# Chapter 6

# Optic Flow to Steer and Avoid Collisions in 3D

Jean-Christophe Zufferey, Antoine Beyeler, and Dario Floreano

**Abstract** Optic flow is believed to be the main source of information allowing insects to control their flight. Some researchers have tried to apply this paradigm to small unmanned aerial vehicles (UAVs). So far, none of them has been able to demonstrate a fully autonomous flight of a free-flying system without relying on other cues such as GPS and/or some sort of orientation sensors (IMU, horizon detector, etc.). Getting back to the reactive approach suggested by Gibson (direct perception) and Braitenberg (direct connection from sensors to actuators), this chapter discusses how a few optic flow signals can be directly mapped into control commands for steering an aircraft in cluttered environments. The implementation of the proposed control strategy on a 10-g airplane flying autonomously in an office-sized room demonstrates how the proposed approach can result in ultra-light autopilots.

## 6.1 Introduction

Current unmanned aerial vehicles (UAVs) tend to fly far away from any obstacles, such as the ground, trees or buildings. This is mainly because aerial platforms have such tremendous constraints in terms of manoeuvrability and weight that enabling them to actively avoid collisions in cluttered or confined environments is highly challenging. Very often, researchers and developers use GPS as the main source of information to achieve what is commonly called "waypoint navigation". By carefully choosing the waypoints in advance, it is easy to make sure that the resulting path will be free of obstacles. The aerial robotics community has therefore, in effect, avoided tackling the collision avoidance problem since GPS has provided an easy way around it. However, the problem of 3D collision avoidance definitely deserves some attention in order to produce UAVs capable of flying at lower altitudes or even within buildings. Such capabilities would enable various applications such as search and rescue operations, low-altitude imagery for surveillance or mapping, environmental monitoring and wireless communication relays.

The classical approach to collision avoidance consists of relying on active distance sensors such as sonars, lasers or radars to construct and maintain a 3D map of the surroundings in real time. However, this results in very heavy and therefore potentially unsafe systems. The only known system that has been able to achieve near-obstacle flight using a 3D scanning laser range finder is a 100-kg helicopter equipped with a 3-kg scanning laser range finder [45]. Since the classical approach tends to be too heavy and power consuming for small flying platforms, we looked at biological systems for some hints about how to solve this problem using far less weight and energy. Flying insects are capable of dynamic navigation in cluttered environments while keeping energy consumption and weight at an incredibly low level. In general, biological systems far outperform today's robots at tasks involving real-time perception and action, in particular if we take energy efficiency and size into account.

It turns out that flying insects (Fig. 6.1) rely mainly on low-resolution monocular vision [34], gyroscopic [40], and airflow sensors [12] to control their flight.

J.-C. Zufferey (✉)
Laboratory of Intelligent Systems, EPFL, Lausanne, Switzerland
e-mail: Jean-Christophe.Zufferey@epfl.ch

D. Floreano et al. (eds.), *Flying Insects and Robots*,
DOI 10.1007/978-3-540-89393-6_6, 

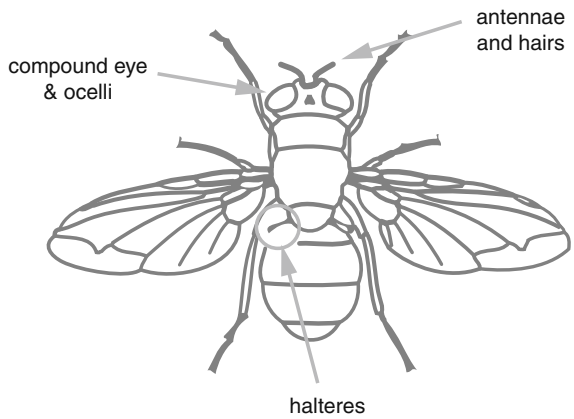


**Fig. 6.1** The most important perceptive organs related to flight control in flies: the large compound eyes (and the ocelli), the halteres and the antennas and hairs

This is interesting because the artificial counterparts of these sensors are now available in small, lightweight and power-efficient packages. Therefore, rather than opting for bulky active range finders weighing several kilograms, dynamic flight in the vicinity of obstacles can be achieved with far lower weight by using passive sensors such as vision, MEMS rate gyros and miniature anemometers.

Approximately two-thirds of the neurons in the insect brain are dedicated to visual information processing [56, 58]. Biologists have unravelled a significant part of their functioning. In summary, image motion, also called *optic flow*, plays a significant role in flight control by providing information on self-motion (see Chap. 9 or [33, 31]) and distances to surrounding objects (see Chap. 4 or [54, 58, 16]). In robots as in insects, optic flow can be estimated with very few pixels, allowing for the use of low-resolution vision sensors. The challenge is rather to have a large field of view (FOV) to cover divergent viewing directions and grab images as fast as possible to obtain good approximations of optic flow.

Optic flow is the perceived visual motion of surrounding objects projected onto the retina of an observer. The fact that visual perception of changes represents a rich source of information about self-motion and distances from surrounding objects is widely recognised [20]. If we assume a mobile observer moving in an otherwise stationary environment, the motion field describing the projection of the object velocities onto its retina depends on its self-motion (translation and rotation), the distance to the surrounding objects and the considered viewing directions [30]. In flying systems, it is usually feasible to estimate self-motion using a set of rate gyroscopes and an anemometer. Optic flow can thus be used to estimate the distance from surrounding objects.

## 6.2 Review of Optic Flow-Based Flying Robots

Several teams considered using optic flow to estimate or control the altitude or, more precisely, the height above ground of various unmanned flying systems. Most of them were inspired by an experiment with honeybees describing how these insects execute grazing landings on horizontal surfaces [52, 55]. The early attempts were carried out either in simulation [38] or with tethered helicopters (see Chap. 3 or [43]). The first experiment of altitude control with a free-flying airplane was performed outdoors by Barrows and colleagues in 2002 with a 1-m-large model airplane equipped with a custom-built optic flow detector [3]. A simple (on/off) altitude control law managed to maintain the aircraft airborne for 15 min, during which three failures occurred where the human pilot had to rescue the aircraft due to it dropping too close to the ground. Further experiments were carried out on larger platforms for controlling descent rate [11], altitude above a flat and homogeneous surface [60] or to simply help with estimating altitude above ground [1]. All these experiments were focused on controlling altitude while the lateral steering was remotely controlled by a human pilot.

