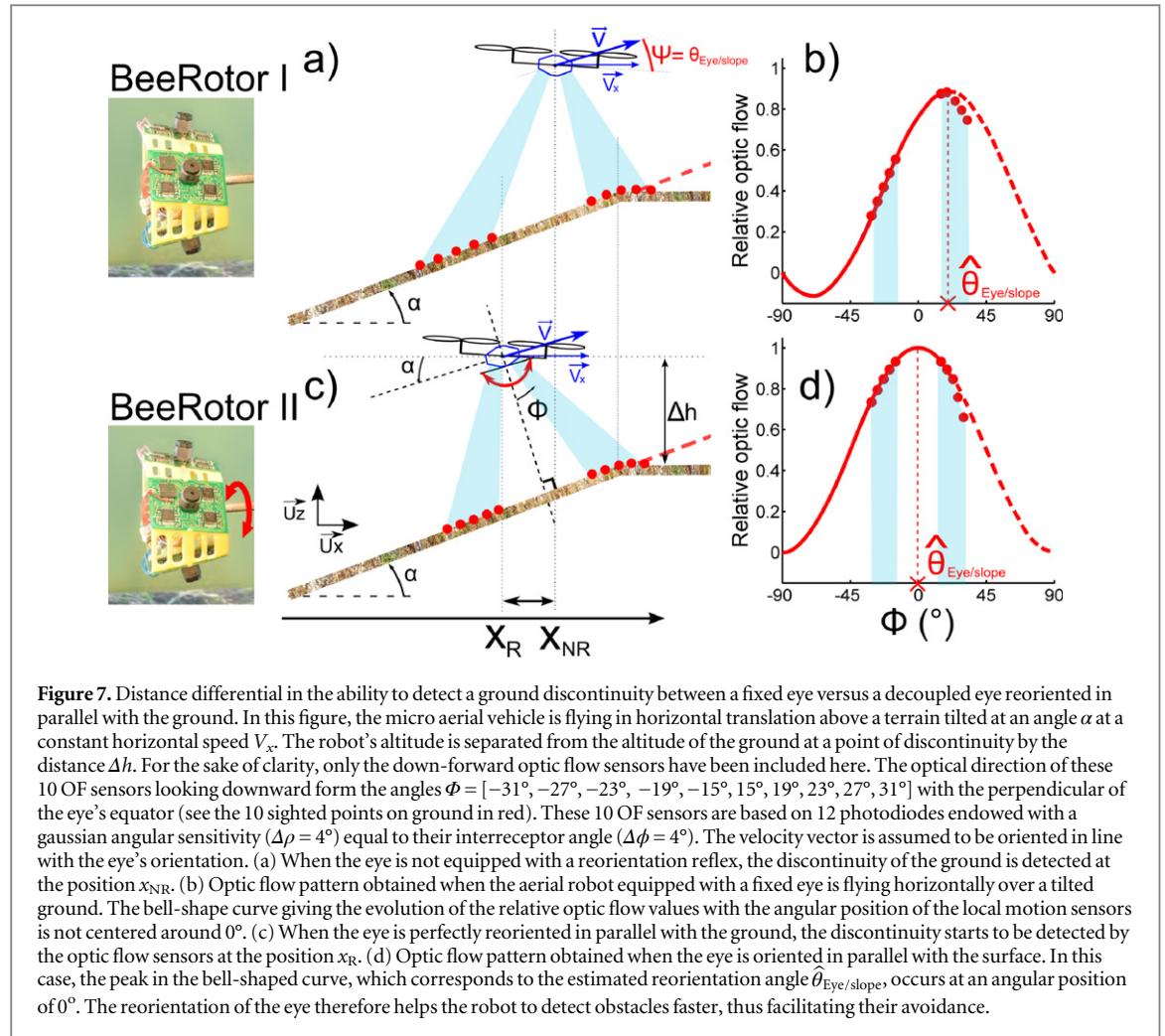
developed to overcome this problem, which is solved every day by flying insects.

To test the validity of the eye-reorientation guidance principle, the BeeRotor II robot (see figure 6) was equipped with a quasi-panoramic eye (see figure 6(c)) consisting of four VMSs, each of which comprised six pixels only and covered a solid angle of 23° . This eye was placed 7 cm from the robot's body to prevent the propellers from entering the visual field of the eye. The robot's miniature quasi-panoramic eye and the OF processing scheme measuring the median OF based on the five measurements delivered by the 2 pixel LMSs have been described in detail in Expert and Ruffier (2012). The robot's eye constantly realigned itself with respect to the slope of the nearest surface (see section 4.1) thanks to the presence of a stepper motor coupled with a gear-reducer giving a resolution of $0.02^\circ/\text{steps}$, which pitches the orientation of the eye up or down with respect to the robot's body. The visual cues (the OF) used by the aerial robot therefore always refer to the slope of the nearest surface and not to the absolute vertical. This eye-reorientation guidance principle enabled to perform all the OF measurements in the new frame of reference associated with the robot's eye, (E, x_e, y_e, z_e) , which is defined by the local slope of the surface followed.

4.1. Eye-reorientation based on the OF: a unique reference frame associated with the aerial robot's eye

4.1.1. Theoretical advantages of reorienting the eye

BeeRotor is assumed here to be flying at a constant horizontal speed V_x above a tilted terrain (with angle α) including a discontinuity (see figures 7(a) and (c)). This aerial robot is equipped with a mechanically



decoupled eye composed of OF sensors looking down forward and down backward (each of them is oriented at an angle Φ relative to the normal of the plane defined by the eye's equator). Depending on the angle of the eye's equator which is zero in the case of a non reoriented (NR) eye and α in the case of a decoupled eye which is perfectly reoriented (R) in parallel with the surface, the ground discontinuity (which occurs at the position where the ground becomes flat) reflected in the OF measurements will be detected respectively at the positions:

$$\begin{aligned} x_{NR} &= \Delta h \times \tan(\Phi) \quad \text{and} \\ x_R &= \Delta h \times \tan(\alpha + \Phi), \end{aligned} \quad (12)$$

where Δh is the difference between the altitude of the robot and the altitude of the ground after the obstacle.

As the BeeRotor is flying at a constant speed V_x , this distance differential will be converted into a prediction time horizon:

$$\begin{aligned} \Delta t &= \frac{x_R - x_{NR}}{V_x} = \frac{\Delta h}{V_x} \\ &\times (\tan(\alpha + \Phi) - \tan(\Phi)). \end{aligned} \quad (13)$$

In a realistic case where the aerial robot is flying over a tilted terrain ($\alpha = 15^\circ$) at the horizontal speed $V_x = 1 \text{ m s}^{-1}$, and the altitude $\Delta h = 1 \text{ m}$ over the ground

discontinuity with a local motion sensor looking in the direction $\Phi = 23^\circ$, the reorientation of the eye will enable the aerial robot to detect the discontinuity 0.35 s earlier in comparison with a fixed eye. In particular, it is worth noting that the reorientation of the eye increases the prediction time horizon of the system by the ratio:

$$\frac{\tau_R}{\tau_{NR}} = \frac{\Delta h}{V_x} \times \frac{\tan(\alpha + \Phi)}{\tan(\Phi)}. \quad (14)$$

In the case of the above example, this ratio is equal to 1.84, which means that the discontinuity will be detected almost twice as fast when the eye is decoupled and oriented parallel to the ground. This example clearly shows that the eye-reorientation system helps to detect obstacles earlier than previously. When following an obstacle with a negative slope, the eye will of course be oriented backwards, thus delaying the detection of any discontinuities. However, we have observed experimentally that although the detection of obstacles is delayed when the aerial robot is following a descending ramp, the reorientation of the eye in parallel with the surface improves the flight performances because the OF will be underestimated in this case with a fixed eye, resulting in a sharp rebound or even a crash at the end of the descending ramp. Thanks to this reorientation, the measured OF does not depend on the slope angle of the closest surface

(see equations (29) and (30)) which has proven, in any case (positive or negative slopes), to improve the performances of the autopilot to avoid obstacles.

4.1.2. OF-based method of estimation of the eye's reorientation angle

As we wanted to use mainly OF on our robot, we tested how its eye could be oriented relative to the surface using only a few LMSs. One simple solution which came to mind was to determine the direction in which the OF was maximal because the distance D would be minimal. In a computer simulations performed, the robot was found to be able to reorient its eye automatically, but it was over-sensitive to OF measurement errors. In particular, as our OF sensor showed a loss of accuracy at high speeds, the reorientation of the eye became noisier when the OF increased.

A particularly robust regression method involving a set of data containing outliers, the 'RANdom SAMple Consensus' algorithm (Fischler and Bolles 1981), might have been suitable here, but it was too costly in terms of computational resources for the BeeRotor's microcontroller because of the large number of iterations necessary to obtain a good fit with the data.

Since the least squares regression method fits the microcontroller's computational abilities, it was implemented here onboard BeeRotor to determine the angle between the slope of the surface and the eye. The OF patterns measured by the ventral OF sensors when there is an angle α between the surface and the eye's orientation and when the eye's equator and the nearest surface are parallel are shown in figures 7(b) and (d), respectively.

In equation (3), it was established that the OF varies with the orientation of each local motion sensor Φ relative to the eye's frame of reference as follows:

$$\omega(\Phi) = \frac{\|\vec{V}\|}{D(\Phi)} \cdot \sin(90 - \Phi - \Psi), \quad (15)$$

where Ψ is the angle between the eye's equator and the direction of the speed vector (see figure 6(b)). If we assume that the local motion sensor is facing downward, we can determine geometrically the distance $D(\Phi)$. Indeed, $D(\Phi)$ depends on the slope of the surface α , the distance between the aerial robot and the ground in the direction normal to the slope of the ground $D_{\text{Down/slope}}$, the angle $\theta_{\text{Eye/slope}}$ between the eye's equator and the slope of the ground and the orientation of the local motion sensor Φ , as follows:

$$D(\Phi) = D_{\text{Down/slope}} \frac{\cos(\alpha)}{\cos(\Phi + \theta_{\text{Eye/slope}})}. \quad (16)$$

From equations (15) and (16), we can deduce that:

$$\omega(\Phi) = \frac{\|\vec{V}\|}{D_{\text{Down/slope}} \cdot \cos(\alpha)} \cdot \cos(\Phi + \Psi) \cdot \cos(\Phi + \theta_{\text{Eye/slope}}). \quad (17)$$

As we were looking for the direction of the maximum OF in order to determine the angle between the eye's equator and the slope of the nearest surface $\theta_{\text{Eye/slope}}$, we differentiated the equation (17) with respect to Φ :

$$\frac{dw(\Phi)}{d\Phi} = \frac{-\|\vec{V}\|}{D_{\text{Down/slope}} \cdot \cos(\alpha)} \cdot \left[\sin(\Phi + \Psi) \cdot \cos(\Phi + \theta_{\text{Eye/slope}}) + \cos(\Phi + \Psi) \cdot \sin(\Phi + \theta_{\text{Eye/slope}}) \right]. \quad (18)$$

The maximum value of the cosine function is then obtained with:

$$\frac{dw(\Phi)}{d\Phi} = 0 \implies \hat{\theta}_{\text{Eye/slope}} = -\frac{\Psi + \theta_{\text{Eye/slope}}}{2}. \quad (19)$$

As we can see from equation (19), when the aerial robot is moving, the maximum value of the OF will not occur when the eye is parallel with the surface followed. But, it will occur and depend on the angle between the plane defined by the eye's equator and the slope of the nearest surface $\theta_{\text{Eye/slope}}$ and the direction of the velocity vector of the aerial robot Ψ in the frame of reference associated with the eye.

