AIX-MARSEILLE UNIVERSITÉ
FACULTÉ DES SCIENCES DU SPORT DE LUMINY
ED 463 - École Doctorale des Sciences du Mouvement Humain
UMR 7287 - Institut des Sciences du Mouvement
163 Avenue de Luminy, 13009 Marseille, France

Thèse de Doctorat
PhD Thesis

présentée pour obtenir le grade universitaire de docteur en
presented to obtain the grade of philosophical doctor in

Robotique Bio-inspirée
Bio-inspired Robotics

Fabien COLONNIER

Œil Composé Artificiel doté d'Hyperacuité : Applications Robotiques à la Stabilisation et à la Poursuite

Hyperacute Artificial Compound Eye: Robotic Applications to Stabilization and Pursuit

Soutenue le 05/04/2017 devant le jury composé de :

P. VASSEUR	Professeur	Univ. Rouen	Rapporteur
G. SICARD	Ing. recherche-HDR	CEA-LETI Grenoble	Rapporteur
N. MARCHAND	DR CNRS	GIPSA-lab Grenoble	Examinateur
X. SAVATIER	MCF-HDR	ESIGELEC Rouen	Examinateur
G. DE CROON	Prof. assistant	TU Delft (Pays-Bas)	Examinateur
S. VIOLLET	DR CNRS	ISM Marseille	Directeur de thèse
F. RUFFIER	CR-HDR CNRS	ISM Marseille	co-Directeur de thèse

Remerciements

Je remercie particulièrement mes directeurs de thèse Stéphane Viollet et Franck Ruffier pour leur présence et disponibilité tout au long de ma thèse. Grâce à eux, j'ai pu découvrir à la fois le monde des insectes et de la robotique aérienne dans un environnement où j'avais les moyens de travailler.

Je remercie vivement les membres du jury pour avoir accepter d'évaluer mon travail et pour leur présence à la soutenance.

J'ai une pensée pour les permanents de l'équipe biorobotique et je tiens à les remercier chacun pour leur accueil, Thibaut Raharijaona, Julien Serres et Michaël Wiertlewski. J'adresse des remerciements particuliers à Julien Diperi pour sa bonne humeur et à Marc Boyron pour sa disponibilité. Leurs compétences m'ont énormément aidé dans les projets entrepris pendant cette thèse. Je tiens à les remercier vivement pour leur aide et à souligner que travailler avec des personnes aussi compétentes facilite la réalisation de projets ambitieux.

Je remercie chaleureusement les doctorants avec qui j'ai pu travailler : Fabien Expert, Raphaël Juston, Fred Roubieu pour m'avoir bien aidé au début de ma thèse. Augustin Manecy pour tout le travail qu'il a pu réaliser dans l'équipe et dont j'ai pu profité, sans sa contribution et les discussions que nous avons pu avoir, cette thèse aurait été surement différente. Stefano Mafrica, Roman Goulard, Erik Vanhoutte, Léandre Varennes, Valentin Rivière et Julien Dupeyroux pour les discussions passionnées et soirées passées à boire quelques bières! Un merci particulier à Jocelyn Monnoyer pour sa bonne humeur et son canapé!

Je n'oublie pas également les différents post-doc, ingénieurs et stagiaires qui sont venus dans l'équipe, qui m'ont aidé de près ou de loin, les moments partagés en dehors du laboratoire avec certains : Bruno Boisseau, Paul Mignon, Lorris Dola, Estelle Loret, Vincent Trichon, Sylvain Durand-Chamontin, Alexandre Desbiez, Soledad Ramirez-Martinez, Hoby Rabenarivo, Théo Louiset, Anne Nespoulous, Anthony Millot, Victor Boutin, Rodolphe Mawonou.

Je remercie également l'Agence Nationale de la Recherche qui a permis le financement de ma thèse dans le cadre du projet IRIS. Je tiens à remercier tous les membres du projet avec lesquels j'ai eu le plaisir de collaborer.

Je tiens à remercier tous ceux qui m'ont soutenus ou bien qui m'ont permis de décompresser. Je pense bien sûr en premier lieu aux doctorants de l'équipe et aux doctorants du laboratoire avec mention spéciale pour à Emmanuelle Lefêvre, Anne-Sophie Zenses, Caroline Pin-Barre, Sophie Le Cann, Mathieu Lecoq, Martin Bossard et Niek Benerink. Je pense aussi à mes colocs, Guillaume Gras, Anne-Sophie Bonnefoi et Ludovic Foucher, à mes amis, Valentin Magnier, Antoine Tan-Kim, Thibaut Courtain, Laurent Boumard, Florian Pohu et Alexis Dupont.

Je finirai par ma famille qui m'a soutenu dans cette aventure, surtout dans les derniers moments.

Abstract

Based on several studies, the fly retina is submitted to periodic micro-movements. Several sensors were designed in the lab mimicking this principle. It was therefore established that this vibration could be used in the localization of contrasts with precision and endowed the fly vision with hyperacuity. Inspired by the fly compound eye properties, such as the Angular sensitivity and the periodic scanning, the sensors were able to localize a contrast very precisely over a small field of view limited to only two pixels.

In this thesis, an artificial compound eye endowed with a wide field of view was used. First, an algorithm that fused the local position measurements of different photosensor (pixel + lens) pairs is proposed. It enables a robot named HyperRob to hover above a naturally textured pattern.

Localizing a contrast precisely over the entire field of view remains difficult with this first solution. But, using 2 pairs of photosensors, a second algorithm allows having, in the case of a bar, a more linear position measurement and its subtended angle too. A calibration process was involved to have a map of the pair measurements relative to the angular position and subtended angle of the bar. It showed some good results, especially in steady conditions, but also a dependency on the contrasts seen and the illuminance in respect to the calibration setup.

Therefore, an effort was done in order to avoid a calibration process. A third algorithm was suggested using previous works of Heiligenberg and Baldi. They established that an array of Gaussian receptive field can provide a linear estimation of a stimulus position, thanks to a weighted sum calculation. Here, this approximation is modified to be robust to ambient lighting and contrast variations. An application to a target pursuit was made with a mobile robot named ACEbot. It was able to reproduce pursuit behavior similar to the hoverfly. An interception behavior was also showed.

Finally, an artificial compound eye with a coarse spatial resolution can be endowed with hyperactuity and enable a robot to follow a target with precision. In this thesis, a step forward has been made toward bio-inspired target localization and pursuit, allowing a better understanding of the strategy used by winged insects.

List of Publications

Journal Papers

F. Colonnier, S. Ramírez-Martinez, S. Viollet and F. Ruffier, “Visual Micro-Scanning Makes a Sighted Robot Capable of Target Following Behavior”, *In preparation* [Colonnier et al., 2017]

F. Colonnier, A. Manecy. R. Juston, H. Mallot, R. Leitel, D. Floreano and S. Viollet, (2015). “A small-scale hyperacute compound eye featuring active eye tremor: application to visual stabilization, target tracking, and short-range odometry”, *Bioinspiration & Biomimetics* [Colonnier et al., 2015a]

S. Viollet, S. Godiot, R. Leitel, W. Buss, P. Breugnon, M. Menouni, R. Juston, F. Expert, **F. Colonnier**, G. L’Eplattenier, A. Brückner, F. Kraze, H. Mallot, N. Franceschini, R. Pericet-Camara, F. Ruffier, and D. Floreano, (2014). “Hardware Architecture and Cutting-Edge Assembly Process of a Tiny Curved Compound Eye”, *Sensors* [Viollet et al., 2014]

F.L. Roubieu, J.R. Serres, **F. Colonnier**, N. Franceschini, S. Viollet and F. Ruffier, (2014). “A biomimetic vision-based hovercraft accounts for bees’ complex behaviour in various corridors”, *Bioinspiration & Biomimetics* [Roubieu et al., 2014]

Patent

S. Viollet, **F. Colonnier** and E. Vanhoutte, “Système de mesure de la distance d’un obstacle par flux optique”, *French patent* (pending n^o 16 59663)

Peer-Reviewed Proceedings

F. Colonnier, A. Manecy, R. Juston, S. Viollet, “Visual Odometry and Low Optic Flow Measurement by Means of a Vibrating Artificial Compound Eye”, pages 153-163. Biomimetic and Biohybrid Systems: 4th International Conference, Living Machines 2015, Barcelona, Spain, July 28 - 31, 2015, Proceedings *Springer International Publishing* [Colonnier et al., 2015b]

Valentino Braitenberg award - Best talk prize

Book chapter

F. Colonnier, A. Manecy, R. Juston, S. Viollet, "Visual Odometry and Low Optic Flow Measurement by Means of a Vibrating Artificial Compound Eye", pages 153-163. Biomimetic and Biohybrid Systems: 4th International Conference, Living Machines 2015, Barcelona, Spain, July 28 - 31, 2015, Proceedings *Springer International Publishing* [Colonnier et al., 2015b]

Workshops

A. Manecy, E. Vanhoutte, S. Mafrica, **F. Colonnier**, F. Ruffier, N. Marchand, S. Viollet, "X4-MaG and RT-MaG: a low-cost open-source micro-quadrotor based on Real-Time Linux and a new Matlab/Simulink toolbox", *IROS Aerial Open Source Robotics Workshop*, 2015

F. Colonnier, A. Manecy, R. Juston, S. Viollet, "Hyperacuity: Application to visual stabilization", *IROS Vision-Based Control and Navigation of Small, Light-Weight UAVs Workshop*, 2015

Other communications

F. Colonnier, S. Viollet, "Mecanum Wheel Robot : Presentation and application", TechDays 2016, 4ème journées technologiques Robotex, IRCCyN, Nantes (France), September 2016.