The attempts at automating free-flying UAVs using bio-inspired vision are quite limited. In 2001, Barrows and colleagues described preliminary experiments on lateral obstacle avoidance in a gymnasium with a model glider carrying a single, laterally oriented, optic flow sensor [4]. An accompanying video shows the glider steering away from a wall when tossed towards it at a shallow angle. More recently, a team in Washington carried out a second experiment on lateral obstacle avoidance with an indoor aircraft equipped with a single lateral optic flow sensor [22]. A video shows the aircraft avoiding a basketball net in a sport hall. Again, only one sensor was used so that the aircraft could sense and avoid obstacles

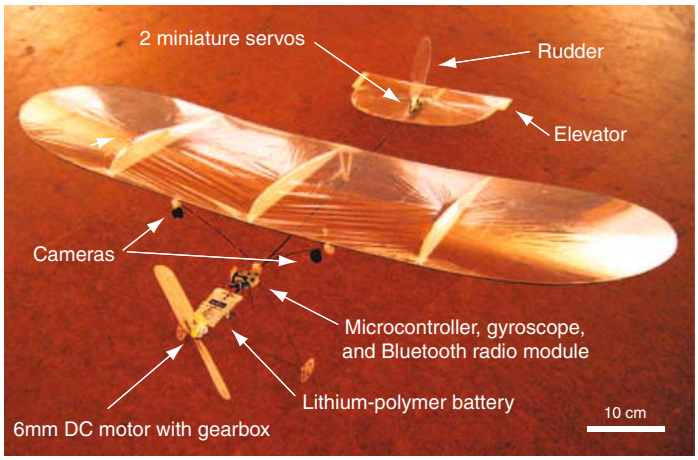


**Fig. 6.2** The *F2* indoor airplane. The on-board electronics consist of a 6-mm geared motor with a balsa-wood propeller, two miniature servos controlling the rudder and the elevator, a microcontroller board with a Bluetooth module and a rate gyro, two horizontal 1D cameras located on the leading edge of the wing and a 310 mAh lithium-polymer battery

only on one side. In 2006, we used a 30-g indoor airplane to demonstrate continuous and symmetrical collision avoidance in a relatively small indoor arena of 16 by 16 m [67]. The robot (Fig. 6.2) was equipped with two miniature, custom-made, optic flow detectors looking at 45° off the forward direction (Fig. 6.3). A model of landing response in flies [7] was used in this airplane to trigger saccadic turn-aways whenever a wall was too close, either in front or on either side of the airplane. While altitude and airspeed were manually controlled, the airplane was able to fly collision-free for more than 4 min without any lateral steering intervention from the safety pilot.[1] The plane was engaged in turning actions only 20% of the time, which means that it was able to fly in straight trajectories most of the time, except when very close to a wall. During a 4-min trial, it would typically generate 50 saccades and cover approximately 300 m in straight motion.

On larger outdoor platforms, Griffiths and colleagues have used optic flow mouse sensors as complementary distance sensors [23]. The UAV was fully equipped with an inertial measurement unit (IMU) and GPS. It computed the optimal 3D path based on an a priori map of the environment stored in its memory. In order to be able to react to unforeseen obstacles on the computed nominal path, it used a frontal laser range finder and two lateral optical mouse sensors. This robot demonstrated low-altitude flight in a natural canyon while the mouse sensors provided a tendency to move towards the centre of the canyon when the nominal path was deliberately biased towards one side or the other. Although no data showing the accuracy of optic flow-based distance measurements were provided, this experiment demonstrated that optical mouse sensors could be used to estimate rather large distances in outdoor environments. In 2005, Hrabar and colleagues [28] also employed lateral optic flow to enable a large helicopter to centre among obstacles outdoors, while stereo vision was utilised to avoid frontal obstacles. This work was quite similar to a study carried out in simulation by Muratet and colleagues the same year [39], which was focused on optic flow-based navigation in urban canyons. However, in these later projects the vision sensors were by no means used as primary sensors for navigation and the control system still

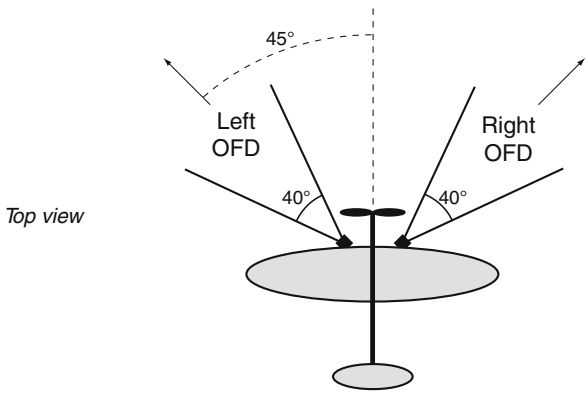


**Fig. 6.3** The arrangement of the two optic flow detectors on the *F2* airplane

[1] Video clips showing the behaviour of the plane can be downloaded from http://lis.epfl.ch/microflyers.

relied mainly on a classical autopilot using IMU and for some of them a GPS.

Although all these early attempts at mimicking insects and using optic flow to achieve flight control and collision avoidance are remarkable, none of them have reached the holy grail of completely automating a free-flying UAV without relying on additional information such as GPS, attitude estimates (IMU) or 3D maps. The first project which got close to achieving this goal was carried out in simulation by Neumann and Bülthoff [42]. A flying agent with a relatively simple helicopter-like dynamics could stabilise its course, control its altitude and avoid obstacles like trees using fly-inspired optic flow and matched filters [32, 63]. For this control strategy to work, the agent was required to be flying level at all time. To ensure this, the attitude of the agent was constantly regulated by a mechanism relying on the ambient light intensity gradient (there is more light when looking at the sky than at the ground). This mechanism was loosely inspired from the dorsal light response found in insects [47]. However, for this attitude control mechanism to work, the environment needs to have a well-defined light intensity gradient. This is easy to ensure in simulated worlds, but not always so in real-world conditions because the sky can be occluded by buildings or trees and the sun will almost never be directly overhead.

Only very recently, we were able to eliminate the need for explicitly estimating or regulating attitude and to rely exclusively on optic flow to achieve fully autonomous flight with a simulated airplane in a confined environment [5]. In this experiment, optic flow was computed in three specific directions (left, right and down) at 45° off the longitudinal axis of the aircraft. Although utilising a dynamics model more realistic than most previous work in simulation, this research still lacked testing in reality. In this chapter, we describe what represents – to the best of our knowledge – the first example of a fully autonomous physical airplane achieving optic flow-based navigation. This endeavour was mainly driven by an attempt at finding the minimal set of sensors and control strategy allowing for fully autonomous flight. Relying on the minimal number of pixels and optic flow detectors (OFD) was taken as a challenge, which may become less formidable as the vision sensors become lighter and faster, or when using larger platforms.

## 6.3 Optic Flow

As an observer moves through its environment, the pattern of light reflected on its retina changes continuously, creating *optic flow* [20], formally defined as the apparent motion of the image intensities or brightness patterns. This optic flow is very useful for navigation because it contains information regarding both the self-motion of the observer (Chap. 9) and the distances to surrounding objects. To study how optic flow depends on self-motion and distances, people usually rely on the notion of *motion field* (sometimes also called *velocity field*), which consists of the 2D projection onto the observer's retina of the relative 3D motion of scene points. In contrast to the actual optic flow field, the motion field is a purely geometrical concept, which is independent of light intensities. Ideally, the optic flow corresponds to the motion field, but this may not always be the case [26]. The main reasons for discrepancies between optic flow and motion field are the possible absence of detectable contrasts on the surrounding objects, or the so-called aperture problem.[2] In practice, only the optic flow can be measured. However, the behaviour of optic flow is best understood by means of the geometrical motion field, and discrepancies between the two are often treated as noise. Therefore, in the rest of this chapter, we will not make this distinction anymore, but we shall keep in mind that when taken as motion field estimates, optic flow measurements may be quite noisy because of the aforementioned problems.

Let us consider a spherical vision system (Fig. 6.4) moving through a stationary environment. The image is formed by spherical projection of the environment onto this sphere. Apart from resembling a fly's eye, the use of a spherical projection makes all points in the image geometrically equivalent, thus simplifying the mathematical analysis.[3] The photoreceptors of the vision sensor are thus assumed to be arranged on this

[2] If the motion of an oriented element is detected by a unit that has a small FOV compared to the size of the moving element, the only information that can be extracted is the component of the motion perpendicular to the local orientation of the element [37, 25, 36]. Interestingly, there is good evidence that flies do not solve the aperture problem [8].