Hypothesis: The velocity vector is always parallel to the nearest surface

$$\Psi = \theta_{\text{Eye/slope}}. \quad (20)$$

As explained in section 3.2, the OF generated by the nearest surface was used to always keep a safe distance from this surface therefore leading to a surface following behavior, and the velocity vector will then automatically line up with the nearest horizontal surface.

In this case, equation (19) becomes:

$$\hat{\theta}_{\text{Eye/slope}} = \theta_{\text{Eye/slope}}. \quad (21)$$

The estimated reorientation angle $\hat{\theta}_{\text{Eye/slope}}$ can therefore be used to reorient the eye in order to keep it in parallel with the surface followed, as $\theta_{\text{EiR}} = \theta_{\text{EiR}} + \hat{\theta}_{\text{Eye/slope}}$ results in $\theta_{\text{Eye/slope}} = 0$.

In the neighborhood of the peak in the OF according to equation (17), the OF will vary according to:

$$\omega(\Phi) \simeq f(\Phi') = \frac{\|\vec{V}\| \cos^2(\Phi')}{D_{\text{Down/slope}} \cdot \cos(\alpha)}. \quad (22)$$

The idea was therefore to use the measurements originating from all the LMSs, which are all separated by a given angle, to identify the angle between the eye's equator and the orientation of the surface followed. We had to seek for the coefficients $\frac{\|\vec{V}\|}{D_{\text{Down/slope}} \cdot \cos(\alpha)}$ and $\hat{\theta}_{\text{Eye/slope}}$ in the function $f(\Phi') = \frac{\|\vec{V}\|}{D_{\text{Down/slope}} \cdot \cos(\alpha)} \cdot \cos^2(\Phi' - \hat{\theta}_{\text{Eye/slope}})$ giving the best

approximation in the least squares sense between the OF measurements $\omega(\Phi)$ and the cosine square function. This procedure could be easily implemented in the microcontroller at almost no cost, as the cosine square function could be estimated by a polynomial second order function using a Taylor expansion around zero:

$$\begin{aligned} f(\Phi') &\simeq \frac{\|\vec{V}\|}{D_{\text{Down/slope}} \cdot \cos(\alpha)} \\ &\cdot \left(1 - (\Phi' - \hat{\theta}_{\text{Eye/slope}})^2\right) \\ &\simeq \text{cst.} \left[-\Phi'^2 + 2 \cdot \hat{\theta}_{\text{Eye/slope}} \Phi' \right. \\ &\quad \left. + \left(1 - \hat{\theta}_{\text{Eye/slope}}^2\right) \right], \end{aligned} \quad (23)$$

$$\Leftrightarrow \omega(\Phi) \simeq a \cdot \Phi'^2 + b \cdot \Phi' + c. \quad (24)$$

We define $X = [\Phi'^2, \Phi', 1]$ and from our set of measurements Γ , we determine the coefficients (a , b , c) using the least-squares method:

$$[a, b, c] = \text{inv}(X^*X') * X^* \Gamma'. \quad (25)$$

In the latter expression, only Γ depends on the OF measurements, whereas all the other parameters are constant and depend only on the fixed orientation of each OF measurement in the reference frame of the eye. Finding the coefficients (a , b , c) requires just one matrix multiplication, and equations (23) and (24) can be used to show that the estimated angle between the eye's equator and the slope of the surface followed can be computed as follows:

$$\hat{\theta}_{\text{Eye/slope}} = \frac{-b}{2 \cdot a}. \quad (26)$$

In our computer simulations, we observed that when noise was present in the OF measurements, we could end up with poorly estimated reorientation angles, resulting in the closed loop in oscillations of the eye. To eliminate these errors, we defined the confidence index:

$$\text{Confidence} = \frac{\sum |X' \times b - \omega(\Phi_i)|}{\text{median}(\omega(\Phi_i))}. \quad (27)$$

This coefficient directly reflects how close the OF measurements $\omega(\Phi_i)$ are from describing a cosine square function which is the theoretical evolution of the OF measurement with Φ in an obstacle free environment. The lower the confidence index is, the greater the similarity between the OF measurements and the approximated cosine square function will be. The reorientation angle is therefore only updated when this confidence index is below a fixed threshold value that has been selected empirically. This index strongly increases the robustness of this 3rd feedback loop as the reorientation angle will be only updated with reliable OF values.

Although we reorient the eye parallel to the closest surface, the information we use to act on the altitude of the robot is still the forward OF. Indeed, the main advantage of the eye reorientation is that it allows us to perform naturally all the OF measurements in the frame of reference defined by the slope of the closest surface.

4.2. Measurement of the reorientation angle in the open loop

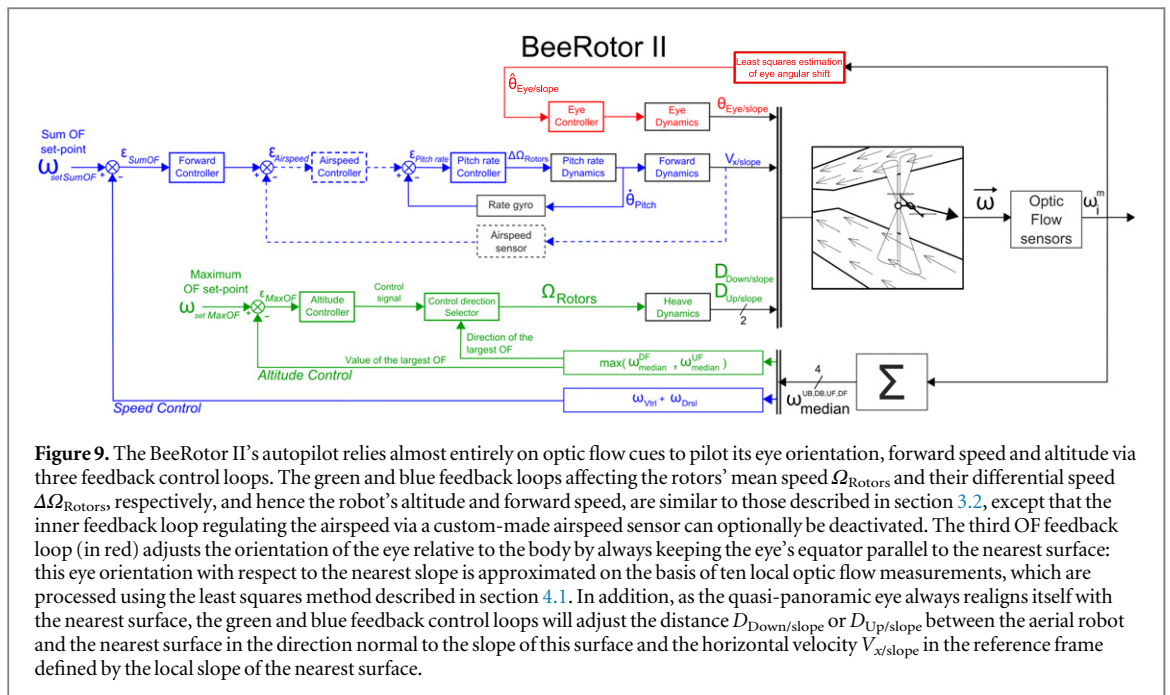
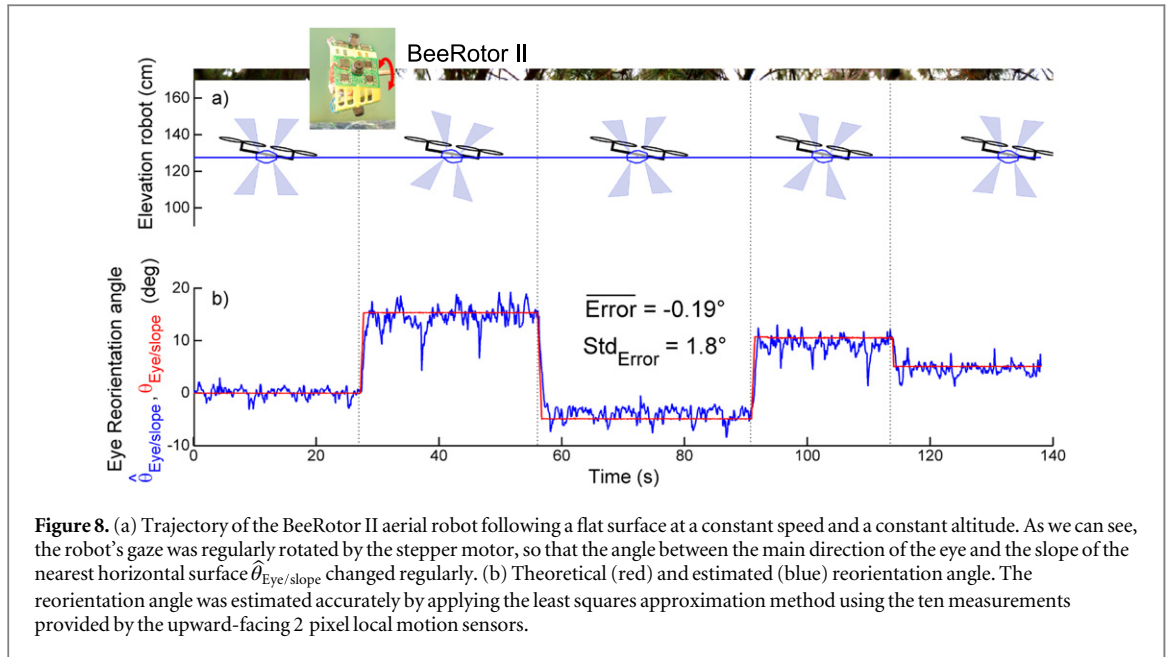
Figure 8 gives the angle of reorientation $\hat{\theta}_{\text{Eye/slope}}$ estimated thanks to the OF measurements in blue superimposed on the theoretical reorientation angle $\theta_{\text{Eye/slope}}$ in red, when the BeeRotor II robot was flying at a constant speed at a constant altitude, and the stepper motor regularly changed the angle of the eye relative to the body θ_{EIR} . The reorientation angle was estimated based on 10 LMSs, as described in section 4.1. Imposing a rotation of α on the eye is equivalent to the situation where the robot is following a surface tilted at this same angle α . As we can see, the accuracy of the estimated reorientation angle was highly satisfactory (the mean error was less than 0.2° and the dispersion was less than 2°), and the response to each of the pitch eye steps was almost instantaneous. As we can see, the reorientation angle was slightly underestimated when the theoretical value departed from zero due to the difference between the cosine square function and the value approximated by a polynomial second order function. However, as the robot uses the measured reorientation angle to constantly align its eye with the closest surface, this underestimation will not unhinge the autopilot as the feedback loop system will always converge in order to make $\theta_{\text{Eye/slope}}$ equals to 0. The same argument can also explain why the autopilot will not be strongly disturbed even if the velocity vector is not perfectly parallel to the closest surface as we hypothesized.