F. Colonnier, "Demonstration of CurvACE (Curved Artificial Compound Eye)", 2nd Workshop on Multi Unmanned Vehicles Systems, Compiègne (France), July 2014

F. Colonnier, "Demonstration of CurvACE (Curved Artificial Compound Eye)", 2nd Workshop on Research, Education and Development of UAS, Compiègne (France), November 2013

A. Manecy, **F. Colonnier**, "Présentation : Présentation de la plateforme ROBOTEX : Arène de vol", TechDays 2013, 1ère journées technologiques Robotex, LAAS, Toulouse (France), July 2013.

A. Manecy, G. Sanahuja, J. Dumon, F. Elisei, **F. Colonnier**, "Session thématique : Systèmes de capture du mouvement", TechDays 2013, 1ère journées technologiques Robotex, LAAS, Toulouse (France), July 2013.

Contents

5 Bio-inspired behaviors: Target following, Target detection and interception 107

6 Conclusion 131

Appendices 147

Chapter 1

Introduction

Contents

1.1 Technological Context

Today, the technology of the Unmanned Aerial Vehicle (UAV) and mobile robots is more and more present in our lives. They were first deployed in military applications. These UAVs are mostly long-range aerial robots with a wide wingspan that are remotely controlled from a distant location. These systems become more and more autonomous (see figure 1.1b).

Other UAVs have appeared for leisure activities in the recent years. Mostly quadrotors were used by the hobbyist community. FPV (First Person View) racing appeared with the use of virtual reality headsets and small cameras embedded onboard the flying robots. All these different uses of flying machine decreased the cost of robots and opened new markets with autonomous systems that required less and less piloting skills. Companies like DJI have developed a know-how in aerial shootings. These kinds of robots can also be involved in visual inspections or with photogrammetry techniques in the generation of 3D digital models.

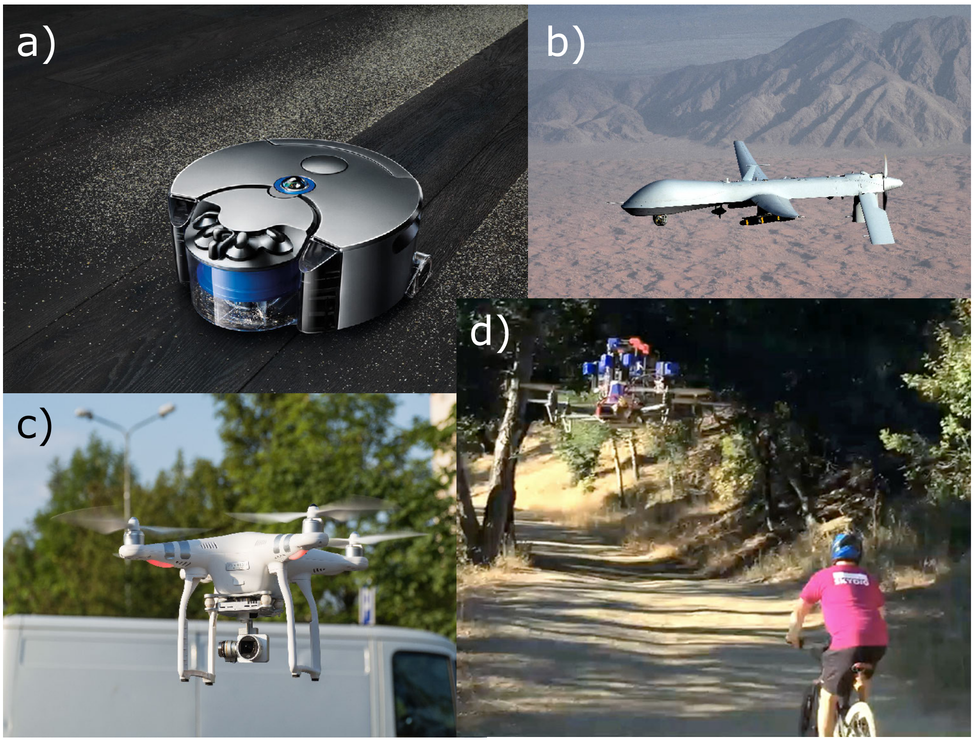


Figure 1.1: a) The Dyson 360eye autonomous vacuum cleaner (source from Dyson official website) b) An unmanned aircraft, the MQ-1B Predator from General Atomics (Picture from the U.S. Air Force, wikimedia commons) c) A quadrotor with a pantilt camera, the DJI Phantom 3 advanced (Picture from Jacek Halicki, wikimedia commons). d) Picture of the flying robot from the startup Skydio performing sportsman following and trees avoidance (image: Skydio via YouTube `https://youtu.be/OTOjO4JipOg`).

The guidance and navigation of the aerial robot is largely based on GPS measurements. Indeed, it provides a global localization but does not take the environment into account. In GPS-denied areas like indoors, it cannot be relied upon. Moreover, a solution to avoid obstacles is necessary in closed environments. Therefore, cameras are embedded onboard and visual feedback is used in autonomous flying robots.

The robot vacuum cleaner is an example of mobile robots which have to deal with an unknown environment. They are using either contact sensors or, more recently, an omnidirectional camera to perform visual SLAM (Simultaneous Localization And Mapping) (see figure 1.1a).

In flying robotics, the "follow me" mode is becoming a standard on consumer robots such as DJI Phantom 3 and Parrot Bebop 2. But the awareness of the surroundings is still challenging. Skydio is a startup which looks at this keypoint and shows an example of sportsman following with obstacle avoidance (see figure 1.1). These solutions, with their onboard sensors and processors, are usually heavy and have a limited autonomy. The camera sensor with their low frame rate are a limitation to aggressive maneuvers and the computation of the full frame image in real-time requires powerful processors.

Technology can help to solve this problem by using a more powerful microprocessor and new batteries. But another point of view could be to see how nature has solved through evolution many of these current problems and be inspired by it. Trying to reproduce animal behaviors and characteristics with robots is also an interesting approach as it could provide feedback to better understands the world of living organisms.

1.2 Biological findings in flying insects sensorimotor system

If the goal is to design small aerial vehicles capable of stabilizing themselves and navigating through an unknown environment while avoiding obstacles, insects are a good example of tiny creatures capable of such behaviors. It should be noted that they display more complex performances, like target following before mating, or chasing prey as in the case of the dragonfly. Bees are able to find food, come back to the hive and communicate the food site to the other members of the community (thanks to "waggle dance" [von Frisch, 1967]). Taking inspiration from the flies and other living forms can provide other solutions to perform the same tasks.