[3] Ordinary cameras are usually not using a spherical projection model. However, if the field of view is not too wide, the spherical

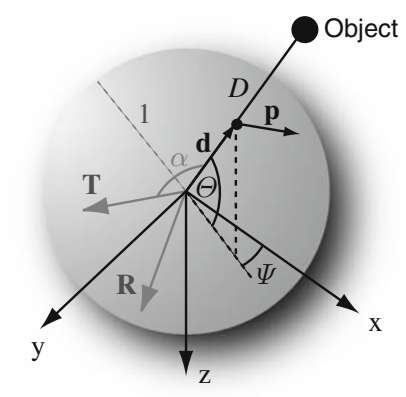


**Fig. 6.4** The spherical model of a visual ensor. A viewing direction indicated by the unit vector **d**, which is a function of azimuth $\Psi$ and elevation $\Theta$ (spherical coordinates). The distance to an object in the direction $\mathbf{d}(\Psi,\Theta)$ is denoted as $D(\Psi,\Theta)$. The optic flow vectors $\mathbf{p}(\Psi,\Theta)$ are always tangential to the sphere surface. The vectors **T** and **R** represent the translation and rotation, respectively, of the visual sensor with respect to its environment. As will be seen in the next section, the angle α between the direction of translation and a specific viewing direction is sometimes called *eccentricity*

unit sphere, each photoreceptor defining a viewing direction indicated by a unit vector $\mathbf{d}(\Psi,\Theta)$, which is a function of both the azimuth $\Psi$ and the elevation $\Theta$ in a spherical coordinate system. When this vision system undergoes a 3D movement described by its translation and rotation vectors, **T** and **R**, the motion field $\mathbf{p}(\Psi,\Theta)$ on the surface of the sphere is given by [30]

$$\mathbf{p}(\Psi,\Theta) = \left[ -\frac{\mathbf{T} - (\mathbf{T}\cdot\mathbf{d}(\Psi,\Theta))\,\mathbf{d}(\Psi,\Theta)}{D(\Psi,\Theta)} \right] + [-\mathbf{R}\times\mathbf{d}(\Psi,\Theta)], \tag{6.1}$$

where $D(\Psi,\Theta)$ is the distance between the sensor and the object seen in a particular direction $\mathbf{d}(\Psi,\Theta)$. This equation is fundamental as it tells us all the basic properties of optic flow, which is a linear combination of a translational and a rotational component due, respectively, to the motion along **T** and around **R** The translational component of optic flow, hereafter denoted TransOF, is inversely proportional to distances from the surrounding objects $D(\Psi,\Theta)$, whereas the rotational component, RotOF, displays no dependency on distance. The translational optic flow field in fact depends on (i) the velocity of the observer, (ii) its distance to objects in the surroundings and (iii) the angles between the objects and the direction of the translation.

A particular case of the general equation (6.1) is often used in biology [64, 27, 49] and in robotics [19, 50, 62, 35] to explain depth perception from optic flow. The so-called *motion parallax* refers to a planar situation where only pure translational motion is considered (Fig. 6.5). In this case, it becomes straightforward[4] to express the optic flow amplitude $p$ (also referred to as *apparent angular velocity*) elicited by an object at distance $D$, seen at an angle $\alpha$ with respect to the motion direction **T**:

$$p(\alpha) = \frac{||\mathbf{T}||}{D(\alpha)}\sin\alpha, \text{ where } p = ||\mathbf{p}||. \tag{6.2}$$

Note that the angle $\alpha$ is also called *eccentricity*, in particular if **T** is aligned with the reference axis of the vision system. From this equation, it becomes evident that if the translational velocity and the optic flow amplitude can be measured, the distance from the object can easily be retrieved as follows:

$$D(\alpha) = \frac{||\mathbf{T}||}{p(\alpha)}\sin\alpha. \tag{6.3}$$

Therefore, if the translation (direction and amplitude) is known, the distance to the surrounding objects

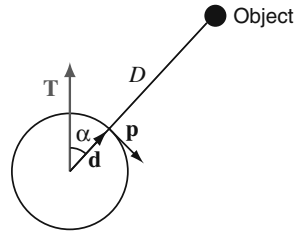


**Fig. 6.5** The motion parallax. The *circle* represents the retina of a moving observer. The *symbols* are the same as defined in Fig. 6.4

approximation is reasonably close [41]. A direct model for planar retinas can be found in [17].

[4] To derive the motion parallax equation (6.2) from the general optic flow equation (6.1), the rotational component must first be cancelled since no rotation occurs. Then, the translation vector **T** can be expressed in the orthogonal basis formed by **d** (the viewing direction) and $\frac{\mathbf{p}}{||\mathbf{p}||}$ (the normalised optic flow vector).

can be estimated by this means. If the agent is rotating, however, derotation of optic flow is required as a preliminary step to depth perception, otherwise the RotOF component may completely distort the distance estimates. Interestingly, most flying insects display mechanisms that tend to minimise the rotational content of optic flow. For instance, flies move in straight segments interspersed with rapid turning actions called saccades [13, 61]. In addition to moving straight most of the time, larger flies actively stabilise their gaze by orienting their head with respect to their body in order to compensate for undesired rotations, a phenomenon referred to as gaze stabilisation (see Chap. 4 or [46, 29]). Similar mechanisms allowing for keeping rotational optic flow at a minimum level have been found in many other species as well [65]. However, when thinking in terms of flying robots, saccadic flight and gaze stabilisation may not always be the most simple and effective way of derotating the optic flow. An alternative method, which can be implemented in software, consists of compensating for the spurious rotational optic flow signals by subtracting the theoretically derived rotational motion field according to Eq. (6.1). This can be achieved by means of rate gyros, which allow to easily reconstruct the rotation vector. This alternative solution can be seen as a substitute for gaze stabilisation. Indeed, performing derotation in software is probably more efficient with artificial systems than implementing mechanical gaze stabilisation, which would require complex mechanisms to pan, tilt and roll the entire vision system.

The remaining question is: how can optic flow be sensed? There are several ways of doing it. Hassenstein and Reichardt [24] have proposed a model called elementary motion detector (EMD) to explain how local optic flow detection could be achieved in the fly visual system (see Chap. 4, Sect. 4.3 for a review). However, the response of EMDs depends not only on image velocity (which would allow them to more accurately represent the motion field) but also on the contrast and the spatial frequency content of the scene. In computer vision, various algorithms have been proposed, which take as input a time series of visual frames from standard cameras (see [2] for a review). These algorithms generally produce outputs linearly following the image motion and are therefore much closer to the actual motion field than EMDs. However, they tend to be computationally expensive. Quite recently, people have been working on faster algorithms [51, 10], which are better suited to real-time computation and embedded processors. An alternative approach consists of using optic flow sensors such as optical mouse sensors (see Chap. 9) or custom-designed analog detectors (see Chap. 8), which feature fast, on-chip optic flow detection.