4.3. BeeRotor II's visuomotor control loops

As described above, the aerial robot equipped with suitable means of performing OF measurements defined in the reference frame associated with its eye, can:

- (i) adjust the aerial robot's altitude to regulate (keep constant) the OF generated by the nearest horizontal surface,
- (ii) adjust its forward speed to regulate (keep constant) the sum of the ventral and dorsal OFs,
- (iii) keep the robot's eye's equator parallel to the nearest surface.

The BeeRotor II's autopilot (see figure 9) is therefore composed of three direct feedback control loops determining: (i) the mean speed of the propellers Ω_{Rotors} , (ii) the differential thrust of the propellers



$\Delta\Omega_{\text{Rotors}}$ and (iii) the pitch orientation of the eye $\theta_{\text{Eye/slope}}$.

The first and second OF feedback control loops adjusting the altitude and the speed implemented onboard BeeRotor II, which are shown here in green and blue, respectively, are similar to the feedback loops implemented in the BeeRotor I autopilot described in section 3.2.

Third OF feedback: based on the least squares estimation method presented in section 4.1 giving the angle between the eye's equator and the slope of the nearest surface followed $\hat{\theta}_{\text{Eye/slope}}$ (see equation (21)), the red feedback loop (see figure 9) reorients the eye, keeping it parallel with the nearest surface. This eye

reorientation will generate an additional small rotational OF that is known and subtracted from every LMS measurement.

At the same time, this method of eye reorientation makes it possible to perform all the OF measurements in the reference frame associated with the aerial robot's eye, which therefore depends on the local slope of the tunnel.

If we take the nearest surface to be the ground and assume that the eye is kept parallel with the local ground orientation (in which case, $D_{\text{Down/slope}} = D(\Phi) \cdot \cos(\Phi)$), we can use equation (3) to express the sum of the ventral OFs measured in the forward and backward oriented ROIs as follows:

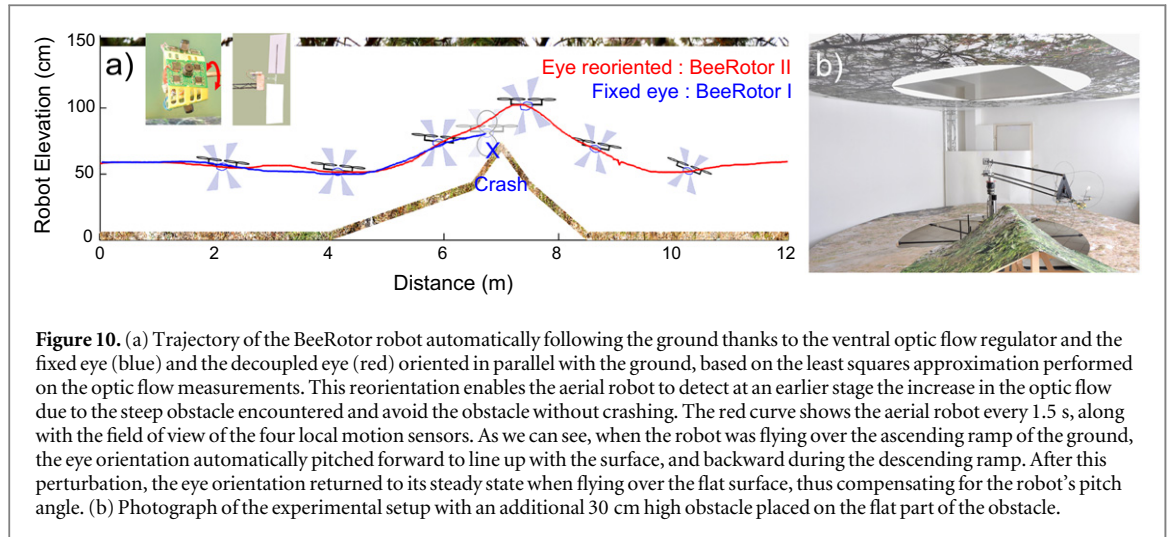


Figure 10. (a) Trajectory of the BeeRotor robot automatically following the ground thanks to the ventral optic flow regulator and the fixed eye (blue) and the decoupled eye (red) oriented in parallel with the ground, based on the least squares approximation performed on the optic flow measurements. This reorientation enables the aerial robot to detect at an earlier stage the increase in the optic flow due to the steep obstacle encountered and avoid the obstacle without crashing. The red curve shows the aerial robot every 1.5 s, along with the field of view of the four local motion sensors. As we can see, when the robot was flying over the ascending ramp of the ground, the eye orientation automatically pitched forward to line up with the surface, and backward during the descending ramp. After this perturbation, the eye orientation returned to its steady state when flying over the flat surface, thus compensating for the robot's pitch angle. (b) Photograph of the experimental setup with an additional 30 cm high obstacle placed on the flat part of the obstacle.

$$\begin{aligned}\omega_{\text{MaxOF}} &= \omega(\Phi) + \omega(-\Phi) \\ &= \frac{2 \|\vec{V}\| \cdot \cos^2(\Phi) \cdot \cos(\Psi)}{D_{\text{Down/slope}}}. \quad (28)\end{aligned}$$

If we decompose the velocity vector \vec{V} into its two components $V_{x/\text{slope}} = V \cdot \cos(\Psi)$ and $V_{z/\text{slope}} = V \cdot \sin(\Psi)$, namely the horizontal and vertical velocity, respectively, we can write equation (28) as follows in the reference frame defined by the local slope of the nearest surface:

$$\begin{aligned}\omega_{\text{MaxOF}} &= \omega(\Phi) + \omega(-\Phi) \\ &= 2 \cdot \cos^2(\Phi) \cdot \frac{V_{x/\text{slope}}}{D_{\text{Down/slope}}}. \quad (29)\end{aligned}$$

In addition, assuming for the sake of simplification that the gaze of the aerial robot is kept parallel with the ground and the ceiling, we can write:

$$\begin{aligned}\omega_{\text{SumOF}} &= \omega(\Phi) + \omega(-\Phi) \\ &\quad + \omega(180^\circ + \Phi) + \omega(180^\circ - \Phi) \\ &= \frac{2V_{x/\text{slope}} \cdot \cos^2(\Phi)}{D_{\text{Down/slope}}} + \frac{2V_{x/\text{slope}} \cdot \cos^2(\Phi)}{D_{\text{Up/slope}}} \\ &= 2V_{x/\text{slope}} \cdot \cos^2(\Phi) \\ &\quad \times \left(\frac{1}{D_{\text{Down/slope}}} + \frac{1}{D_{\text{Up/slope}}} \right). \quad (30)\end{aligned}$$

It is impossible for the eye to be parallel to both surfaces in a corridor with a contrasting obstacle on the ground. But, if the eye is always parallel to the nearest surface, the OF generated by the other surface will be smaller and the error due to this simplification will not affect the result very greatly.

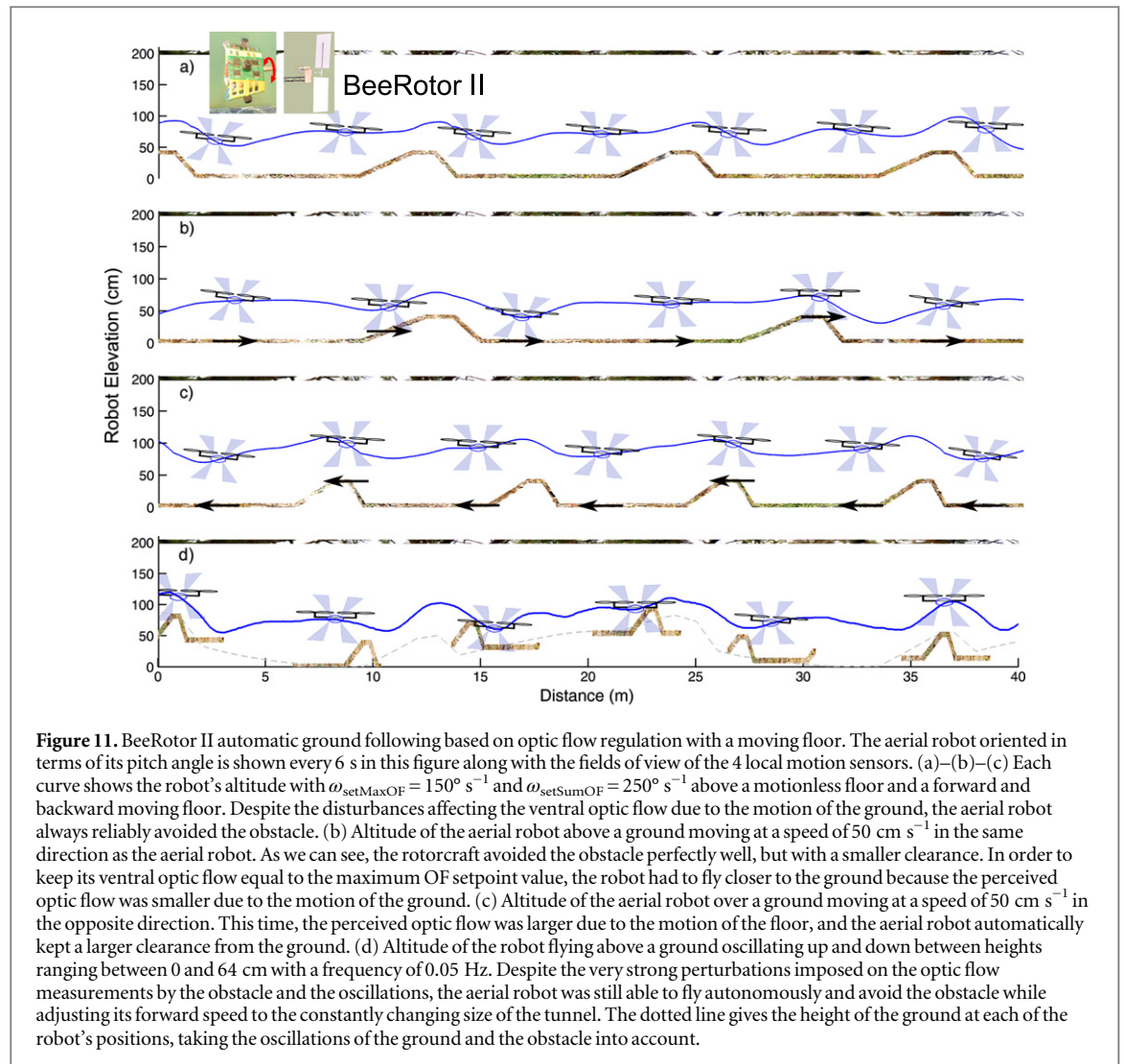
It can be seen from equations (29) and (30) that thanks to the eye-reorientation principle, the first and second OF feedback control loops will adjust the aerial robot's horizontal speed $V_{x/\text{slope}}$ in the frame of reference defined by the local ground and the distance $D_{\text{Down/slope}}$ between the robot and the ground in the direction normal to the ground. The possibility of

detecting the obstacle in advance improves the robustness of the BeeRotor autopilot's performances, and hence its ability to steer a collision-free course in a moving or cluttered environment, as we will see in the following sections.