1.2.1 Anatomy of the flies

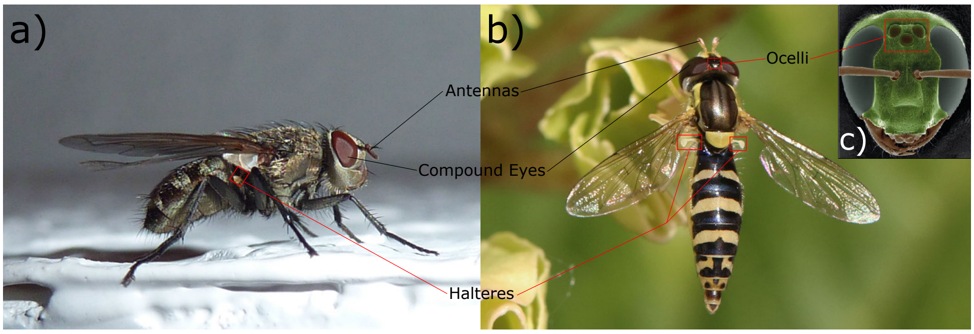


Figure 1.2: a) The housefly *Musca domestica* endowed with compound eyes, halteres, antennas and ocelli not visible on the picture (source Wikimedia commons) b) Long hoverfly, *Sphaerophoria scripta*, displays its big compound eyes and halteres (Picture: Charles J. Sharp, Wikimedia commons) c) Nocturnal bee *Megalopta genalis* with its three ocelli on the dorsal surface of the head [Cronin et al., 2014])

Flying insects are endowed with a specific sensory system to enable a skilled flight (see figure 1.2 and [Taylor and Krapp, 2007] for a review). First, visual information is perceived thanks to the compound eyes. The primary cue processes the optic flow (see appendix A), which is the apparent motion of features in the observer field of view. The compound eyes are also used to detect the horizon. These output measurements are useful to stabilize the flight. The ocelli are placed on the dorsal surface of the head and they are usually three. Like the compound eyes, they are sensitive to light but with a higher dynamic range and a wide field of view about 150° [Land and Nilsson, 2012]. They appear to contribute to the dorsal light response, which is involved in head stabilization [Hengstenberg, 1993]. Halters mediate flight stabilization in Dipterans and Strepsipterans : their oscillations make them sensitive to the Coriolis force thus providing body's mechanical angular speed sensitivity to these insects. They have evolved from old hind wings to fulfill the inertial sensing functionality. They are oscillating in opposite phase of the wing oscillation and the Coriolis force applied on the knob at the end of each halter is sensed. The antennae, as the cils in the head of the fly, seem to be sensitive to the airflow acceleration and speed. This air speed sensitivity helps to control their air speed, in a similar way the Pitot tubes can do on planes.

1.2.2 The compound eyes of insects

The insect's compound eyes are an example of evolution of vision (see [Nilsson, 2009] for a philogenetic tree). They are the most common eye type in the animal kingdom [Cronin et al., 2014]. The part of the fly's brain dedicated to visual information processing is about two-thirds of the 360000 neurons (for the *Musca* and *Calliphora*) [Strausfeld, 1976], which indicates the importance given to visual inputs.

The architecture of the compound eyes is presented in figure 1.3. A compound eye is composed of multiple ommatidia, knowing that an ommatidium is composed of a hexagonal lens (this is the most common shape observed but it can vary along with the position on the head), a crystalline cone and a rhabdom. The rhabdom is separated into several rhabdomeres (usually 7 or 8) surrounded by pigment cells (see figure 1.3b). By migrating toward the center of the rhabdom, the pigment cells are absorbing more or less light, which create a first light adaptation mechanism [Kirschfeld and Franceschini, 1969].

Two main optical parameters are used to characterize these photoreceptors: the interommatidial angle $\Delta\varphi$, which is the angle between 2 photoreceptors' directions and the acceptance angle $\Delta\rho$, defined as the angle at half width of the Gaussian-shaped Angular Sensitivity Function (ASF). This particular shape of ASF results from the combination of the airy diffraction pattern and the geometrical angular width of the rhabdom at the nodal point of the lens [Snyder, 1979, Land and Nilsson, 2012]. It realizes a first spatial filter of the visual scene.

The spatial resolution of compound eyes is relatively coarse compared to the human eye (with an interreceptor angle of 0.009° in the fovea [Cronin et al., 2014]) and various between the different species. The interommatidial angle can be as small as 0.24° for the dragonfly (*Anax junius*), up to 7° for the Coleoptera (*i.e. Phyllobius urticae*) [Land, 1997]. It means that this feature is perhaps not the most important for visually-guided flight.

The compound eyes usually present an anisotropic spatial resolution. It usually reveals that the insect's eyes contain an acute zone similar to the human eye fovea. The difference between male and female demonstrates an evolution toward specification. Indeed, the male, which has the duty to pursue the female for mating, usually has a distinctive area in the center of its head called "love spot" which has a higher resolution. It is possible that the left and right part can touch, while staying separated in their female counterpart [Perry and Desplan, 2016].

There are two types of compound eyes (see figure 1.3). The appositition eyes and the superposition eyes. In the first, the light that enters through a lens is guided to a unique rhabdom. In the superposition eyes, the ommatidium are not separated with pigment cells, allowing the presence of a clear zone between the crystalline cones and the rhabdoms. It enables light from different lenses but with the same angular direction to target one rhabdom. The apposition eyes is mostly present in diurnal insects, as the superposition eyes are for the nocturnal ones [Cronin et al., 2014].

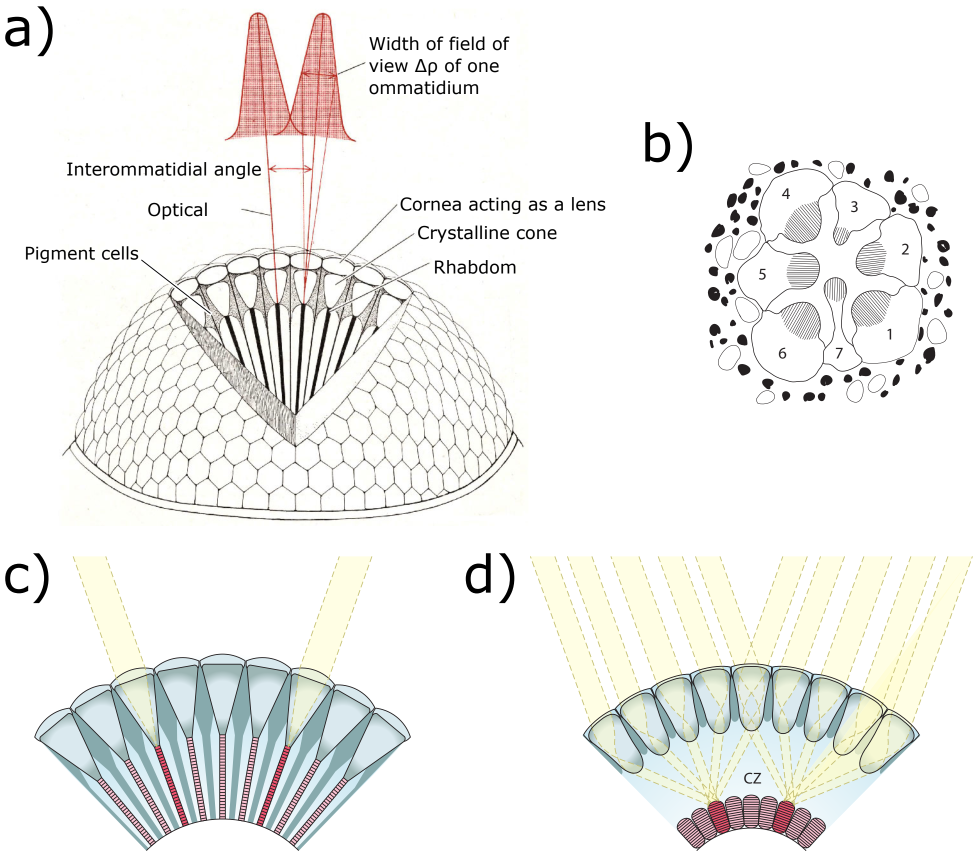


Figure 1.3: a) Schematic view of a compound eye, showing the two main optical parameters of interest: the interommatidial angle $\Delta\varphi$, defined as the angle between optical axes of two adjacent ommatidia, and the acceptance angle $\Delta\rho$ (adapted from [Horridge, 1977]). b) Schematic transverse section through the open rhabdom of a fly, showing the seven distal retinular cells (hatched in black) with their separated rhabdomeres. c-d) The two broad subtypes of compound eyes. c) Apposition eyes (in this case a focal apposition eye). d) Superposition eyes (in this case a refracting superposition eye). cz = clear zone. (subfigures b-d are reprinted [Cronin et al., 2014])