To sum up, optic flow contains information on distance to obstacles and can be sensed using either dedicated sensors or a camera connected to a processor running one of the various existing algorithms. However, distance information in optic flow is mixed with other factors such as rotation speed and orientation, translation speed and orientation, and looking direction with respect to the translation direction. In brief, if rotation and velocity can be assumed to remain constant or estimated using alternative sensors such as inertial or airflow sensors, derotated optic flow can be directly interpreted as a proximity indication and thus serve as input to flight control and obstacle avoidance.

## 6.4 Control Strategy

Several researchers have considered what can be classified as 2D optic flow-based control strategies. They were working either with robots moving on flat surfaces (see Chap. 3 and [50, 14, 44, 53, 48]) or with constrained flying robots that were either tethered helicopters (see Chap. 3 or [43]) or horizontally flying agents (see Chap. 5, and [42, 39]). Here instead, we aim at controlling aircraft moving in 3D and relying on roll and pitch movements in order to steer. Airplanes and helicopters in translational flight (as opposed to near-hover or aerobatic flight) are indeed using rolling and pitching movements to alter their trajectory (see Fig. 6.6).[5] Rolling allows them to preset the direction in which a pitching movement will reorient the aircraft translation vector, which corresponds to its flight direction. This can be observed in any standard turn where translating aircraft first rolls by a few degrees before pitching up to curve their trajectory. It is inter-

[5] We deliberately let aside the yawing movements in this discussion because in standard flight they can be seen as a byproduct of the steering mechanism. Indeed, yawing is usually employed in aircraft as a way of cancelling any lateral acceleration to achieve what is known as “coordinated turns” [56].

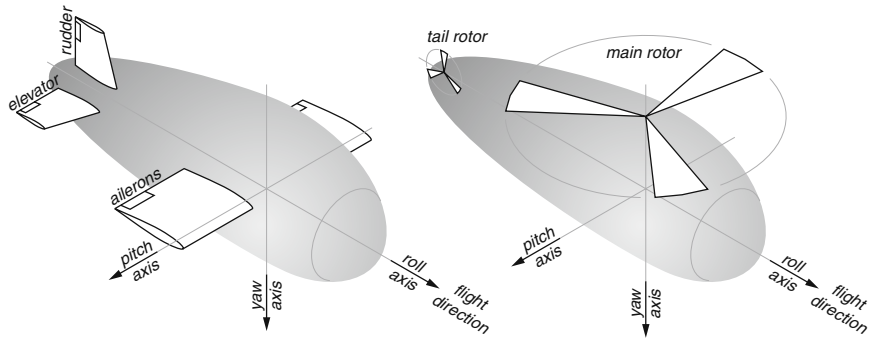


**Fig. 6.6** Idealised translating aircraft (winged airplane or a rotorcraft) can steer by rotating around their roll and pitch axes. The yaw rate is passively or actively controlled so that no lateral acceleration occurs (coordinated turns). The lift produced by the wings or the rotor is assumed to exactly compensate for the gravity and centripetal forces. Under these conditions, the translation vector (flight direction) remains fixed with respect to the aircraft body

esting to note that these steering dynamics do not seem to be shared by flying insects, which tend to generate quite significant side-slip movements and are able to alter their trajectory while keeping their head orientation stable [46]. However, most existing flying platforms (especially fixed-wing aircraft) are unable to display such capabilities typically found in insects and are therefore well described by this translating aircraft model.

Translating aircraft can be assumed to have a translation vector that is always aligned with their main axis. This is of utmost importance because it allows to greatly simplify the interpretation of TransOF since the eccentricity angle alpha (Fig. 6.4) can be assumed to be constant for any given viewing direction. Furthermore, assuming that the velocity of the aircraft is kept constant, the output of derotated optic flow detectors (DOFD) can be directly interpreted as a proximity signal. Note that if the assumption of constant speed does not hold, the TransOF signals will still provide estimates of the time to contact (instead of proximity), which is perfectly suitable as control input in most cases.

Aiming at a simple control strategy that can fit any small microcontroller, we propose to follow a reactive paradigm where perception is directly linked to action without intermediary cognitive layers [20, 21, 9, 15]. Since optic flow can be turned into proximity information as seen above, the simplest way of achieving an efficient reactive behaviour of an aircraft is to linearly combine a series of DOFDs to command its rolling and pitching rates. Since no assumption can be made regarding the orientation of the aircraft with respect to the surrounding environment, the only policy that can be adopted is to react to any sensed object in the simplest way: steering away from it with a magnitude that is proportional to its proximity. When nothing is sensed, just keep moving straightforward. Braitenberg's vehicles [9] are good examples of reactive systems using such a control strategy to directly map sensor activations into actuator commands. Many wheeled robots are using such a scheme for low-level obstacle avoidance (e.g. [18]). Here however, we need to adapt it to 3D-moving systems featuring the specific dynamics of translating aircraft.

By using four DOFDs looking left/right and up/down with some eccentricity angle with respect to the flight direction (Fig. 6.7), a very simple, Braitenberg-like control strategy can be devised to

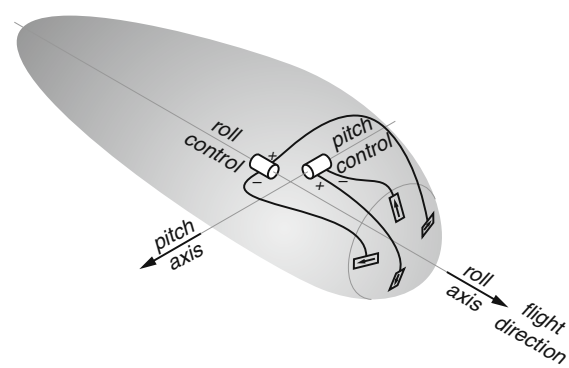


**Fig. 6.7** A 3D Braitenberg-like vehicle using optic flow to control its flight and to steer away from obstacles. The underlying vehicle is a translating aircraft with the only possibility to roll and pitch to alter its trajectory

produce reactive collision avoidance. The left/right detectors should be linked to the roll rate control in such a way that if an obstacle is detected on one side, the aircraft rolls away from it. Similarly, the up/down sensors should control the pitch rate so as to steer away from objects sensed in the upper or lower regions of the field of view.

This reactive control strategy has several advantages. First of all, the fact that the optic flow is inversely proportional to the distance means that far objects will have little impact on the controls, whereas very close obstacles will induce strong reactions. Second, this control strategy does not require any kind of inclinometers in order to fly straight over flat surfaces. As soon as the aircraft inclines left or right it will sense the proximity of the ground on that side and directly correct towards the opposite side. In some sense, the attitude control commonly seen in classical autopilots here comes for free. Third, no assumption needs to be made on the layout of the environment. In principle, this strategy works as well in open terrains as in cluttered environments, the only limitation being that the objects need to be large enough to be perceived early enough by the optic flow detectors. Note that if the environment has no, or high, ceilings and the gravity attracts the aircraft towards the ground at all time, the top detector may be useless. The same holds ture for outdoor environments where obstacles hanging in the sky are very rare, if the aircraft does not need to fly in tunnels or under bridges.

Note that this way of directly connecting translational optic flow to controls using (weighted) lines is reminiscent of the structure of the visuomotor pathway of insects (Chap. 4, Sect. 3). Tangential cells basically combine local optic flow signals and connect to descending cells, which modulate the wing kinematics in order to steer.