The controllers implemented onboard BeeRotor are described in detail in appendix D.

4.4. Comparison between BeeRotors I and II: the eye-reorientation principle makes the aerial robot avoid crashes in very steeply sloping terrain

To test the performances of the BeeRotor II's improved autopilot endowed with the novel eye reorientation system, an experiment was performed in which an additional feature was included on the ground of the environment (see extension 2) by replacing the 1 m long flat part of the obstacle by a 30 cm high supplementary steep obstacle (see figure 10(b)). Figure 10(a) shows the trajectory of the aerial robot equipped with a fixed eye versus a decoupled eye which was automatically kept parallel to the ground thanks to the eye-reorientation principle presented in section 4.1. In both cases, the robot's forward speed and altitude were automatically adjusted based on the OF measurements via the first and second OF feedback control loops described in section 4.3. The fixed eye was kept at an angle of 0° relative to the body. As we can see, the reorientation of the eye made it possible for the robot to avoid the additional steep obstacle without crashing because the OF measurements reflected the distance to the ground more accurately. When the aerial robot was flying over the obstacles, we observed that the eye pitched upward in front of the ascending ramp and downward when BeeRotor reached the descending ramp, keeping its body parallel with the surface below. When flying over the flat part of the environment, the eye was oriented at an angle which compensated for the body pitch angle. However, as we can see, the duration of the eye reorientation process was quite long as the eye required several meters to return to the steady state



again. This was mainly due to the large time constant chosen for the low-pass filter applied to the signals driving the stepper motor rotating the eye. When this time constant was reduced, the eye reorientation process was faster, generating a large rotational OF superimposed on the translational OF and disturbing the estimation of the required reorientation angle $\hat{\theta}_{\text{Eye/slope}}$, which often resulted in eye oscillations. With a fixed panoramic eye measuring the OF in all directions, the angle between the eye's equator and the surface could be determined faster, and it could reflect the OF generated by the surface followed more accurately, thus enhancing the robot's ability to follow a terrain at short range while avoiding prominent features.

4.5. BeeRotor II's performances in highly variable environments

Figures 11(a)–(c) show the altitude of the BeeRotor II in the case of a stationary floor, a floor moving at 50 cm s^{-1} in the same direction as the robot and a floor moving at 50 cm s^{-1} in the opposite direction. Each curve gives the robot's altitude during a 40 m long flight

with $\omega_{\text{setMaxOF}} = 150^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^\circ \text{ s}^{-1}$. As we can see, the robot flew at a lower altitude when the floor was rotating in the same direction (figure 11(b)). The OF perceived decreased due to the motion of the ground, and the altitude feedback loop therefore decreased the mean thrust of the propellers in order to compensate for this decrease. On the other hand, when the floor was rotating in the opposite direction (figure 11(c)), the OF perceived was greater and the aerial robot's altitude increased so that the OF was kept at the setpoint value.

The aerial robot is shown here every 6 s during each trajectory. Due to the changes in the OF caused by the movement of the ground, the sum of the ventral and dorsal OFs varied. This led to an increase in the robot's forward speed when the ground was moving in the same direction because the sum of the OFs decreased, and vice versa.

Figure 11(d) shows the altitude of the aerial robot flying over a floor moving up and down: the robot's position and orientation are shown every 6 s with $\omega_{\text{setMaxOF}} = 180^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 220^\circ \text{ s}^{-1}$. The height of the ground was oscillating between 0 and 64 cm at a frequency of 0.05 Hz, which strongly

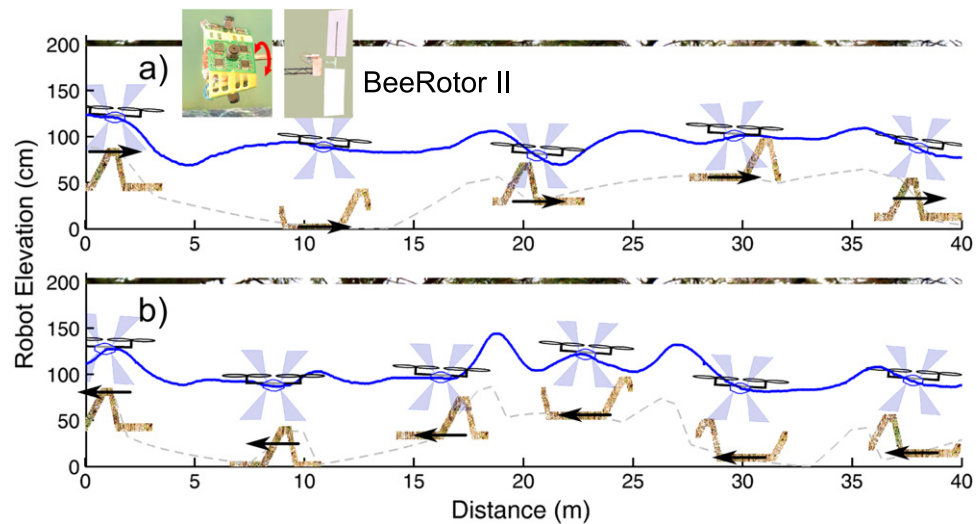


Figure 12. BeeRotor II automatic ground following based on optic flow regulation with a rotating and up and down oscillating floor. Each curve shows the the aerial robot's altitude during a 40 m long flight with respect to the ground below. (a) Altitude of the robot flying above the oscillating ground moving at a speed of 50 cm s^{-1} in the same direction as the robot. Once again, the rotorcraft neatly avoided the obstacle, but at a higher speed and a lower altitude relative to the ground. (b) Altitude of the robot flying above the oscillating ground moving at 50 cm s^{-1} in the opposite direction to that of the aerial robot. This time, the perceived optic flow was greater due to the motion of the floor, and the aerial robot therefore flew automatically at a higher altitude and a lower speed.

disturbed the perceived ventral OF. Despite this considerable disturbance, the aerial robot still followed the ground and avoided the obstacle autonomously. In addition, the robot's forward speed was constantly changing due to the changes in the size of the tunnel. In particular, the aerial robot flew at a greater speed when the ground was minimally elevated, which explains why the rotorcraft's altitude did not decrease as much as the height of the ground: the altitude feedback loop adjusted the propellers' speed in order to maintain the ventral OF equal to the maximum OF setpoint value.

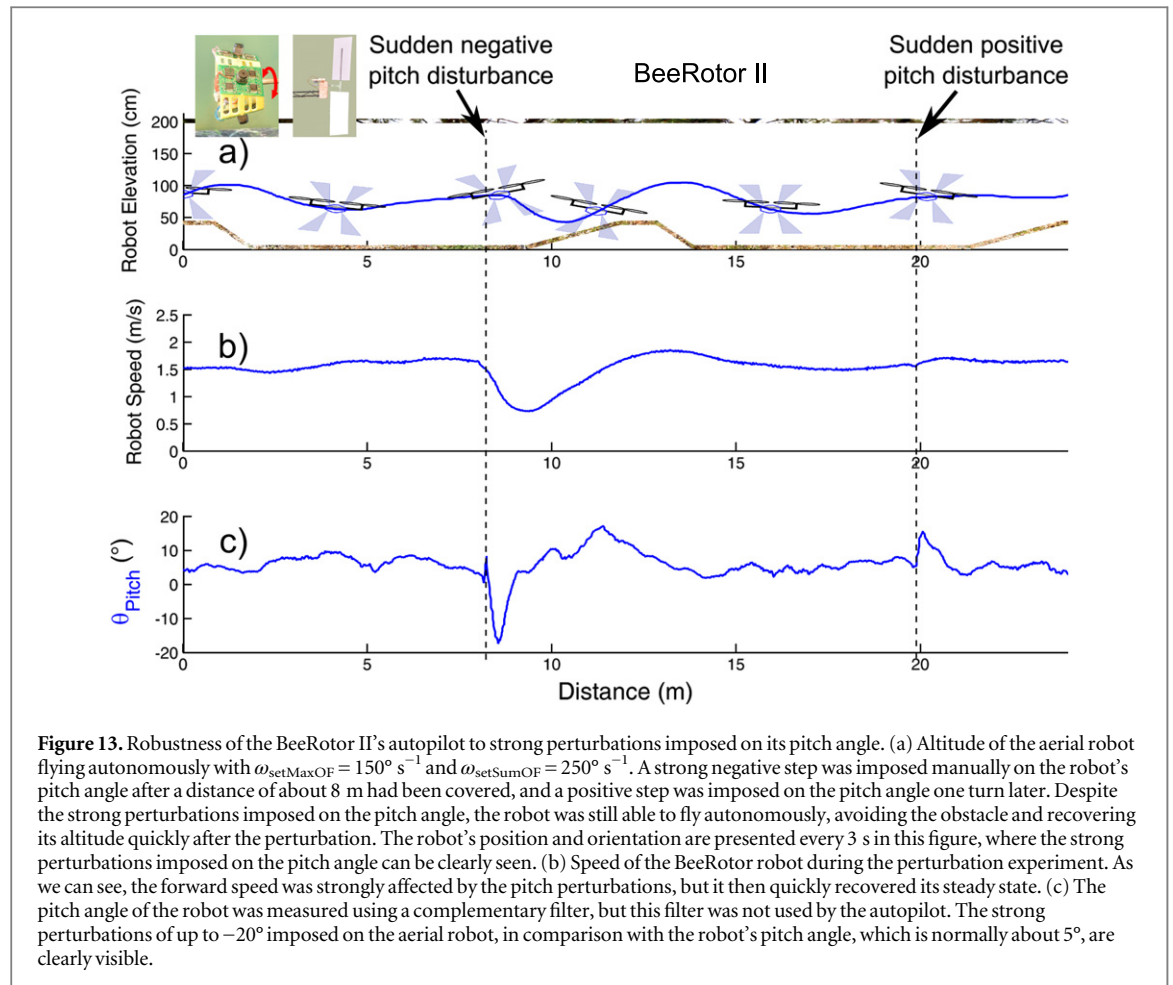
Like the BeeRotor I, the BeeRotor II was able to fly autonomously based on OF measurements and adjust its forward speed and its clearance from the ground and the ceiling without any need for an accelerometer or any measurements in the inertial frame of reference. However, thanks to the addition of the eye-reorientation principle, the BeeRotor II's autopilot can reject stronger perturbations and therefore navigate without crashing, even in a moving environment or in the presence of a sudden rise of the terrain.