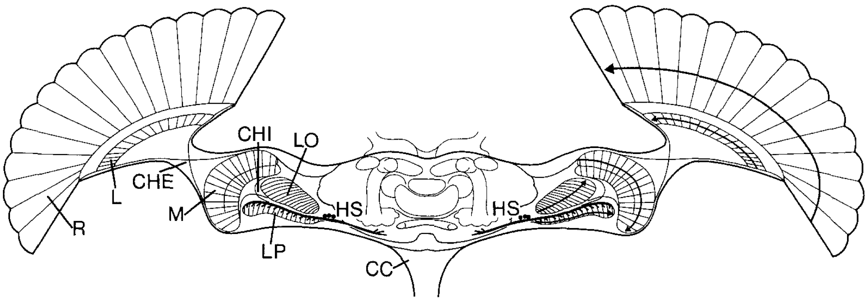


Figure 1.4: Schematic horizontal section through the compound eyes, the optic lobes and the protocerebrum of the fly. Each optic lobe contains the visual neuropils Lamina (L), Medulla (M), Lobula (LO), and Lobula plate (LP), which are connected by the external and internal chiasma (CHE, CHI). The retina (R) and the visual neuropils are organized in columns, the orientation of which are indicated by thin lines. Visual information is conveyed in strict retinotopic order from the retina onto the successive visual neuropils (see arrows in the right eye and optic lobe). The horizontal system (HS) consists of three giant tangential cells residing at the anterior surface of the lobula plate and projecting into the posterior ventrolateral protocerebrum near the cervical connective (CC). (Reprinted from [Hausen, 1982])

1.2.3 Neural pathways

The ommatidia of the fly are connected to 4 neuropiles: the lamina, the medulla, the lobula and the lobula plate (lobulas can be referenced as lobulla complex). The six light-sensitive photoreceptors, R1-R6, are directly connected to the lamina, when the two other photoreceptor neurons, R7 and R8, pass through the lamina to end in specific layers of the medulla [Shaw, 1981].

First, the lamina is in charge of a first high pass temporal filtering. It is where the information about the background illuminance is removed [Tuthill et al., 2013].

The medulla is considered to be the place where the first steps of the motion processing are made. Due to the small size of its neurons, it is difficult to perform physiological recordings [Douglass and Strausfeld, 1996].

The lobula perfoms local motion detection and the lobula plate the integration of all the directonal motion information. The Lobula Plate Tangential Cells (LPTCs) are in charge of this processing and provide a wide-field motion estimation (see [Taylor and Krapp, 2007]. It was observed that the LPTCs are directionally selective (front to back, back to front, left to right and right to left) [Hausen, 1982] and that parallel pathways for ON and OFF motion detection exist [Franceschini et al., 1989,Maisak et al., 2013].

1.2.4 Retinal movements

A - In the fly compound eyes

Observations showed the presence of small muscles in the flies' head which enables small movements of the rhabdomeres [Hengstenberg, 1972]. Figure 1.5 shows the position of 2 muscles MOT and MOS able to translate the retina and therefore modify the direction of vision of the ommatidia. By measuring the activity of the MOT muscle, movements of $5 - 6Hz$ has been recorded during flight [Franceschini and Chagneux, 1997] with an amplitude close to $\Delta\varphi$ (see figure 1.6).

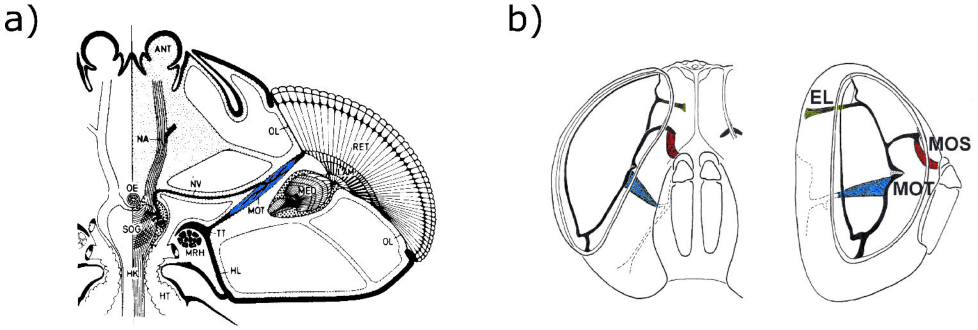


Figure 1.5: a) Anatomical reconstruction of the eye muscle and its motoneurone of a *Musca domestica* (adapted from [Hengstenberg, 1972]). b) Frontal and sagital sections of the same fly. It shows the muscles MOS and MOT which enables a translation of the retina (from [Kerhuel, 2009], initially published in [Franceschini et al., 1991])
Keys: MOT, muscle orbito-tentorialis, MOS: muscle orbito-scapalis, NV: motor nerve to MOT, TT: Tentorium, RET: ommatidia, LAM: Lamina, ANT: antenna base, OE: esophagus

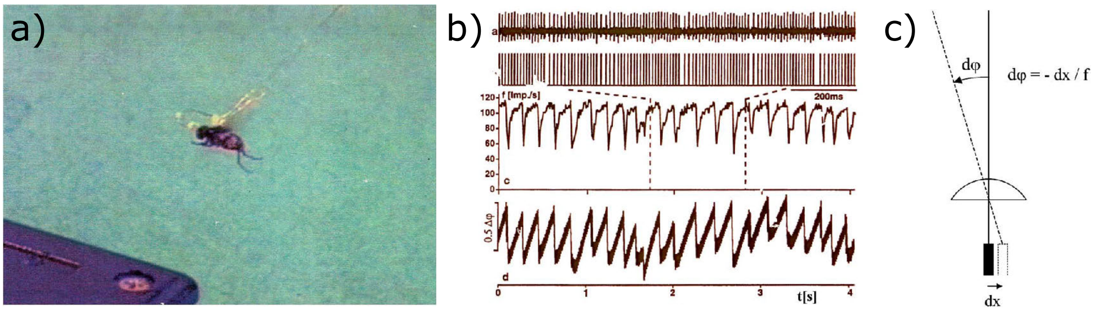


Figure 1.6: a) Flying housefly on the leash. The insect was flying freely at the tip of a double enameled copper wire (20-μm diameter, 0.4-m length) serving as a differential electrode (picture taken from a video film). (ba) In-flight recording of the spikes generated by the MOT eye muscle. (bb) Unit pulses triggered by the individual spikes showing the periodic dropouts. (bc) Instantaneous spike frequency plot showing that about five times per second, the high frequency (approximately $110Hz$) of the MOT spontaneously dropped by about half. (bd) and (c) The periodic dip in the firing rate caused periodic displacements dx of the photoreceptors in the focal plane of each facet lenslet, resulting in periodic angular displacements $d\varphi$ of the visual axes. (Reprinted from [Franceschini, 2014])

Their function has not yet been completely understood. A plausible explanation of these micro-movements could be to endow the eye with hyperacuity, as has been shown with robotic applications (see [Viollet et al., 2014] for a review).

B - Observation in the jumping spiders

The visual system of the jumping spiders is very different from what has been described up until here (see figure 1.7). It is composed of secondary eyes which are fixed to the carapace and are in charge of detecting the motion of a potential prey or threat. Then, the spider turns toward the target to enable the primary eyes to locate and identify it. These primary eyes have a narrow retina shaped like boomerangs [Land and Nilsson, 2012]. It has been observed that this retina is able to move horizontally and vertically by as much as 50°, and also rotate about the optic axis (torsion)(see Land 1985). When presented with a novel target, the eyes scan it in a stereotyped way moving slowly from side-to-side at speeds between 3 and $10°.s^{-1}$, and rotating through ±25° as they do so [Land, 1969].