## 6.5 Application to a 10-g Indoor Microflyer

As a first step towards the realisation of minimalist, but completely autonomous free-flying systems, we looked at indoor flight. More precisely, we decided to test whether the previously described control strategy could work in the office-sized room shown in Fig. 6.8. Flying indoors requires slow motion, high manoeuvrability and small size, which calls for ultra-light overall weight, resulting in tremendous constraints in terms of sensors and computational power. In addition, flying indoors means that the aircraft needs to constantly steer in order to avoid hitting the ground or the walls.

### *6.5.1 Platform*

The prototype we developed [68], called *MC2* (Fig. 6.9), consists mainly of carbon-fibre rods and thin Mylar plastic films. The wing and the battery are connected to the frame by small magnets such that they can easily be taken apart. The propulsion is ensured by a 4-mm brushed DC motor, which transmits its torque to a lightweight carbon-fibre propeller via a

**Fig. 6.8** The 7 × 6-m test room features eight projectors on the ceiling, each projecting on half of the opposite wall. This system permits an easy modification of the textures on the walls. The ground is covered by a randomly textured carpet in order to provide enough detectable contrasts even for a low-performance vision system

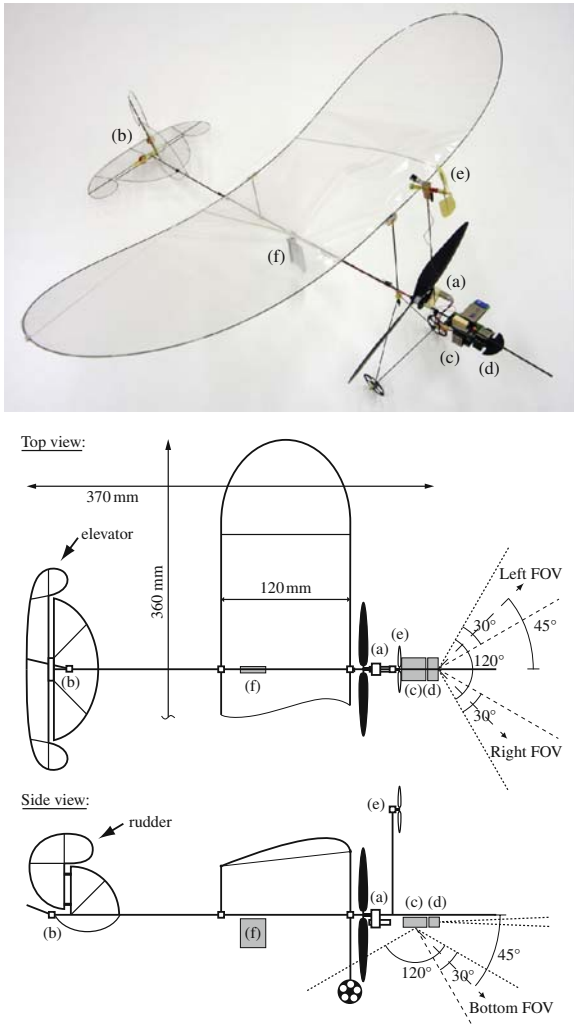


**Fig. 6.9** The 10-g *MC2* microflyer. The on-board actuators and electronics consist of (**a**) a 4-mm geared motor with a lightweight carbon-fibre propeller, (**b**) two magnet-in-a-coil actuators controlling the rudder and the elevator, (**c**) a microcontroller board with a Bluetooth wireless communication module and a ventral camera with its pitch rate gyro, (**d**) a front camera with its yaw rate gyro, (**e**) an anemometer and (**f**) a 65 mAh lithium-polymer battery

1:12 gearbox. The rudder and elevator are actuated by two magnet-in-a-coil actuators. Note that no ailerons are present on the main wing, but the action of the rudder has a direct impact on the roll rate as on most high-winged aircrafts. The propeller has been placed unusually close to the wing leading edge in order to free the frontal field of view for the vision system. The total weight of the *MC2* reaches 10.3 g. The airplane is capable of flying in reasonably small spaces (about $3 \times 3$ m) at low velocity (around 1.5 m/s). The average power consumption of the entire system is on the order of 1 W and the on-board 65 mAh lithium-polymer battery provides an endurance of about 10 min.

Regarding the sensor suite, the exact same sensory modalities as in flies is implemented on the *MC2*. Due to the tight available payload, the vision system is made of two wide FOV linear cameras of 102 pixels each (as shown in the bottom part of Fig. 6.9). Only three segments of 20 pixels out of these two cameras are selected for optic flow extraction in three specific directions: left, right and down. Figure 6.10 shows the regions covered by the two cameras and the zones in them where optic flow is extracted. In addition, two MEMS gyros are used to sense pitching and yawing rates in order to derotate the optic flow signals (further details about how this is achieved can be found in [67]). Note that no roll gyro is present on the plane

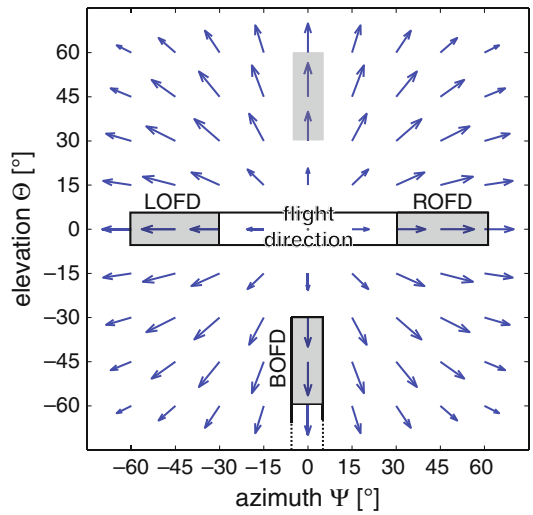


**Fig. 6.10** An azimuth-elevation graph displaying the typical optic flow field experienced during straight motion. The zones covered by the cameras mounted on the *MC2* are represented by the two *thick rectangles*. By carefully defining the regions where the optic flow algorithm is applied (*grey zones* within the *thick rectangles*), three radial optic flow detectors (OFD) can be implemented at an equal eccentricity of 45° with respect to the flight direction. Note that this angle is not only chosen because it fits the available cameras but also because the maximum OF values occur at $\alpha = 45°$ when approaching a fronto-parallel surface. These OFDs are prefixed with L, B and R for left, bottom and right, respectively. A fourth OFD could have been located in the top region (also at 45° eccentricity), but since the airplane never flies inverted and the gravity attracts it towards the ground, there is no need for sensing obstacles in this region

because rolling motion cannot be detected by radially oriented lines of pixels. Of course, rotations about the roll axis can produce spurious radial optic flow signals, but these can be treated as noise. Finally, a small custom-made anemometer ensures the functionality of airflow sensing in order to provide a rough estimate of the airspeed. These sensors are connected to the on-board 8-bit microcontroller, which processes image sequences to extract optic flow in the three viewing directions using an image interpolation algorithm [51, 67]. These optic flow signals are then derotated by subtraction of the related rate gyro signal [66]. In fact, these processing steps implement three derotated optic flow detectors (DOFD).

### 6.5.2 Control Strategy

Equipped with such DOFDs that act as proximity sensors, the implementation of the presented control strategy is straightforward. If an obstacle is detected on one side, the airplane should roll away using its rudder.[6] If the proximity signal increases in the ventral part of the FOV, the airplane should pitch up using its elevator. As suggested by Braitenberg, this is achieved through direct connections between the DOFDs and the control surfaces (Fig. 6.11). In practice, some gains and thresholds should be implemented on the connections in order to be able to tune the resulting behaviour. In Fig. 6.11, these parameters are hidden in the transfer functions $\Omega$.