Figure 12 shows the altitude of the BeeRotor II in a very unstable environment, where the ground was rotating in the forward or backward direction while its elevation was oscillating (see extension 2). The robot's altitude is plotted in figures 12(a) and (b) when the ground was oscillating up and down and moving at 50 cm s^{-1} in the same direction as the robot with $\omega_{\text{setMaxOF}} = 125^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 280^\circ \text{ s}^{-1}$ and with the ground oscillating up and down and rotating at 50 cm s^{-1} in the opposite direction with $\omega_{\text{setMaxOF}} = 175^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^\circ \text{ s}^{-1}$. The aerial robot was able to consistently adjust its altitude and its forward speed based on the OF measurements and

avoid the obstacle. As observed previously with BeeRotor I, the robot flew at a lower altitude and a higher speed when the ground was rotating in the same direction because the OF perceived in this case was lower. On the other hand, the robot flew at a higher altitude and a lower speed when the ground was rotating in the opposite direction.

4.6. BeeRotor II's rejection of pitch disturbances

To assess the robustness of the autopilot mounted onboard the BeeRotor II robot, strong perturbations were applied manually to the robot's pitch angle while it was automatically following the stationary ground with $\omega_{\text{setMaxOF}} = 150^\circ \text{ s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^\circ \text{ s}^{-1}$ (see extension 2). Figure 13(a) shows the aerial robot's trajectory during a 25 m-long flight, where a strong negative step was imposed on the robot's pitch angle at a distance of about 8 m and a positive step on the pitch angle after it had travelled 20 m. The robot's pitch angle is presented in figure 13(c), and the negative pitch step of almost -20° and the positive pitch step of more than 10° can be clearly distinguished, whereas the robot's pitch angle during autonomous flight is generally about 5° . Although these perturbations strongly affected the rotorcraft's trajectory, it was still able to avoid the sudden rise of the terrain encountered and regain its initial altitude after travelling a few meters. As the pitch angle directly affects the robot's forward speed, the negative step imposed on the pitch angle strongly reduced its forward speed (see figure 13(b)), whereas the positive step was immediately rejected by the feedback control loops and hardly affected the robot's altitude or its forward speed at all.



4.7. BeeRotor II's autopilot performances without any accelerometers or airspeed sensors

Insects' antennae are thought to contribute importantly to controlling their airspeed. Data obtained on free-flying insects deprived of their antennae provided direct evidence that these organs are involved in flight control: a decrease in the forward speed was observed in *Dipterans* deprived of their antennae, although they were still able to fly (Campan 1964, Burkhardt and Gewecke 1965). It has been suggested recently in flies that the position of the antennae is actively controlled and adjusted during flight by the 'antennal positioning reaction' which could allow the animal to measure absolute airspeed when including efference copies sent to the antennae muscles and the signals from the Johnston's organ that measure changes in airspeed (Taylor and Krapp 2007). The airspeed sensor implemented in our robot could then be compared with insects' antennae. Other studies have hypothesized that the 'antennal positioning reaction' enhances the insects' ability to sense changes in the airspeed (Mamiya *et al* 2011). This finding suggests that the antennae are sensitive only to the acceleration of the air (Fuller *et al* 2014), which may be processed in a nested feedback loop. With our custom-made airspeed sensor, we tried to use the changes in airspeed as a control parameter, but the signals turned out to be too

noisy for this parameter to be suitable for use onboard the aerial robot.

However, it seemed to be worth testing the ability of the BeeRotor robot to fly in its environment without the inner feedback loop regulating the airspeed (see figure 9), since the flight performances of insects deprived of their antennae were degraded. In this configuration, the aerial robot relies solely on the measurements performed by the OF sensors and the rate gyro to adjust its forward speed and its clearance from the ground and the ceiling and automatically orient its eye relative to the nearest surface. Only the second OF feedback loop adjusting the robot's forward speed (in blue in the figure) was modified, and the output from the forward controller (which then became a proportional derivative controller) was then used directly as a setpoint by the nested feedback loop regulating the pitch rate. The conditions of our experiment can not be compared with the ones we could have outdoors due to the wind and the turbulences that would affect the stability of the rotorcraft which would therefore greatly benefit from the inner feedback loop regulating the airspeed.

Despite the removal of the airspeed feedback loop, it can be seen from figure 14(a) (extension 2) that BeeRotor II could still follow the terrain and adjust its forward speed based on the ventral and dorsal OFs, robustly avoiding all the obstacles encountered. The

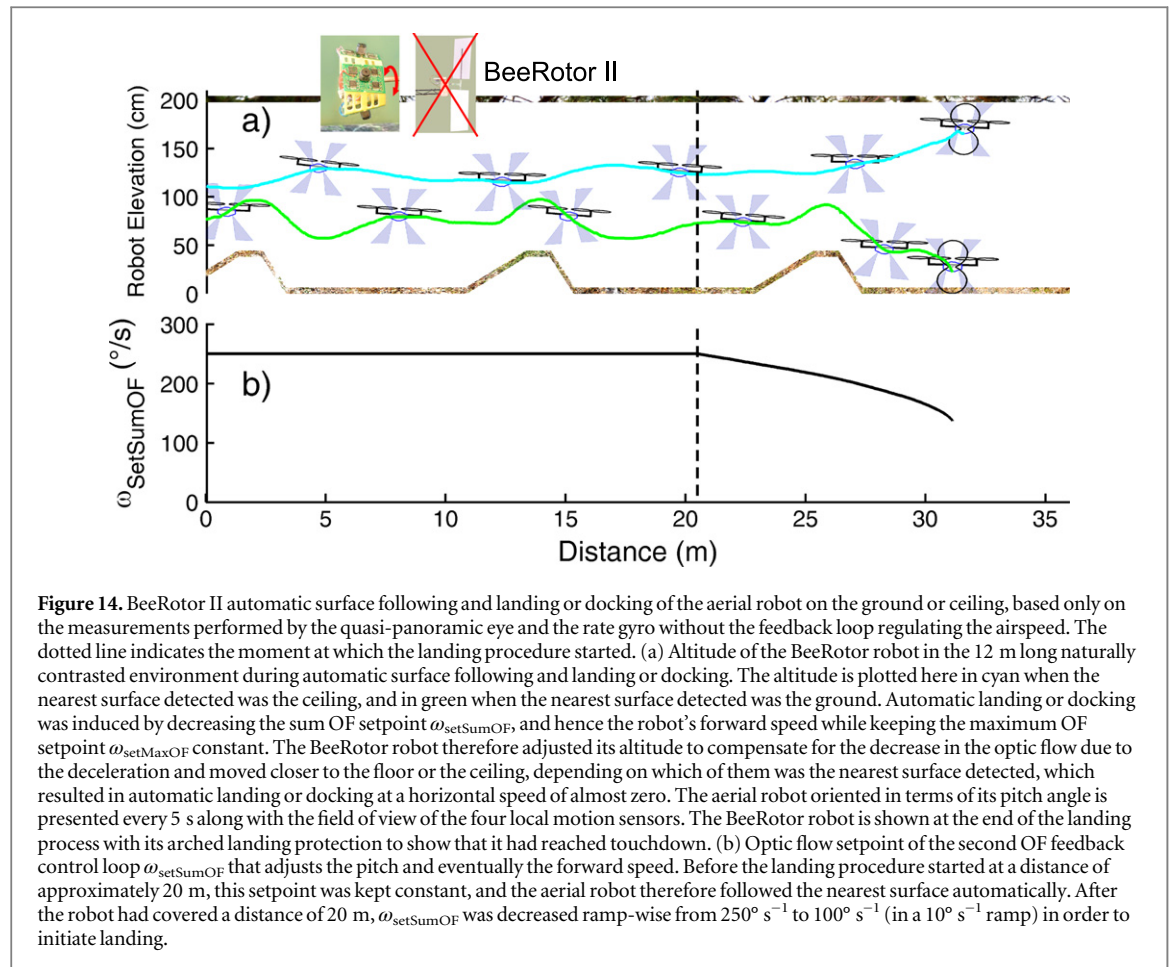


Figure 14. BeeRotor II automatic surface following and landing or docking of the aerial robot on the ground or ceiling, based only on the measurements performed by the quasi-panoramic eye and the rate gyro without the feedback loop regulating the airspeed. The dotted line indicates the moment at which the landing procedure started. (a) Altitude of the BeeRotor robot in the 12 m long naturally contrasted environment during automatic surface following and landing or docking. The altitude is plotted here in cyan when the nearest surface detected was the ceiling, and in green when the nearest surface detected was the ground. Automatic landing or docking was induced by decreasing the sum OF setpoint ω_{setSumOF} , and hence the robot's forward speed while keeping the maximum OF setpoint ω_{setMaxOF} constant. The BeeRotor robot therefore adjusted its altitude to compensate for the decrease in the optic flow due to the deceleration and moved closer to the floor or the ceiling, depending on which of them was the nearest surface detected, which resulted in automatic landing or docking at a horizontal speed of almost zero. The aerial robot oriented in terms of its pitch angle is presented every 5 s along with the field of view of the four local motion sensors. The BeeRotor robot is shown at the end of the landing process with its arched landing protection to show that it had reached touchdown. (b) Optic flow setpoint of the second OF feedback control loop ω_{setSumOF} that adjusts the pitch and eventually the forward speed. Before the landing procedure started at a distance of approximately 20 m, this setpoint was kept constant, and the aerial robot therefore followed the nearest surface automatically. After the robot had covered a distance of 20 m, ω_{setSumOF} was decreased ramp-wise from 250 $^{\circ} \text{s}^{-1}$ to 100 $^{\circ} \text{s}^{-1}$ (in a 10 $^{\circ} \text{s}^{-1}$ ramp) in order to initiate landing.

robot's altitude is presented here while it was following the ground (green curve) and the ceiling (cyan curve) with the same setpoint values $\omega_{\text{setMaxOF}} = 150^{\circ} \text{s}^{-1}$ and $\omega_{\text{setSumOF}} = 250^{\circ} \text{s}^{-1}$. The aerial robot kept on following the nearest surface detected by adjusting its two rotors' rotational speeds. The robot is shown here every 5 s: it can be seen from this figure that the robot adjusted its pitch angle and eventually its forward speed depending on the size of the tunnel, even when it was no longer equipped with an airspeed sensor.

In order to make the robot land automatically after travelling a little more than 20 m (dotted line), the sum OF setpoint ω_{setSumOF} was decreased rampwise from 250 $^{\circ} \text{s}^{-1}$ to 100 $^{\circ} \text{s}^{-1}$ in a 10 $^{\circ} \text{s}^{-1}$ ramp (see figure 14(b)) so as to gradually decrease its forward speed. In order to keep the maximum OF constant, the first feedback control loop gradually decreased the robot distance from the closest horizontal surface detected, so that it eventually landed or docked with a negligible forward speed at touchdown. The duration of the landing phase depended directly on the slope of the ramp imposed on the sum OF setpoint.

The results obtained in the present experiments (see figure 14 and extension 2) show that OF based visual guidance is possible both with and without absolute airspeed sensors: this may explain how insects improve their flight performances using the

airspeed or the changes in the air speed (the air acceleration) sensed by their antennae.