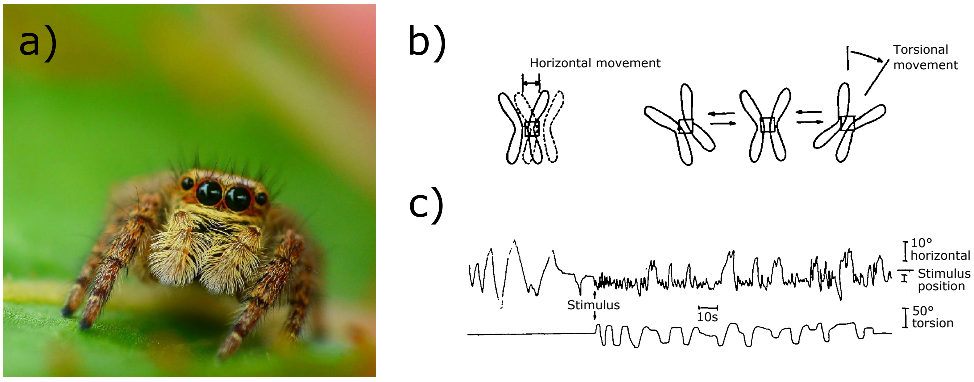


Figure 1.7: a) Picture of a jumping spider (source flickr coniferconifer) b) A diagram and record of the movements of the boomerang-shaped retinae of the two principal eyes while scanning a novel target. These movements are conjugate, and consist of a stereotyped pattern of horizontal oscillations and slower torsional rotations. This scanning pattern apparently allows the narrow retinae to determine the angular pattern of edges in the target, and thus enables the spider to distinguish other jumping spiders from potential prey (modified from [Land, 1969]

C - Ocular movements of the human eye

The human eye is in the camera eye category, where the retina is at the back of an ocular globe and the light enters through a unique lens. The information is transmitted to the brain through the optical nerve. The ocular movements can be classified as follows. There are the smooth pursuit movements to track a slow target, in order to keep it in the highest spatial resolution area of the retina, the fovea. The saccades are quick movements to change the attention area from one to another. Some micro-movements also occur when the eye is steady (see [Rolfs, 2009] for a review). Three types of Fixational eye movements can be observed:

- the drift
- the tremor
- the microsaccades

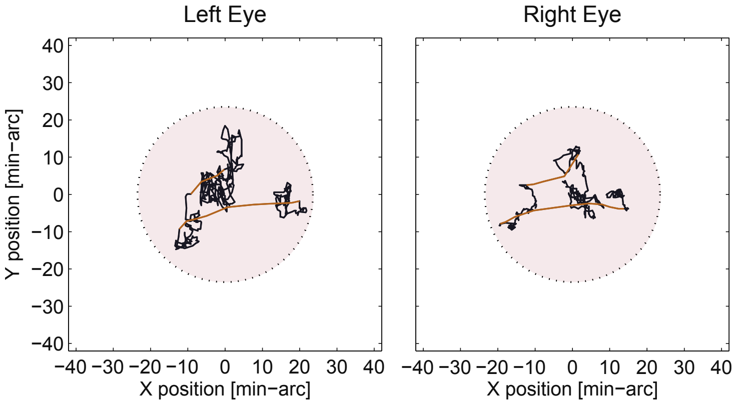


Figure 1.8: Trajectory of the two eyes during $1s$ of fixation on a black spot on a neutral background (highlighted by red dotted circle). Slow drift movements are displayed in dark color while rapid jerk-like microsaccades are highlighted in orange. (from [Rolfs, 2009])

Figure 1.8 shows drifts, which are smooth erratic movements at low velocity in black, and microsaccades,which are rapid shifts of the eye, in orange. The microsaccades have the particularity of being correlated on both eyes. The tremor (or physiological nystagmus) is a wave-like motion superimposed to the drift movements. Studies reports movement cycles of 0.1-0.5 min-arc in amplitude with frequencies ranging from 30 to $103Hz$.

1.2.5 Head-body reflex in roll in flying insects

The halteres mentioned previously (see section 1.2.1) is a sensor which plays an important role in the flight controller of the flies. It provides measurements of fast rotations, those that vision can not cope with [Hengstenberg, 1988]. It is involved in the head stabilization. This stabilization is very interesting when it comes to visual processing as it cancels rotational movements. The vision is therefore only submitted to translation movements. It is very helpful in the case of flight regulation thanks to optic flow. Indeed, the translational optic flow is directly linked to the speed and distance of the features seen, whereas the rotational optic flow is only dependent on the individual rotational motion. Head-roll compensation has also been reported on the wasps with a uniform visual environment, suggesting the use of rotational measurements used as input [Viollet and Zeil, 2013].

It is similar to the vestibular system in humans which the rotation of the head and provide feedback to eye in order to stabilize the vision. This mechanism is known as the Vestibulo Ocular Reflex (VOR). The VOR is very interesting when it comes to target tracking. It creates an inner loop controller which helps in fixation tasks by rejecting perturbations.

1.3 Bio-inspired visual sensors and their application on robotic platforms

As it has been seen, the compound eyes show interesting features, such as compactness and optical properties. However, it is very hard to mimic with technological processes. Only 2 recent examples of artificial compound eyes have been built. The first one is a hemispherical array of 180 lens and diodes and is capable of recording images [Song et al., 2013]. The second one is a cylindrical compound eye with a VLSI circuit capable of light adaptation [Floreano et al., 2013], especially designed for optic flow measurements. As most of the experiments performed in this thesis were with this sensor, it will be further detailed in section 1.3.4. But some others examples inspired by the fly eyes were designed to measure optic flow or perform target tracking.

1.3.1 Works on Optic flow developed in the laboratory

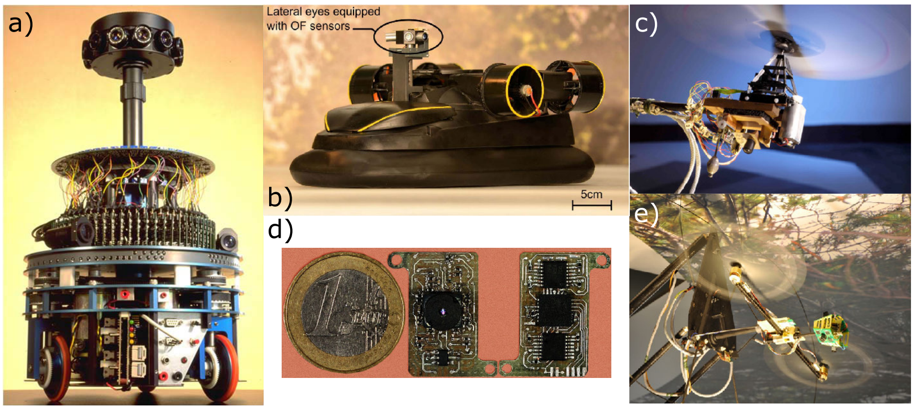


Figure 1.9: a) The "Robot mouche" was the first bio-inspired robot and was able to perform obstacle avoidance thanks to Elementary Motion Detectors. (Reprinted from [Franceschini et al., 1992]) b) The hovercraft named LORA was able to reproduce bees' trajectories navigating in a corridor thanks to 2 optic flow measurements [Roubieu et al., 2014] c) OCTAVE was the first robot to perform ventral optic flow regulation, in order to assess altitude and speed. (Copyright H. Raguet) d) Local Motion Sensor (LMS) with lens, photodiodes and a microcontroller to compute the optic flow using two different pathways for ON and OFF contrasts. (Adapted from [Roubieu et al., 2012]) e) BeeRotor was a tethered robot endowed with a decoupled eye and able to navigate into a tunnel. It stabilizes its flight with 4 LMS and reorients its eyes to look perpendicularly to the ground and is therefore able to avoid steep relief [Expert and Ruffier, 2015].

The first example of a bio-inspired motion detector was directly embedded on a robot. The "Robot-Mouche" (literally robot-fly, see figure 1.9a) was endowed with 114 Elementary Motion Detectors (EMD) spread horizontally around the robot. Each are implemented on parallel analog electronic circuits, which matches the fly's neural parallel processing and avoids any quantization and sampling of digital devices. It was the first implementation of EMD based on the "time-of-travel' scheme, which is a process that measures the time delay between 2 filtered photodiode signals crossing a threshold [Blanes, 1986]. The "Robot-Mouche" was able to navigate to a goal and avoid obstacles along the way. The strategy was to move straight at $50cm.s^{-1}$ until an obstacle is detected, then stop to perform rotation toward the closest gap and repeat until the goal is reached [Franceschini et al., 1992].

Then, optic flow algorithms were implemented on microcontrollers to reduce the size. First, it was only composed of two photodiodes behind a lens with a first analog filtering step before the computation of time delay on the microcontroller [Ruffier et al., 2003]. Then, improvements were made by involving more photodiodes in the processing to increase the robustness [Roubieu et al., 2012]. Figure 1.9d shows the Local Motion Sensor algorithm which was duplicated 5 times and fused through a median filter. Later, an adaptation allowed for reduction of the computational load [Expert et al., 2012]. A novel retina endowed with auto-adaptive pixels that obey the Michaelis-Menten law, called M^2APix was developed to perform optic flow measurements. It provides a light adaptation through 7-decade range while assessing transient changes accurately [Mafrica et al., 2015].

These optic flow sensors have demonstrated good performance and were applied on robots to test bio-inspired autopilots (see figure 1.9). Indeed, it has been established that knowledge of neither airspeed nor ground speed, nor altitude is needed to perform terrain following. Only the ventral optic flow measurements could provide sufficient information to perform an altitude control [Ruffier and Franceschini, 2005].