In order to maintain airspeed in a reasonable range (above stall and below over-speed), the anemometer signal is compared to a given set point before being used to proportionally drive the propeller motor. Note that this airspeed control process also ensures a reasonably constant $||\mathbf{T}||$ in Eq. (6.3).

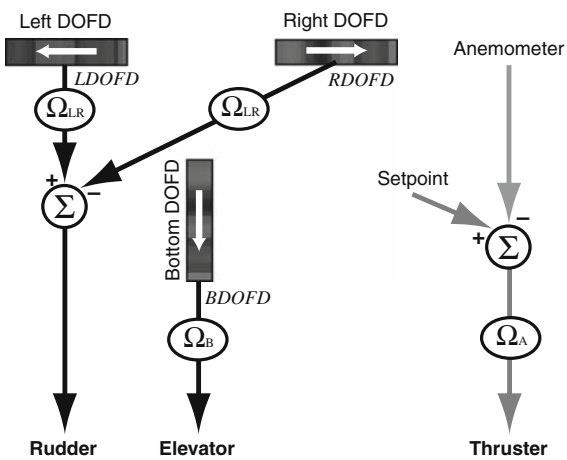


**Fig. 6.11** The control strategy allowing for autonomous operation of the *MC2*. The three OFDs are prefixed with D to indicate that they are filtered and derotated (this process is not explicitly shown in the diagram). The signals produced by the *left* and *right* DOFDs, i.e. *LDOFD* and *RDOFD*, are basically subtracted to control the rudder, whereas the signal from the bottom DOFD, i.e. *BDOFD*, directly drives the elevator. The anemometer is compared to a given set point to output a signal that is used to proportionally drive the thruster. The $\Omega$ ellipses indicate that a transfer function is used to tune the resulting behaviour. These are usually simple multiplicative factors or combinations of a threshold and a factor

### 6.5.3 Results

After some tuning of the parameters included in the $\Omega$ transfer functions, the airplane could be launched by hand in the air and flew completely autonomously in its test arena (Fig. 6.8).[7] Figure 6.12 shows the data recorded during such a flight over a 90-s period. In the first row, the higher right DOFD signal indicates that the airplane was launched closer to a wall on its right, which produced a leftward reaction (indicated by the negative yaw gyro signal) that was maintained throughout the trial duration. Note that in this environment, there is no good reason for modifying the initial turning direction since flying in circles close to the walls is more efficient than figure of eights, for instance. However, this first graph clearly shows that the controller does not simply hold a constant turning rate. The rudder deflection is continuously adapted according to the DOFD signals, which leads to a continuously varying yaw rotation

[6] On lightweight airplanes, the ailerons on the wings are often replaced by a rudder (vertical control surface located at the rear of the airplane), which produces an asymmetry in lift between the left and the right wings by introducing a momentary sideslip angle. As a result, the airplane will roll because of one wing dipping and the other lifting.

[7] A video of this experiment is available for download at http://lis.epfl.ch/microflyers.

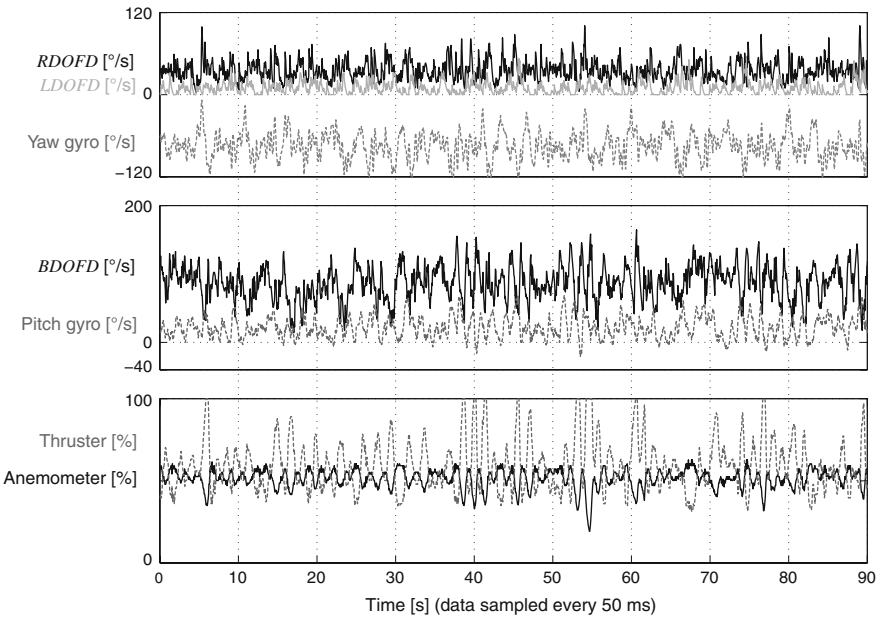


**Fig. 6.12** A 90-s autonomous flight with the *MC2* in the test arena. The first row shows lateral OF signals together with the yaw rate gyro. The second row plots the ventral OF signal together with the pitch rate gyro. The third graph displays the evolution of the anemometer value together with the motor setting. Flight data are sampled every 50 ms, corresponding to the sensory-motor cycle duration

rate. The average turning rate of approximately 80°/s indicates that a full rotation around the room is accomplished every 4–5 s. Therefore, a 90-s trial corresponds to approximately 20 circumnavigations of the test arena.

The second graph shows that the elevator actively reacts to the bottom DOFD signal, thus continuously affecting the pitch rate. The non-zero mean of the pitch gyro signal is due to the fact that the airplane needs to bank for turning. Therefore, the pitch rate gyro also measures a component of the overall circling behaviour. It is interesting to note that the elevator actions are due to the proximity not only of the ground but also of the walls. When the airplane detects the proximity of a wall on its right, the rudder action increases its leftward bank angle. In this case the bottom DOFD is oriented directly towards the close-by wall and no longer towards the ground. In most cases, this would result in a quick increase in the bottom DOFD signal and thus trigger a pulling action of the elevator. This reaction is highly desirable since the absence of a pulling action at steep roll angles would result in an immediate loss of altitude.

The bottom graph shows that the motor power is continuously adapted according to the anemometer value. In fact, as soon as the controller steers up due to a high ventral optic flow, the airspeed quickly drops, which needs to be counteracted by a prompt increase in power.

## 6.6 Conclusions and Outlook

Considering the typical motion constraints of translating aircraft on the one hand, and the properties of optic flow on the other hand, allowed us to develop a generic 3D control strategy allowing for autonomous flight in the presence of obstacles. We showed that a simple reactive approach à la Braitenberg can efficiently solve this problem while relying on extremely lightweight and low-power sensors as insects do. Furthermore, no explicit measurement of aircraft attitude nor active gaze stabilisation is required. The proposed control strategy steers aircraft in a very natural way by simply rolling and pitching away from any encountered object. If the only present obstacle is a flat ground, the behaviour will result in a straight and level flight over it. In confined or cluttered environments, the aircraft will constantly steer by rolling and pitching in order to avoid collisions. It is interesting to note that the natural behaviours described in Chap. 3, i.e. terrain following, adapting altitude in case of tail or head wind and achieving smooth landing when decreasing the forward speed, are also byproducts of our control strategy.