4.8. BeeRotor II summary

Based on three main feedback loops regulating the maximum value between the ventral and dorsal OFs and the sum of these OFs and orienting the robot's eye, BeeRotor II was able to follow terrain autonomously, adjust its forward speed to the size of the tunnel and avoid obstacles. The BeeRotor II robot is equipped for this purpose with a quasi-panoramic eye processing the OF generated by the ground and the ceiling. The eye is automatically kept parallel to the nearest surface by performing a least squares approximation on the OF pattern formed by the array of LMSs. This reorientation method enhances the BeeRotor aerial robot's ability to avoid steep obstacles even in a moving environment (see extension 2) without any need for absolute reference cues indicating the vertical, based, for example, on an accelerometer such as those commonly used these days in flying robots.

5. Conclusion

This is the first time an aerial robot has been presented which is able to fly in a steeply sloping, unstable environment without an accelerometer and without any need to refer to the absolute vertical. The

advantages of not using an accelerometer are that the strategies we presented here could be embedded into the lightest of robots, such as the insect-scale aerial robot weighing only a few hundred milligrams recently developed by Ma *et al* (2013). In addition, our strategies provide a useful alternative solution to those used in flapping wing robots as it is still not easy to use the information provided by an accelerometer on a flapping wing robot because of the oscillations of the body. By applying OF criteria, the aerial robot adjusts its pitch, its speed of flight and its altitude without requiring a state vector referring to the inertial reference frame.

The two novel guidance principles tested here are based on (i) an autopilot requiring no inertial reference frame (BeeRotor I) and (ii) OF measurements referred to the slope of the nearest surface followed (BeeRotor II). The autopilot mounted onboard the aerial robot functions almost entirely on the basis of the OF generated by the robot's own motion as it travels along a high-roofed tunnel, the ground and ceiling of which are lined with photographs of natural scenes. By adjusting the thrust and the differential speed of the propellers, it automatically reaches a 'safe height and a safe forward speed'.

The BeeRotor I aerial robot equipped with a full cylindrical CurvACE sensor proved to be able to automatically adjust its altitude and its forward speed based only on measurements performed in the robot body's frame of reference, even in a moving environment under several decades of background light level. The CurvACE sensor proved to perfectly meet the needs of MAVs because of its wide FOV, its small size, electrical consumption and weight, and its ability to measure the OF regardless of the background light level.

In the BeeRotor II aerial robot, the eye is automatically reorientated on the basis of the OF measurements so that it is always parallel to the nearest surface. This eye-reorientation principle improved the performances of the aerial robot and its ability to avoid steeply sloping obstacles even in a highly unstable environment, while eliminating the need for absolute reference cues giving the vertical direction. Thanks to its three direct OF feedback control loops, BeeRotor II opens up new avenues for the development of the MAVs of the future. Instead of using complex sensors and state estimators in the inertial frame of reference to fly autonomously, as previous vehicles of this kind have been designed to do, this novel autopilot functions on the sole basis of OF measurements in relation to the local environment.

Like flying insects endowed with compound eyes, the BeeRotor robot mainly relies on visual cues, but it is also equipped with a rate gyro which senses the inertial forces in a similar way to the halteres of *Diptera*, which have been found to be sensitive to Coriolis forces (Fraenkel and Pringle 1938). It is also equipped with a custom-made anemometer sensing the airspeed

as the hairs and antennae of flying insects do (Campan 1964, Burkhardt and Gewecke 1965). Thus equipped with sensors similar to a few of the various sensing modalities of flying insects, the BeeRotor robot successfully performed the complex flying tasks it was set. Even in the absence of an airspeed sensor, the BeeRotor rotorcraft proved to be able to navigate autonomously and land or dock smoothly in an unknown environment, using the bio-inspired guidance principles based on findings on how honeybees perform grazing landings (Srinivasan *et al* 2000). This simple landing procedure enables the robot to automatically land or dock, depending on whether the nearest surface is the floor or the ceiling, with a negligible forward speed at touchdown without any need for extra equipment or sensors on the aerial robot itself or on the ground.

However, the main limitation of the BeeRotor robot is that there is still a long way to go for a robot with such strategies to fly autonomously outdoors. Indeed, we have built this robot more as a proof of concept to prove the feasibility of flying without any information about the states of the robot in the inertial frame of reference. We have simplified the problem by tethering our aerial robot in order to perform repetitive tests under controlled conditions in the lab. Firstly, in order for the robot to freely fly in a 3D environment, it is necessary to adapt its mechanical structure and build, for example, a quadrotor in order to have the possibility to stabilize the aerial robot along its roll and yaw axes. Then, the idea would be to extend the eye-reorientation to two-dimensions and reproduce a similar strategy along the roll axis thanks to a visual sensor with a wide FOV like the CurvACE sensor. Finally, the feedback loops should be modified and use as setpoints the maximum OF generated by the closest surface (ground, ceiling or lateral walls) and the sum of the OFs in the horizontal or the vertical direction depending on which one is maximal. A similar strategy directly inspired by honeybees (Portelli *et al* 2011) has been successfully tested in simulation (Portelli *et al* 2010b). Another important difference between our experiments and a robot freely flying outdoor is the fact that we have used photographs of natural patterns and controlled illuminance conditions whereas outdoor conditions can be really more challenging for OF computation. However, as we have shown in Expert *et al* (2011), we now have OF sensors robust to illuminance variations that can accurately process OF outdoors.

One other limitation that can appear from the proposed strategies is the fact that the autopilot uses the OF generated by both the ground and the ceiling. Outdoors, robots like insects will mostly fly in environments where the ceiling is the sky which is mostly featureless (in the absence of clouds or nearby trees) and will therefore not generate measurable OF (if there is no rotational component present in the self-motion of the robot) in the dorsal visual hemisphere

Table A1. Index to multimedia extensions. (videos available online at stacks.iop.org/bb/10/026003/mmedia)

Extension	Type	Description
1	Video	Performances of the BeeRotor I robot equipped with CurvACE over unsteady terrain and under several decades of illuminance.
2	Video	Improved performances of the BeeRotor II robot flying over a highly unsteady environment including a steeply sloping obstacle. Even when the airspeed sensor was removed, the aerial robot was able to follow terrain and land automatically.

due to the large distance. In that case, with the proposed strategies, the BeeRotor robot will progressively increase its forward speed in order to reach the sum OF setpoint. As the forward speed increases, the altitude of the aircraft will also increase in order to maintain the ventral OF constant. By using only the sensors on the BeeRotor robot and without any estimation of the states of the rotorcraft in the inertial frame of reference, one elegant strategy to prevent the robot from accelerating continuously would be to use the reorientation angle. Indeed, if the eye reorientation angle exceeds a chosen threshold value, it would mean that the robot pitch angle with the local ground is important therefore indicating that the forward speed of the robot is significant. This information could be used to trigger a limiter placed after the Forward controller to limit the airspeed which would, ultimately, limit the altitude at which the BeeRotor robot can fly in an environment with a featureless ceiling.

The autonomous BeeRotor aerial robot inspired by insects' sensory systems is capable of achieving outstanding performances in a complex, unstable environment although it requires remarkably few resources as it does not require referenced states, an accelerometer and possibly even an airspeed sensor and is therefore perfectly suitable for the smallest of robots (Ma *et al* 2013) or for the spatial robots of the future where inertial information may not be available. By taking inspiration from flight guidance principles that have been developed during millions of years of evolution, we should be able to equip small MAVs (Duhamel *et al* 2013) with miniature autopilots at some point soon, enabling them to perform complex flight maneuvers safely in a variety of different environments.

Acknowledgments

We are very grateful to M Boyron for his involvement in the electronic design, F Paganucci, D Dray, Y Luparini and J Dipéri for their help with the mechanical design, J Blanc for improving the English manuscript and A Manecy, G Sabiron, G Portelli, J Serres, N Franceschini and S Viollet for their fruitful comments and suggestions during this research. This work was supported partly by CNRS Institutes (Life Science; Information Science; Engineering Science and Technology), the Aix- Marseille University, the French

National Research Agency -ANR- (EVA project under ANR-ContInt grant number: ANR-08-CORD-007-04) and the European Commission via the CURVACE project. The CURVACE project acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 237940.

Appendix A. Index to multimedia extensions

See table A1.

Appendix B. Dynamic identification

Propeller dynamics. In order to identify the transfer function between the propellers' control signals and the rotational speed of the propellers measured via a Hall effect sensor, a series of steps were applied to the propellers' command. A first order system with the following transfer function was identified:

$$G_{\text{propeller}}(s) = \frac{K_{\text{rotor}}}{\tau_{\text{rotor}}p + 1} = \frac{4.116}{0.1398s + 1}. \quad (\text{B.1})$$

The propellers' speed is regulated by a PI controller $C_{\text{propeller}}(s)$ in order to cancel the static error. This PI controller was adjusted in order to have a shorter response time than 0.1 s at 5% and an overshoot of less than 10%.

Forward speed dynamics. During the identification of the dynamics of the aerial robot, we noted that the robot's forward speed depended only on its body pitch, and that there was no coupling between the mean flight force and the horizontal speed. It was therefore possible to identify the transfer function between the differential thrust of the propellers and the robot's forward speed. To facilitate the feedback control, we first identified the dynamics between the pitch angle θ and the robot's forward speed G_{Speed} by applying a series of steps to the robot's pitch angle, which was controlled via a servomotor at several operating points around an initial pitch angle of 5.85°, corresponding to a forward speed of 1.3 m s⁻¹.

Table C1. Transfer functions of the BeeRotor robot.