Behavioral studies of bee navigation through a corridor [Srinivasan et al., 1991, Srinivasan et al., 1996, Serres et al., 2008] highlight the fact that an optic flow regulation could explain the bee trajectories. An autopilot called LORA (stands for Lateral Optic flow Regulation Autopilot) was proposed using 2 optic flow sensors oriented towards left and right. The strategy is to assess the maximum optic flow between left and right to a setpoint which regulates the distance to the nearest wall. The speed is however controlled with the sum of the left and right optic flows. LORA was tested onboard a hovercraft robotic platform and reproduced most of the insect flight behaviors [Roubieu et al., 2014] (see figure 1.10 for details).

A similar behavior was observed on the vertical axis, using the dorsal and ventral otpic flow as an input of the controller which regulates the altitude [Portelli et al., 2010]. Robotic experiments were able to reproduce such a behavior and perform steep relief avoidance using a rotated eye [Expert and Ruffier, 2015].

BioCarBot was a car-like robot which was able to assess its trajectory thanks to an optic flow feedback loop and perform odometry. The optic flow was computed thanks to a cross-correlation based algorithm with the M^2APix retina which enables constant performances over a wide illuminance range [Mafrica et al., 2016].

1.3.2 Optic flow in robotic applications

Other ways of implementing optic flow measurements have also been explored. The first bio-inspired algorithm proposed was the local 1-D Hassenstein-Reichardt correlator [Hassenstein and Reichardt, 1956], but very few robotic implementations were made, because of its limited range and lack of a scaled output.

Optic flow measurements are also used in computer vision to perform motion analysis and compute transformation between two images. As camera is used as input, they are adapted to work on pixels' array. The popular algorithms are the Interpolation Image Algorithm (I2A)

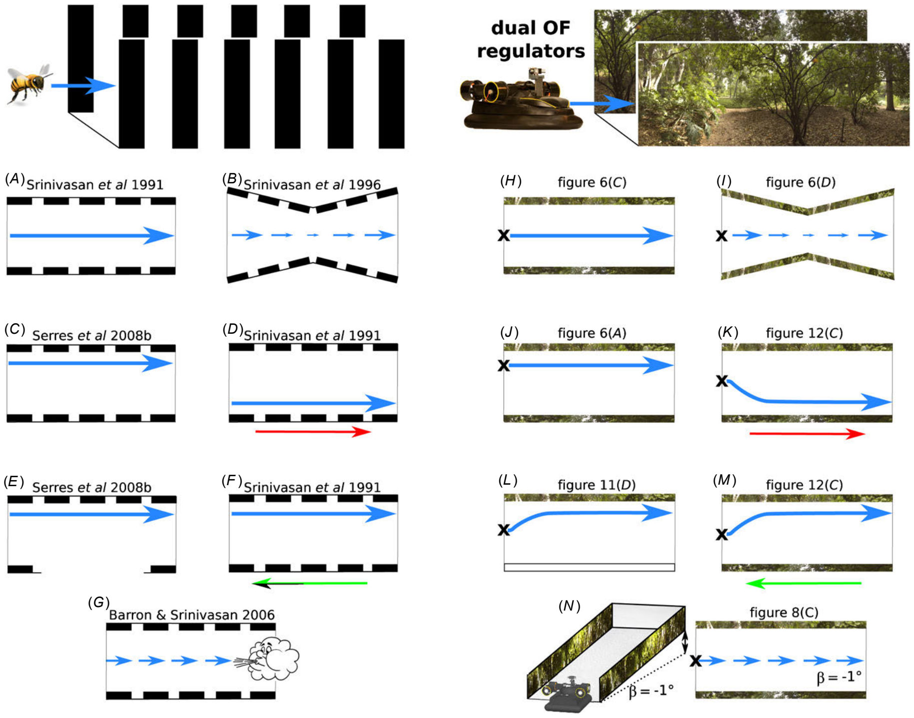


Figure 1.10: Summary of the similarities of the bee behaviors observed in the last 25 years ethological studies (A)-(G) and the fully actuated robot behaviors equipped with the bee-inspired LORA autopilot tested in similar situations (H)-(N). These results show the dual OF regulator combined with a heading-lock system can be viewed as a working hypothesis to explain how the honeybee *Apis Mellifera* controls both its speed and position on the horizontal plane of a constraint corridor. Blue arrows represent the main direction of the agent moving in a corridor where the entrance is represented by a cross. Red and green arrows represent the direction of motion of the moving wall in case of a non-stationary corridor. The last comparison (G) versus (N) shows how constant speed behaviour despite head wind may be explained by the presented model.
The references above each subfigure are [Srinivasan et al., 1991] for A, D and F, [Srinivasan et al., 1996] for B, [Serres et al., 2008] for C and E, [Barron and Srinivasan, 2006] for G. For H to N, it is the figure number of the reference [Roubieu et al., 2014], where this figure is coming from.

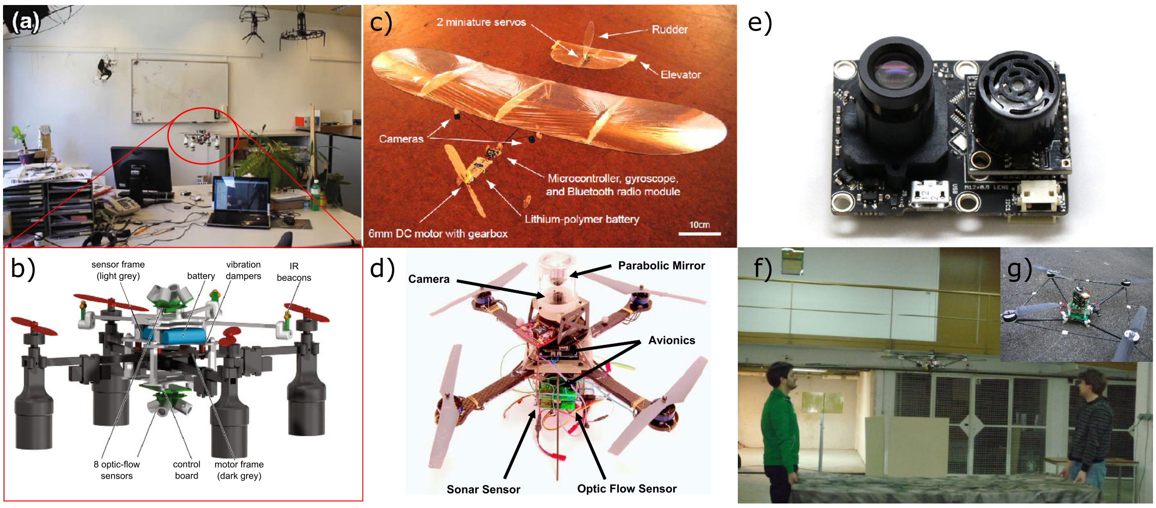


Figure 1.11: a-b) Quadrotor equipped with 8 Optic flow sensors capable of hovering in 3D dimension with low drift (adapted from [Briod et al., 2016]). c) 30-g aircraft with its electronics was able to perform obstacle avoidance (Reprinted from [Zufferey and Floreano, 2006]) d) Quadrotor based on the Hummingbird platform from Ascending Techologies endowed with a panoramic camera on top and optic flow sensor with a sonar on the bottom (Reprinted from [Conroy et al., 2009]). e) The PX4FLOW optical flow sensor estimating the optic flow using block matching with mounted lens on the left and ultra sonic distance sensor on the right. The distance sensor can be used to scale the measured optic flow which is derotated based on the embedded rate gyro (Reprinted from [Honegger et al., 2013]). f-g) Quadrotor with a camera pointing downward performs hovering and landing above a moving platform thanks to Optic Flow (adapted from [Herissé et al., 2012]).

[Srinivasan, 1994], the Horn-Schunk [Horn and Schunck, 1981] and the Lucas Kanade [Lucas et al., 1981], but its pyramidal implementation [Bouguet, 2001] is the most widespread in robotic application due to its accuracy and efficiency in regard to the computational load. An improved version with specific feature detection named LKT was proposed [Tomasi and Kanade, 1992].

Several studies use the optic flow cues to perform different tasks with more or less bio-inspired principles. Navigation through a corridor was early a very interesting task to test optic flow based controller. First, a mobile robot computed optic flow using a gradient-based intensity algorithm to reproduced the centering-reflex [Santos-Victor and Sandini, 1997], suggested by [Srinivasan et al., 1996]. Others performed the same task. Conroy *et al.* used a Wide-Field integration algorithm, previously presented for a hovering task [Humbert et al., 2007], to provide an usable output for navigation from local optic flow measurements [Conroy et al., 2009]. Zingg *et al.* used a LKT algorithm computed at $20Hz$ [Zingg et al., 2010].