Since translating aircraft cannot remain level at all time, the problem of controlling their flight cannot be decoupled into vertical altitude control on the one hand and horizontal steering on the other hand. The ventral part of such an aircraft is not always oriented towards the ground and any turning action requires rolling and pitching, which is fundamentally different from steering with a robot constrained to 2D motion.

Of course the proposed control strategy has some limitations. Leaving aside the limitations related to sensor performance and optic flow detection, situations can still be encountered where hesitations occur due to all proximity signals reaching similar levels. These situations typically arise when perpendicularly approaching a flat surface. However, a work-around exists. By monitoring the sum of all detector signals, one can tell whether an object is really close in front of the aircraft. If this sum reaches a given threshold an emergency action should be taken. For instance, such an action could be to pitch up during a given period of time or to initiate a left/right turn by rolling and pitching. Such a safety action would very much resemble saccadic movements that are so typical in flies [13, 61, 46].

Saccades could be used more widely within the scope of the proposed approach. For instance, a UAV can be controlled so as to fly straight most of the time and enter a saccadic turn-away (lateral or vertical, depending on the closest obstacle) only if the monitored optic flow signals reach a given threshold [67, 5]. Such a saccadic behaviour presents advantages and disadvantages. On the positive side, optic flow extraction is usually more accurate if the aircraft is not rotating all the time and flight efficiency can be improved by flying straight instead of being continuously steering due to far objects producing small amounts of optic flow. However, the drawback of producing tight turns is that optic flow usually cannot be estimated reliably during saccades. Therefore, the aircraft would need to fly blindly during the saccade and may end up facing another obstacle immediately after the end of the manoeuvre, which would induce a second saccade and so on. In addition, it is not trivial to decide on the angle or the duration of the saccade. Also, because of the relatively larger inertia, these turning actions may last significantly longer in UAVs than in flies.

Finally, the presented control strategy is a minimalist one, essentially in terms of vision sensors (number and quality). By increasing the number of optic flow detectors, one could easily improve the robustness and the smoothness of the resulting trajectories. Current work in our lab aims at better understanding the effects of increasing the number of optic flow detectors as well as optimising their orientations all around the flight direction [6]. Another line of research concerns the integration of such a reactive collision-avoidance autopilot with higher level control strategy such as waypoint navigation.

**Acknowledgements** We wish to thank Jean-Daniel Nicoud, Adam Klaptocz and André Guignard for their significant contributions in the development of the hardware and the electronics of the *MC2*. Many thanks also to Tim Stirling for his help in improving this manuscript. This project is funded by EPFL and by the Swiss National Science Foundation, grant number 200021-105545/1.

## References

1. Barber, D., Griffiths, S., McLain, T., Beard, R.: Autonomous landing of miniature aerial vehicles. AIAA Infotech@Aerospace (2005)
2. Barron, J., Fleet, D., Beauchemin, S.: Performance of optical flow techniques. International Journal of Computer Vision **12**(1), 43–77 (1994)
3. Barrows, G., Chahl, J., Srinivasan, M.: Biomimetic visual sensing and flight control. Bristol Conference on UAV Systems (2002)
4. Barrows, G., Neely, C., Miller, K.: Optic flow sensors for MAV navigation. In: T.J. Mueller (ed.) Fixed and Flapping Wing Aerodynamics for Micro Air Vehicle Applications, *Progress in Astronautics and Aeronautics*, vol. 195, pp. 557–574. AIAA (2001)
5. Beyeler, A., Zufferey, J., Floreano, D.: 3D vision-based navigation for indoor microflyers. IEEE International Conference on Robotics and Automation (ICRA'07) (2007)
6. Beyeler, A., Zufferey, J., Floreano, D.: Vision-based control of near-obstacle flight. Autonomous Robots. In Press (2010)
7. Borst, A., Bahde, S.: Spatio-temporal integration of motion. Naturwissenschaften **75**, 265–267 (1988)
8. Borst, A., Egelhaaf, M., Seung, H.S.: Two-dimensional motion perception in flies. Neural Computation **5**(6), 856–868 (1993)
9. Braitenberg, V.: Vehicles – Experiments In Synthetic Psychology. The MIT Press, Cambridge, MA (1984)
10. Camus, T.: Calculating time-to-contact using real-time quantized optical flow. Tech. Rep. 5609, National Institute Of Standards and Technology NISTIR (1995)
11. Chahl, J., Srinivasan, M., Zhang, H.: Landing strategies in honeybees and applications to uninhabited airborne vehicles. The International Journal of Robotics Research **23**(2), 101–110 (2004)