$G_{\text{propeller}}(s)$	$\frac{\Omega_{\text{Rotors}}(s)}{U_{\text{Rotors}}(s)} = \frac{4.116}{0.1398s + 1}$
$G_{\text{Speed}}(s)$	$\frac{\delta V_x(s)}{\delta \theta(s)} = \frac{0.10498}{5.4312s + 1}$
$G_{\text{PitchRate}}(s)$	$\frac{\Delta \Omega_{\text{Rotors}}(s)}{V_x(s)} = \frac{-0.042s - 0.02683}{0.001742s^2 + 0.05445s + 1}$
$G_{\text{FwdSpeed}}(s)$	$\frac{\Delta \Omega_{\text{Rotors}}(s)}{V_{\text{air}_x}(s)} = \frac{1}{s} \cdot G_{\text{Speed}}(s) \cdot G_{\text{PitchRate}}(s)$
$G_{\text{speedCL}}(s)$	$\frac{V_x(s)}{V_{\text{air}_x\text{ref}}(s)} = \frac{1.0395}{1.62s^2 + 0.994s + 1}$
$G_{\text{SumOF}}(s)$	$\frac{\omega_{\text{setSumOF}}}{\text{Alt}(s)} = C_{\text{Forward}}(s) \cdot G_{\text{speedCL}}(s) \cdot \frac{d}{dV_x}(\omega_{V_{\text{trl}}} + \omega_{D_{\text{rsl}}})$
$G_{\text{Alt}}^{\theta_{\text{pitch}}}(s)$	$\frac{\Omega_{\text{Rotors}}(s)}{\text{Alt}(s)} = \frac{K_{\text{alt}}}{s^2 + 2 \cdot \zeta_{\text{alt}} \cdot \omega_0 \cdot s + \omega_0^2}$
$G_{\text{Alt}}^{3.5^\circ}(s)$	$\frac{\Omega_{\text{Rotors}}(s)}{\text{Alt}(s)} = \frac{0.1202}{0.8616s^2 + 0.33s + 1}$
$G_{\text{Alt}}^{15.3^\circ}(s)$	$\frac{\Omega_{\text{Rotors}}(s)}{\text{Alt}(s)} = \frac{0.419}{3.8852s^2 + 1.833s + 1}$
$G_{\text{MaxOF}}(s)$	$\frac{\omega_{\text{setMaxOF}}}{dh} = \frac{d}{dh} \left(\frac{1}{h} \right) \cdot C_{\text{Altitude}}(s) \cdot G_{\text{Alt}}(s)$

$$G_{\text{Speed}}(s) = \frac{K}{\tau s + 1} = \frac{0.10498}{5.4312s + 1}. \quad (\text{B.2})$$

The dynamic $G_{\text{PitchRate}}$ between the differential propellers' thrust $\Delta \Omega_{\text{Rotors}}$ and the robot's pitch rate $\dot{\theta}$ was then identified after removing the servomotor used previously. In order to avoid strong rotations of the robot along its pitch axis when steps were imposed on the differential thrust of the propellers, the identification was performed in the closed loop mode with a proportional derivative controller designed to roughly stabilize the pitch angle around a setpoint. Although the robot's pitch angle θ was not used during autonomous flight, it was estimated here using a complementary filter (Pflimlin 2007, Mahony *et al* 2008) fusing the inertial information provided by the accelerometer and the rate gyro. By applying a series of steps to the rotorcraft's pitch angle, we identified the dynamics between the differential thrust and the pitch rate, and obtained a second order system with a zero:

$$G_{\text{PitchRate}}(s) = \frac{-0.042s - 0.02683}{0.001742s^2 + 0.05445s + 1}. \quad (\text{B.3})$$

The whole transfer function linking the differential thrust of the propellers to the forward speed of the aerial robot can therefore be written:

$$G_{\text{FwdSpeed}}(s) = G_{\text{Speed}}(s) \cdot \frac{1}{s} \cdot G_{\text{PitchRate}}(s). \quad (\text{B.4})$$

The aerial robot's pitch rate $\dot{\theta}$ is stabilized via a PID controller $C_{\text{pitchRate}}(s)$ based on the rate gyro measurements, and the airspeed is regulated via a PD controller $C_{\text{Airspeed}}(s)$ based on the measurements performed by the airspeed sensor limiting the response time and preventing the system from overshooting. The transfer function between the airspeed setpoint and the forward speed of the aerial robot $G_{\text{speedCL}}(s)$, which was stabilized by these two nested loops, was identified by dynamically changing the robot's airspeed setpoint. The open loop transfer function linking the sum OF setpoint to the forward speed of the aerial robot $G_{\text{SumOF}}(s)$ was linearized to compute $C_{\text{Forward}}(s)$, which is a double phase lead controller designed to

increase the gain and the phase margin of the feedback loop.

Altitude dynamics. As in the case of the forward speed dynamics, the relationship between the mean speed of the propellers Ω_{Rotors} and the altitude of the aerial robot $G_{\text{Alt}}^{\theta_{\text{pitch}}}(s)$ was identified by applying a series of steps to the thrust of the propellers at several pitch angle values ranging between 3.75° and 15.3° .

Unlike the forward speed dynamics, the altitude response does not depend on a single parameter. The aerial robot's dynamic response depends not only on the propellers' speed Ω_{Rotors} but also on the robot's pitch angle θ . With each pitch angle value, the transfer function was identified between the propellers' speed and the altitude of the aerial robot, which always behaved like an underdamped second order system with a damping ratio ranging from $\zeta_{\text{alt}} = 0.17$ at a pitch angle of 3.75° to $\zeta_{\text{alt}} = 0.47$ with a pitch angle of 15.3° , whereas the gain ranged between $K_{\text{alt}} = 0.12$ and $K_{\text{alt}} = 0.42$ in the same range. In order to control the system without the accelerometric data and therefore the robot's pitch angle, we designed a controller that guarantees the stability of the system independent of the value of the pitch angle.

The open loop transfer functions linking the maximum OF setpoint to the robot's altitude $G_{\text{MaxOF}}(s)$ were linearized to determine the altitude controller $C_{\text{Altitude}}(s)$ which is a (PI) controller backed up with a double phase lead controller eliminating the static error and strongly increasing the gain of the feedback loop while preventing the robot from oscillating by increasing the phase margin.

Appendix C. Identified transfer functions

See table C1.

Appendix D. Controllers implemented onboard BeeRotor

See table D1.

Table D1. Transfer functions of the controllers mounted on the BeeRo-tor robot.

$C_{\text{Propeller}}(s)$	$\frac{U_{\text{Rotor}}(s)}{e_{\text{Rotor}}(s)} = \frac{0.5s + 5}{s}$
$C_{\text{PitchRate}}(z)$	$\frac{\Delta\Omega_{\text{Rotors}}(z)}{e_{\text{PitchRate}}(z)} = \frac{0.1057z^2 - 0.2056z + 0.1}{0.005z^2 + 0.005z}$
$C_{\text{Airspeed}}(z)$	$\frac{\theta_{\text{setpoint}}^{\text{Pitch}}(z)}{e_{\text{Airspeed}}(z)} = \frac{0.0626z - 0.06}{0.01z}$
$C_{\text{Forward}}(s)$	$\frac{V_{\text{setpoint}}^{\text{Airspeed}}(z)}{e_{\text{SumOF}}(z)} = 5 \cdot \left(\frac{9s + 26}{s + 8.665}\right) \cdot \left(\frac{9s + 26}{s + 8.665}\right) \cdot \left(\frac{1}{0.6s + 1}\right)$
$C_{\text{Altitude}}(s)$	$\frac{\Omega_{\text{Rotors}}(s)}{e_{\text{MaxOF}}(s)} = \left(\frac{1.5s + 0.3}{s}\right) \cdot \left(\frac{1.866s + 1}{0.134s + 1}\right) \cdot \left(\frac{1.866s + 1}{0.134s + 1}\right) \cdot \left(\frac{1}{0.6s + 1}\right)$
$C_{\text{Eye}}(s)$	$\frac{\theta_{\text{EyeCmd}}(s)}{\theta_{\text{Eye/Slope}}(s)} = \frac{0.0375s + 0.5003}{s + 0.5013}$

References

- Baird E, Srinivasan M, Zhang S, Lamont R and Cowling A 2006 Visual control of flight speed and height in the honeybee *From Animals to Animats 9* vol 4095 ed S Nolfi, G Baldassarre, R Calabretta, J Hallam, D Marocco, J-A Meyer, O Miglino and D Parisi (Berlin: Springer) pp 40–51
- Barron A and Srinivasan M 2006 Visual regulation of ground speed and headwind compensation in freely flying honey bees (apis mellifera l) *J. Exp. Biol.* **209** 978–84
- Barrows G and Neely C 2000 Mixed-mode VLSI optic flow sensors for in-flight control of a Micro Air Vehicle *Critical Technologies for the Future of Computing* vol 4109 (San Diego, CA: SPIE) 52–63
- Berry R, van Kleef J and G S 2007 The mapping of visual space by dragonfly lateral ocelli *J. Comparative Physiol. A* **193** 495–513
- Beyeler A, Zufferey J-C and Floreano D 2009a Optipilot: control of take-off and landing using optic flow *European Micro Aerial Vehicle Conf. (EMAV)* (Delft) 1–8
- Beyeler A, Zufferey J-C and Floreano D 2009b Vision-based control of near-obstacle flight *Auton. robots* **27** 201–19
- Briod A, Zufferey J-C and Floreano D 2013 Optic-flow based control of a 46 g quadrotor *Workshop on Vision-based Closed-Loop Control and Navigation of Micro Helicopters in GPS-denied Environment* (Tokyo: IROS)
- Bristeau P, Callou F, Vissiere D and Petit N 2011 The navigation and control technology inside the AR drone micro UAV *IFAC World Congr.* **18** 1477–84
- Burkhardt D and Gewecke M 1965 Mechanoreception in arthropoda: the chain from stimulus to behavioral pattern *Cold Spring Harbor Symp. Quantitative Biology* **30** pp 601–14
- Campan M 1964 Recherches sur la fonction dynamogène de l'antenne des insectes *Expériences sur la géotaxie ascensionnelle et la phototaxie positive de drosophila melanogaster meigen Comptes rendus des séances de la Société de biologie* **158** 2189–94
- Chahl J, Srinivasan M and Zhang S 2004 Landing strategies in honeybees and applications to uninhabited airborne vehicles *Int. J. Robot. Res.* **23** 101–10
- Conroy J, Gremillion G, Ranganathan B and Humbert J 2009 Implementation of wide-field integration of optic flow for autonomous quadrotor navigation *Auton. Robots* **27** 189–98
- de Croon G, Ho H, de Wagter C, van Kampen E, Remes B and Chu Q 2013 Optic-flow based slope estimation for autonomous landing *Int. J. Micro Air Vehicles* **5** 287–97
- Duhamel P-E, Perez-Arancibia N, Barrows G and Wood R 2013 Biologically inspired optical-flow sensing for altitude control of flapping-wing microrobots *IEEE Trans. Mechatronics* **18** 556–68
- Expert F, Roubieu F and Ruffier F 2012 Interpolation based 'time of travel' scheme in a visual motion sensor using a small 2D retina *IEEE Sensors Conf. (Taipei)* pp 2231–4
- Expert F and Ruffier F 2012 Controlling docking, altitude and speed in a circular high-roofed tunnel thanks to the optic flow *IEEE Int. Conf. on Robots and Systems (IROS)* (Vilamoura) pp 1125–32
- Expert F, Viollet S and Ruffier F 2011 Outdoor field performances of insect-based visual motion sensors *J. Field Robot.* **28** 529–41
- Fischler M and Bolles R 1981 Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography *Commun. ACM* **24** 381–95
- Floreano D *et al* 2013 Miniature curved artificial compound eyes *Proc. Natl Acad. Sci.* **110** 9267–72
- Fraenkel G and Pringle J 1938 Halteres as gyroscopic organs of equilibrium *Nature* **141** 919–20
- Fuller S, Straw A, Peek M, Murray R and Dickinson M 2014 Flying drosophila stabilize their vision-based velocity controller by sensing wind with their antennae *Proc. Natl Acad. Sci.* **111** 1182–91
- Garcia-Carrillo L-R, Flores G, Sanahuja G and Lozano R 2012 Quad-rotor switching control: an application for the task of path following *American Control Conf. (ACC)* (Montreal)
- Garratt M and Chahl J 2008 Vision-based terrain following for an unmanned rotorcraft *J. Field Robot.* **25** 284–301
- Green W, Oh P and Barrows G 2004 Flying insect inspired vision for autonomous aerial robot maneuvers in near-Earth environments *IEEE Int. Conf. on Robotics and Automation (ICRA)*
- Griffiths S, Saunders J, Curtis A, Barber B, McLain T and Beard R 2006 Maximizing miniature aerial vehicles *Robot. Autom. Mag.* **13** 34–43
- Hengstenberg R 1993 Multisensory control in insect oculomotor systems *Visual Motion and its Role in the Stabilization of Gaze* (Amsterdam: Elsevier) pp 285–98
- Hengstenberg R 1998 Biological sensors: controlling the fly's gyroscopes *Nature* **392** 757–8
- Herisse B, Hamel T, Mahony R and Russotto F-X 2010 The landing problem of a VTOL unmanned aerial vehicle on a moving platform using optical flow *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)* (Taipei) 77–89
- Hérissé B, Hamel T, Mahony R and Russotto F-X 2010 A terrain-following control approach for a vtol unmanned aerial vehicle using average optical flow *Auton. Robots* **29** 381–99
- Herisse B, Hamel T, Mahony R and Russotto F-X 2012 Landing a VTOL Unmanned Aerial Vehicle on a moving platform using optical flow *IEEE Trans. Robot.* **28** 77–89
- Herisse B, Russotto F-X, Hamel T and Mahony R 2008 Hovering flight and vertical landing control of a VTOL unmanned aerial vehicle using optical flow *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)* (Nice) 801–6
- Hyslop A and Humbert J 2010 Autonomous navigation in three-dimensional urban environments using wide-field integration of optic flow *J. Guid. control Dyn.* **33** 147–59
- Iida F 2001 Goal-directed navigation of an autonomous flying robot using biologically inspired cheap vision *Proc. of the XXXII Int. Symp. on Robotics (ISR)* (Seoul) pp 1404–9
- Kendoul F, Fantoni I and Nonami K 2009a Optic flow-based vision system for autonomous 3d localization and control of small aerial vehicles *Robot. Auton. Syst.* **57** 591–602
- Kendoul F, Nonami K, Fantoni I and Lozano R 2009b An adaptive vision-based autopilot for mini flying machines guidance, navigation and control *Auton. Robots* **27** 165–88
- Kendoul F, Yu Z and Nonami K 2010 Guidance and nonlinear control system for autonomous flight of minirotorcraft unmanned aerial vehicles *J. Field Robot.* **27** 311–34
- Kennedy J 1951 The migration of the desert locust (schistocerca gregaria forsk): i. The behavior of swarms: ii. A theory of long-range migrations *Phil. Trans. R. Soc.* **235** 163–290
- Koenderink J and Doorn A 1987 Facts on optic flow *Biol. Cybern.* **56** 247–54