The first flying robot to perform obstacle avoidance thanks to optic flow was using the I2A algorithm onboard a lightweight fixed-wing robot [Zufferey and Floreano, 2006]. Autopilots were designed for hovering using optic flow measurements thanks to mouse sensors pointing toward the ground only [Lim et al., 2012] or toward every directions [Briod et al., 2016]. Stabilization thanks to optic flow on different surfaces often uses a camera and a Lucas-Kanade Pyramidal (LKP) computation [Bristeau et al., 2011, Herissé et al., 2012]. With a specific optic flow sensor and a non-linear observer, it has also been shown that it is possible to stabilize a quadrotor [Conroy and Humbert, 2013].

The integration of the optic flow calculation is also used for odometry task in combination with sonar measurements. Carillo *et al.* used this strategy in a line following application when none is detected in the FOV [Carrillo et al., 2012]. Specific sensors were also developed as the Pix4flow sensor to provide the traveled distance of a MAV (Micro Aerial Vehicle) [Honegger et al., 2013]. Mobile robot endowed with mouse sensor can also be able to perform odometry at very low cost [Ross et al., 2012].

1.3.3 Target positioning sensors

The low spatial resolution of the compound eyes should be not well suited for target localization. Therefore, fewer examples of sensors were designed for this task. Two major types of bio-inspired sensors were developed, both taking advantage of the ASF of the fly eyes.

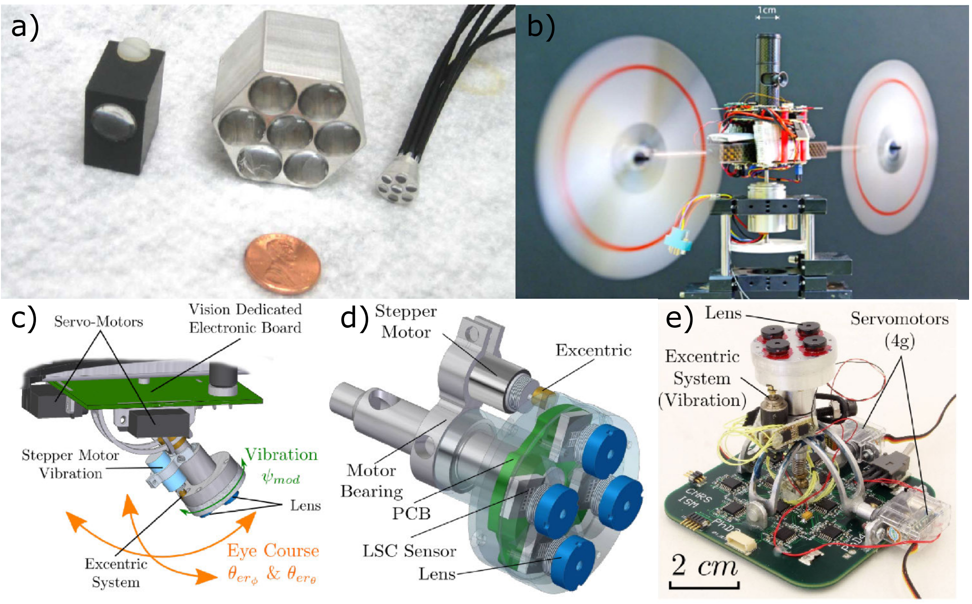


Figure 1.12: a) Different sensors based on the compound eyes. One with only one artificial ommatidia whereas the 2 others are an hexagonal array of 7 artificial ommatidia (reprinted from [Luke et al., 2012]. b) OSCAR robot with its scanning eye decoupled from the body with one rotational degree of freedom. The scanning photoreceptors are actuated thanks to a piezo actuators with an amplitude of 0.1° (Reprinted from [Kerhuel et al., 2010]) c) CAD view of the complete oculomotor system oriented toward the ground. The orientation of the 2-axis gimbal eye is controlled through two fast tiny servomotors. d) CAD view of the eye: the stepper motor combined with the eccentric mechanism yield a periodic ($55Hz$) rotation of small amplitude (around 3.5°) making the lens of each eye translate periodically above the four linear arrays composed of 6 pixels (LSC sensor). e) Picture of the complete oculomotor system. (c-e) are reprinted from [Manecy et al., 2016])

A solution is to only recontruct the position of the feature placed in between by measuring the signal amplitude at the output of the artificial ommatidia. This solution has to be endowed with an external photodiode to cancel the ambient lighting information [Benson et al., 2009,Luke et al., 2012,Frost et al., 2016] (see figure 1.12a).

Another solution, developed in the lab, is based on the retinal movements observed in the fly compound eyes. The first sensor to displays such property was named OSCAR (Optical Scanning sensor for the Control of Aerial Robots) [Viollet and Franceschini, 2010]. It consists of 2 photodiodes placed behind a lens and submitted to a known scanning pattern thanks to a piezo actuator. It is able to locate an edge accurately by computing the time delay between the two photodiodes signals, properly filtered. Onboard a tethered robot, it demonstrates its performance in target tracking by localizing it with a resolution 70-fold better than its static resolution. Then, the VODKA algorithm, reproduced the performance with a similar device. However, the scanning law was sinusoidal and no knowledge of the scanning phase is needed [Kerhuel et al., 2012]. Then, OSCAR II (see figure 1.12b), with this new algorithm and a decoupled-eye, showed the ability to track an edge and a bar, even with strong perturbations, with a very small FOV ($\approx 4°$) [Kerhuel et al., 2010]. HyperQuad was a quadrotor with a decoupled eye based on similar visual principles. It was able to locate a cross and to stabilize its flight without accelerometer measurements thanks to its eye fixation reflex. The performance was also achieved with a moving cross. The HyperQuad's eye was composed of 4 arrays of 6 pixels placed orthogonally and it is the lenses which are submitted to a quasi-translational movement [Manecy et al., 2016] (see figure (see figure 1.12 for details). For target tracking, a bio-inspired approach is based on Elementary Small Target Motion Detectors (ESTMDs) using spatially filtered camera images [Bagheri et al., 2015] mimicking the dragonfly [Wiederman and O'Carroll, 2011].

In the robotic community, the standard cameras are mostly preferred. Some alternatives with Infrared Led beacons appear with a reduced cost and limited payload for advantages [Wenzel et al., 2010,Raharijaona et al., 2015].

1.3.4 CurvACE: A miniature Curved Artificial Compound Eye

A - Overview

The CurvACE project was an European project that had the purpose of developing an artificial compound eye with microlens arrays and adaptive photoreceptors made of VLSI [Pericet-Camara et al., 2011]. The resulting sensor is shown in figure 1.13 and is more similar to the trilobite eye with its cylindrical shape than the sphere of the fly eye. It is composed of 42 columns of 15 photosensors, enabling, after bending on a cylindrical scaffold, a field of view of 180 × 60°.

It reproduces some characteristics of one drosophila compound eye (see the table 1.1 from [Floreano et al., 2013]). Indeed the interommatidial angle $\Delta\varphi$ and the acceptance angle $\Delta\rho$ are similar to the ones of the drosophila eye with 4.2° on average in both. The CurvACE sensor also includes an adaptation to illuminance.

B - Sensor Architecture

Optical characteristics To provide a wide field of view in both directions, two different methods were used. The bending of a flexible PCB provides the horizontal distribution of the columns, whereas a clever shift of the lens relative to the photodiode enables the vertical distribution of the direction of vision (see figures 1.13 and 1.14).