12. Chapman, R.: The Insects: Structure and Function, 4th edn. Cambridge University Press (1998)
13. Collett, T., Land, M.: Visual control of flight behavior in the hoverfly, syritta pipiens. Journal of Comparative Physiology **99**, 1–66 (1975)
14. Coombs, D., Herman, M., Hong, T., Nashman, M.: Real-time obstacle avoidance using central flow divergence and peripheral flow. International Conference on Computer Vision, pp. 276–283 (1995)
15. Duchon, A., Warren, W.H., Kaelbling, L.: Ecological robotics. Adaptive Behavior **6**, 473–507 (1998)
16. Egelhaaf, M., Kern, R., Krapp, H., Kretzberg, J., Kurtz, R., Warzechna, A.: Neural encoding of behaviourally relevant visual-motion information in the fly. Trends in Neurosciences **25**(2), 96–102 (2002)
17. Fermüller, C., Aloimonos, Y.: Primates, bees, and ugvs (unmanned ground vehicles) in motion. In: M. Srinivisan, S. Venkatesh (eds.) From Living Eyes to Seeing Machines, pp. 199–225. Oxford University Press (1997)
18. Floreano, D., Mondada, F.: Automatic creation of an autonomous agent: Genetic evolution of a neural-network driven robot. From Animals to Animats **3**, 421–430 (1994)
19. Franceschini, N., Pichon, J., Blanes, C.: From insect vision to robot vision. Philosophical Transactions of the Royal Society B **337**, 283–294 (1992)
20. Gibson, J.: The Perception of the Visual World. Houghton Mifflin, Boston (1950)
21. Gibson, J.: The Ecological Approach to Visual Perception. Houghton Mifflin, Boston (1979)
22. Green, W., Oh, P., Barrows, G.: Flying insect inspired vision for autonomous aerial robot maneuvers in near-earth environments. Proceeding of the IEEE International Conference on Robotics and Automation, vol. 3, pp. 2347–2352 (2004)
23. Griffiths, S., Saunders, J., Curtis, A., McLain, T., Beard, R.: Obstacle and Terrain Avoidance for Miniature Aerial Vehicles, *Intelligent Systems, Control and Automation: Science and Engineering*, vol. 33, chap. I.7, pp. 213–244. Springer (2007)
24. Hassenstein, B., Reichardt, W.: Systemtheoretische analyse der zeit-, reihenfolgen- und vorzeichenauswertung bei der bewe-gungsperzeption des rüsselkäfers chlorophanus. Zeitschrift für Naturforschung **11b**, 513–524 (1956)
25. Hildreth, E.: The Measurement of Visual Motion. MIT, Cambridge (1984)
26. Horn, B.: Robot vision. MIT Press (1986)
27. Horridge, A.: Insects which turn and look. Endeavour **1**, 7–17 (1977)
28. Hrabar, S., Sukhatme, G.S., Corke, P., Usher, K., Roberts, J.: Combined optic-flow and stereo-based navigation of urban canyons for uav. IEEE International Conference on Intelligent Robots and Systems, pp. 3309–3316. IEEE (2005)
29. Kern, R., van Hateren, J., Egelhaaf, M.: Representation of behaviourally relevant information by blowfly motion-sensitive visual interneurons requires precise compensatory head movements. Journal of Experimental Biology **206**, 1251–1260 (2006)
30. Koenderink, J., van Doorn, A.: Facts on optic flow. Biological Cybernetics **56**, 247–254 (1987)
31. Krapp, H.: Neuronal matched filters for optic flow processing in flying insects. In: M. Lappe (ed.) Neuronal Processing of Optic Flow, pp. 93–120. San Diego: Academic Press (2000)
32. Krapp, H., Hengstenberg, B., Hengstenberg, R.: Dendritic structure and receptive-field organization of optic flow processing interneurons in the fly. Journal of Neurophysiology **79**, 1902–1917 (1998)
33. Krapp, H., Hengstenberg, R.: Estimation of self-motion by optic flow processing in single visual interneurons. Nature **384**, 463–466 (1996)
34. Land, M.: Visual acuity in insects. Annual Review of Entomology **42**, 147–177 (1997)
35. Lichtensteiger, L., Eggenberger, P.: Evolving the morphology of a compound eye on a robot. Proceedings of the Third European Workshop on Advanced Mobile Robots (Eurobot '99), pp. 127–134 (1999)
36. Mallot, H.: Computational Vision: Information Processing in Perception and Visual Behavior. The MIT Press (2000)
37. Marr, D.: Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. W.H. Freeman and Company, New York (1982)
38. Mura, F., Franceschini, N.: Visual control of altitude and speed in a flying agent. From Animals to Animats III, pp. 91–99. MIT Press (1994)
39. Muratet, L., Doncieux, S., Brière, Y., Meyer, J.: A contribution to vision-based autonomous helicopter flight in urban environments. Robotics and Autonomous Systems **50**(4), 195–209 (2005)
40. Nalbach, G.: The halteres of the blowfly calliphora. I. Kinematics and dynamics. Journal of Comparative Physiology A **173**(3), 293–300 (1993)
41. Nelson, R., Aloimonos, Y.: Obstacle avoidance using flow field divergence. IEEE Transactions on Pattern Analysis and Machine Intelligence **11**(10), 1102–1106 (1989)
42. Neumann, T., Bülthoff, H.: Behavior-oriented vision for biomimetic flight control. Proceedings of the EPSRC/BBSRC International Workshop on Biologically Inspired Robotics, pp. 196–203 (2002)
43. Ruffier, F., Franceschini, N.: Optic flow regulation: the key to aircraft automatic guidance. Robotics and Autonomous Systems **50**(4), 177–194 (2005)
44. Santos-Victor, J., Sandini, G., Curotto, F., Garibaldi, S.: Divergent stereo for robot navigation: A step forward to a robotic bee. International Journal of Computer Vision **14**, 159–177 (1995)
45. Scherer, S., Singh, S., Chamberlain, L., Saripalli, S.: Flying fast and low among obstacles. Proceedings of the 2007 IEEE Conference on Robotics and Automation, pp. 2023–2029 (2007)
46. Schilstra, C., van Hateren, J.: Stabilizing gaze in flying blowflies. Nature **395**, 654 (1998)
47. Schuppe, H., Hengstenberg, R.: Optical properties of the ocelli of calliphora erythrocephala and their role in the dorsal light response. Journal of Comparative Physiology A **173**, 143–149 (1993)
48. Serres, J., Ruffier, F., Viollet, S., Franceschini, N.: Toward optic flow regulation for wall-following and centring behaviours. International Journal of Advanced Robotic Systems **3**(27), 147–154 (2006)
49. Sobel, E.: The locust's use of motion parallax to measure distance. Journal of Comparative Physiology A **167**, 579–588 (1990)

50. Sobey, P.: Active navigation with a monocular robot. Biological Cybernetics **71**, 433–440 (1994)
51. Srinivasan, M.: An image-interpolation technique for the computation of optic flow and egomotion. Biological Cybernetics **71**, 401–416 (1994)
52. Srinivasan, M., Chahl, J., Nagle, M., Zhang, S.: Embodying natural vision into machines. In: M. Srinivasan, S. Venkatesh (eds.) From Living Eyes to Seeing Machines, pp. 249–265 (1997)
53. Srinivasan, M., Chahl, J., Weber, K., Venkatesh, S., Zhang, H.: Robot navigation inspired by principles of insect vision. In: A. Zelinsky (ed.) Field and Service Robotics, pp. 12–16. Springer-Verlag (1998)
54. Srinivasan, M., Lehrer, M., Kirchner, W., Zhang, S.: Range perception through apparent image speed in freely-flying honeybees. Visual Neuroscience **6**, 519–535 (1991)
55. Srinivasan, M., Zhang, S., Chahl, J., Barth, E., Venkatesh, S.: How honeybees make grazing landings on flat surfaces. Biological Cybernetics **83**, 171–183 (2000)
56. Stevens, B., Lewis, F.: Aircraft Control and Simulation, 2nd edn. Wiley (2003)
57. Strausfeld, N.: Atlas of an Insect Brain. Springer (1976)
58. Tammero, L., Dickinson, M.: The influence of visual landscape on the free flight behavior of the fruit fly drosophila melanogaster. The Journal of Experimental Biology **205**, 327–343 (2002)
59. Taylor, G., Krapp, H.: Sensory systems and flight stability: What do insects measure and why. Advances in Insect Physiology **34**, 231–316 (2008)
60. Thakoor, S., Morookian, J., Chahl, J., Hine, B., Zornetzer, S.: BEES: Exploring mars with bioinspired technologies. Computer **37**(9), 38–47 (2004)
61. Wagner, H.: Flight performance and visual control of flight of the free-flying housefly (Musca domestica L.). I. organization of the flight motor. Philosophical Transactions of the Royal Society B **312**, 527–551 (1986)
62. Weber, K., Venkatesh, S., Srinivasan, M.: Insect inspired behaviours for the autonomous control of mobile robots. In: M.V. Srinivasan, S. Venkatesh (eds.) From Living Eyes to Seeing Machines, pp. 226–248. Oxford University Press (1997)
63. Wehner, R.: Matched filters - neural models of the external world. Journal of Comparative Physiology A **161**, 511–531 (1987)
64. Whiteside, T., Samuel, G.: Blur zone. Nature **225**, 94–95 (1970)
65. Zeil, J., Boeddeker, N., Hemmi, J.: Vision and the organization of behaviour. Current Biology **18**(8), 320–323 (2008)
66. Zufferey, J.C.: Bio-inspired Flying Robots: Experimental Synthesis of Autonomous Indoor Flyers. EPFL/CRC Press (2008)
67. Zufferey, J.C., Floreano, D.: Fly-inspired visual steering of an ultralight indoor aircraft. IEEE Transactions on Robotics **22**, 137–146 (2006)
68. Zufferey, J.C., Klaptocz, A., Beyeler, A., Nicoud, J.D., Floreano, D.: A 10-gram vision-based flying robot. Advanced Robotics, Journal of the Robotics Society of Japan **21**(14), 1671–1684 (2007)