- Kuenen L and Baker T 1982 Optomotor regulation of ground velocity in moths during flight to sex pheromone at different heights *Physiol. Entomology* **7** 193–202
- Labhart T 1980 Specialized photoreceptors at the dorsal rim of the honeybee's compound eye: polarizational and angular sensitivity *J. Comparative Physiol. A* **141** 19–30
- Labhart T and Meyer E 1999 Detectors for polarized skylight in insects: a survey of ommatidial specializations in the dorsal rim area of the compound eye *Microsc. Res. Tech.* **47** 368–79
- Ma K, Chirarattananon P, Fuller S and Wood R 2013 Controlled flight of a biologically inspired, insect-scale robot *Science* **340** 603–7
- Mahony R, Hamel T and Pflimlin J 2008 Nonlinear complementary filters on the special orthogonal group *IEEE Trans. Autom. Control* **53** 1203–18
- Mamiya A, Straw A, Tomasson E and Dickinson M 2011 Active and passive antennal movements during visually guided steering in flying *Drosophila* *J. Neurosci.* **31** 6900–14
- Mellinger D, Michael N and Kumar V 2012 Trajectory generation and control for precise aggressive maneuvers with quadrotors *Int. J. Robot. Res.* **31** 664–74
- Mittelstaedt H 1950 Physiologie des gleichgewichtssinnes bei fliegenden libellen *Z. für vergleichende Physiologie* **32** 422–63
- Moore R, Thurrowgood S, Soccol D, Bland D and Srinivasan M 2011 A method for the visual estimation and control of 3-dof attitude for uavs *Australasian Conf. on Robotics and Automation (ACRA) (Melbourne, Australia)*
- Moore R, Thurrowgood S, Bland D, Soccol D and Srinivasan M 2010 Uav altitude and attitude stabilization using a coaxial stereo vision system *IEEE Int. Conf. on Robotics and Automation (ICRA) (Anchorage, USA)* pp 29–34
- Moore R, Thurrowgood S and Srinivasan M 2012 Vision-only estimation of wind field strength and direction from an aerial platform *IEEE Int. Conf. on Intelligent Robots and Systems (IROS) (Vilamoura, Portugal)* pp 4544–9
- Nalbach G 1994 Extremely non-orthogonal axes in a sense organ for rotation: behavioural analysis of the dipteran haltere system *Neuroscience* **61** 149–63
- Parsons M, Krapp H and Laughlin S 2010 Sensor fusion in identified visual interneurons *Curr. Biol.* **20** 1–5
- Pflimlin J 2007 Commande d'un minidrone a helice carenee : de la stabilisation dans le vent a la navigation autonome *PhD Thesis Ecole Doctorale Systèmes de Toulouse*
- Portelli G, Ruffier F and Franceschini N 2010a Honeybees change their height to restore their optic flow *J. Comparative Physiol.* **196** 307–13
- Portelli G, Ruffier F, Roubieu F and Franceschini N 2011 Honeybees' speed depends on dorsal as well as lateral, ventral and frontal optic flows *PLoS One* **6** e19486
- Portelli G, Serres J, Ruffier F and Franceschini N 2010b Modelling honeybee visual guidance in a 3D environment *J. Physiol.* **104** 27–39
- Preiss R and Kramer E 1984 Control of flight speed by minimization of the apparent ground pattern movement *Localization and Orientation in Biology and Engineering* (Berlin: Springer) pp 140–2
- Roubieu F L, Expert F, Sabiron G and Ruffier F 2012a Two-directional 1 g visual motion sensor inspired by the fly's eye *IEEE Sensors J.* **13** 1025–35
- Roubieu F L, Serres J, Franceschini N, Ruffier F and Viollet S 2012b A fully-autonomous hovercraft inspired by bees; wall-following and speed control in straight and tapered corridors *IEEE Int. Conf. on Robotics and Biomimetics (ROBIO) (Guangzhou)* pp 1311–8
- Ruffier F and Franceschini N 2003 Octave, a bioinspired visuomotor control system for the guidance of micro-air vehicles (*Bioengineered and Bioinspired Systems* vol 5119) ed A Rodriguez-Vazquez, D Abbott and R Carmona (Bellingham, USA: SPIE) pp 1–12
- Ruffier F and Franceschini N 2004 Visually guided micro-aerial vehicle: automatic take off, terrain following, landing and wind reaction *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA) (New Orleans, LA)* pp 2339–46
- Ruffier F and Franceschini N 2005 Optic flow regulation: the key to aircraft automatic guidance *Robot. Auton. Syst.* **50** 177–94
- Ruffier F and Franceschini N 2008 Aerial robot piloted in steep relief by optic flow sensors *Int. Conf. on Intelligent Robots and Systems (IROS). Nice* **1** pp 1266–73
- Ruffier F and Franceschini N 2013 Optic flow regulation in unsteady environments: terrain following and landing evaluated by a tethered aerial robot *J. Intell. Robot. Syst.* doi:10.1007/s10846-014-0062-5
- Schuppe H and Hengstenberg R 1993 Optical-properties of the ocelli of calliphora erythrocephala and their role in the dorsal light response *J. Comparative Physiol. A* **173** 143–9
- 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 *Auton. Robots* **25** 103–22
- Shabayek A, Demonceaux C, Morel O and Fofi D 2012 Vision based uav attitude estimation: progress and insights *J. Intell. Robot. Syst.* **65** 295–308
- Shen S, Mulgaonkar Y, Michael N and Kumar V 2013 Vision-based state estimation for autonomous rotorcraft mavs in complex environments *IEEE Int. Conf. on Robotics and Automation (ICRA) (Karlsruhe)* pp 1758–64
- Soundararaj S, Sujeeth A K and Saxena A 2009 Autonomous indoor helicopter flight using a single onboard camera *IEEE Int. Conf. on Intelligent Robots and Systems* pp 5307–14
- Srinivasan M, Lehrer M, Kirchner W and Zhang S 1991 Range perception through apparent image speed in freely flying honeybees *Vis. Neurosci.* **6** 519–35
- Srinivasan M, Zhang S, Chahl J, Barth E and Venkatesh S 2000 How honeybees make grazing landings on flat surfaces *Biol. Cybern.* **83** 171–83
- Srinivasan M, Zhang S, Lehrer M and Collett T 1996 Honeybee navigation en route to the goal: visual flight control and odometry *J. Exp. Biol.* **199** 237–44
- Stange G 1981 The ocellar component of flight equilibrium control in dragonflies *J. Comparative Physiol.* **141** 335–47
- Strausfeld N, Seyan H and Milde J 1987 The neck motor system of the fly calliphora erythrocephala: I. Muscles and motor neurons *J. Comparative Physiol. A* **160** 205–24
- Strydom R, Thurrowgood S and Srinivasan M 2014 Visual odometry: autonomous uav navigation using optic flow and stereo *Australasian Conf. on Robotics and Automation (ACRA) (Melbourne)*
- Taylor G and Krapp H 2007 Sensory systems and flight stability: What do insects measure and why? *Insect Mechanics and Control* **34** pp 231–316
- Thurrowgood S, Soccol D, Moore R, Bland D and Srinivasan M 2009 A vision based system for attitude estimation of uavs *IEEE Int. Conf. on Intelligent Robots and Systems (IROS) (St Louis, MO)* pp 5725–30
- Ullman S 1981 Analysis of visual motion by biological and computer systems *Computer* **14** 57–69
- van Hateren J H and Schilstra C 1999 Blowfly flight and optic flow: II. Head movements during flight *J. Exp. Biol.* **202** 1491–500
- Viollet S and Zeil J 2013 Feed-forward and visual feedback control of head roll orientation in wasps (*Polistes humilis*, vespidae, hymenoptera) *J. Exp. Biol.* **216** 1280–91
- Whiteside T and Samuel G 1970 Blur zone *Nature* **225** 94–95
- Wilson M 1978 Generation of graded potential signals in the second order cells of locust ocellus *J. Comparative Physiol.* **124** 317–31
- Zufferey J-C, Beyeler A and Floreano D 2010 Autonomous flight at low altitude using light sensors and little computational power *J. Micro Air Vehicles* **2** 107–17
- Zufferey J-C and Floreano D 2006 Fly-inspired visual steering of ultralight indoor aircraft *IEEE Trans. Robot.* **22** 137–46