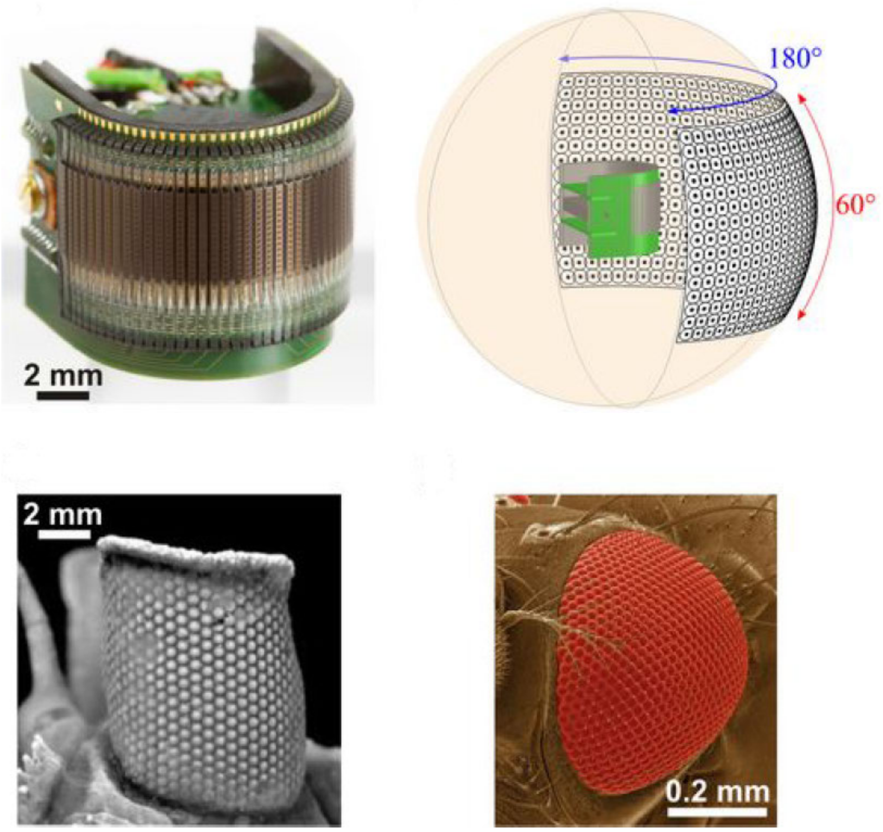


Figure 1.13: Artificial and natural curved compound eyes. (A) Image of the CurvACE prototype. The entire device occupies a volume of $2.2cm^3$, weighs $1.75g$ and consumes $0.9W$ at maximum power. (B) Illustration of the panoramic FOV of the fabricated prototype. The dots and circles represent the angular orientation and acceptance angle $\Delta\rho$ of every ommatidium, respectively. Compound eye of the extinct trilobite *Erbenochile erbeni* [Fortey and Chatterton, 2003] (C) and of the fruit fly Drosophila melanogaster (D). (from [Floreano et al., 2013])

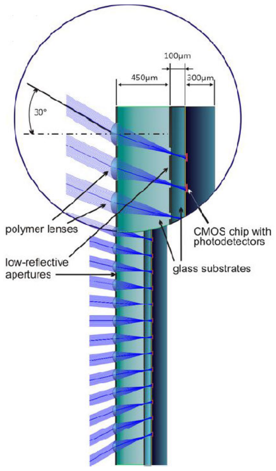


Figure 1.14: Cross section through an ommatidial column and ray tracing (Zemax; Radient Zemax, LLC) in the optics layer. The blue rays show the path of the light focused by each microlens on its photoreceptor (red) via two apertures (Reprinted from [Floreano et al., 2013])

Table 1.1: Specifications of CurvACE prototype compared with the charactersitics of the Drosophila Melanogaster compound eye (from [Floreano et al., 2013])

Characteristics	CurvACE	Drosophila eye
Number of ommatidia	630	600-700
Facet diameter [μm]	172	16 [Franceschini and Kirschfeld, 1971]
Eye diameter [mm]	12.8	0.36 [Franceschini and Kirschfeld, 1971]
Facet diameter/Eye diameter [%]	1.3	4.4
Interommatidial angle $\Delta\varphi$ [°]	~ 4.2	$\sim 4.7 - 5.5$ [Franceschini and Kirschfeld, 1971]
Acceptance angle $\Delta\rho$ [°]	4.2	~ 4.5 [Götz, 1965]
FOV [°]	180×60	160×180 [Heisenberg and Wolf, 1984]
Signal acquisition bandwidth [Hz]	300	< 100 [Laughlin and Weckström, 1993]
Adaptability to illuminance	Yes	Yes [Gu et al., 2005]
Crosstalk prevention	Yes	Yes [Götz, 1965]

An aperture pattern is realized between the polymer lenses and the CMOS chip to prevent cross talks between the photonic flows coming from different lenses. This layout is to compared with the apposition compound eyes previously presented (in section1.2.2). Moreover, this aperture pattern is used to create an Angular Sensitivity Function (ASF) close to a Gaussian as it can be found in the fly compound eyes. It uses the diffraction property of light that goes through a hole.

Description of the VLSI circuit The CurvACE sensor is composed of 630 cells, each one developed around an octogonal photodiode. One cell is composed of a Delbrück cell responsible for the light adaptation; a low-pass filter, to ensure the same cutting frequency for the Analog-to-Digital Converter (ADC); and a follower circuit.

The Delbrück adaptive photodetector consists of a logarithmic circuit associated with a high gain negative feedback loop, as shown in Figure 1.15. It is based on a MOSFET feedback (MFB) transistor operating in the sub-threshold region where the current-to-voltage characteristic shows logarithmic variations in a large dynamic range of up to several decades. The adaptive element responsible for the DC output levels acts like a very high resistance and makes the output signal V_{out} follow the gate voltage of the MFB transistor. The non-linear resistance of the adaptive element decreases in the case of fast transient signals. The adaptation to variations in the ambient light levels is therefore relatively slow, whereas the compensations for changes in contrast are much faster.

The low-pass filter next in line limits the cut-off frequency of the photodetector to a value ($300Hz$) that is compatible with the sampling rate of the 10-bit ADC used to digitize each photodetector's output signal.

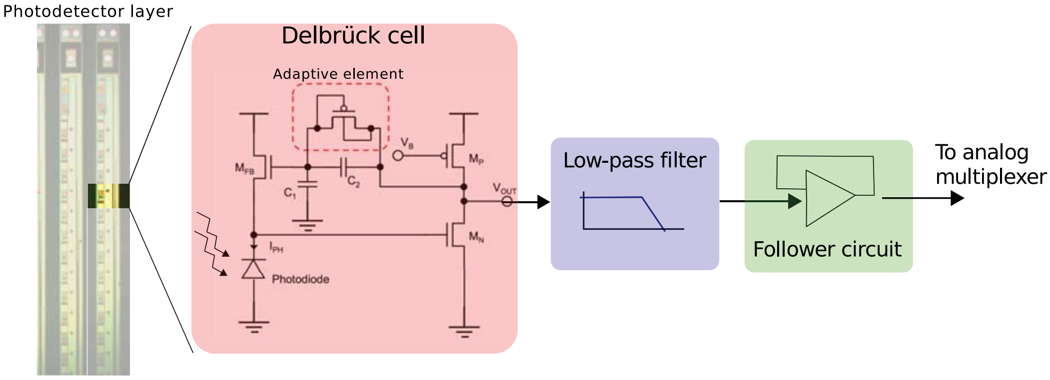


Figure 1.15: One of the 15 photodetector cells of one CurvACE column. The original circuit developed by Delbrück and Mead [Delbruck and Mead, 1994] was enhanced here by cascading a first-order low-pass filter to prevent temporal aliasing. (Adapted from [Floreano et al., 2013] and [Viollet et al., 2014])

At last, as shown in Figure 1.15, a follower stage based on an operational amplifier was introduced between the anti-aliasing filter and the ADC to reduce the input time constant and, thus, to improve the accuracy required by the sampling rate of $2ksamples/s$.

There is one ADC per column and one readout circuit to enable a custom digital communication to a microcontroller (μC). In this architecture, there are 2 μC responsible of the readout of columns 1 to 22 and 21 to 42 respectively. This means that column 21 and 22 can be read by both.

C - Photodetector response

Characterization of the Auto-Adaptive Photodetectors The auto-adaptive photodetectors were designed in order to reduce the sensitivity to ambient lighting variations and increase the sensitivity to the contrasts change. Figure 1.16 shows the dynamic gain adjustment of one CurvACE element (photodetector equipped with optics) facing a set of still or moving black and white stripes placed $195mm$ away. It is obvious that the Delbrück pixel compensates the change in the illuminance induced by opening a sun-blind and modulates its response according to the contrasts of the bars in its FOV.

Characterization of the sensor response relative to lighting variation To add more information about the light adaption, a characterization of the photodetector's response to a variation of the ambient light intensity was made. A light source is pointing to the pixel as depicted in figure 1.17a and its intensity is modulated. The Steady state point is noted in red and the peak transient response is marked in green on an axis showing the Voltage photodetector output versus the irradiance in $W.cm^{-2}$. Figure 1.17b shows the resulting Sshapes. It highlights that the pixel response has the same slope whatever the steady state point at the beginning. Moreover, the Steady state response varies slightly over 4 decades.

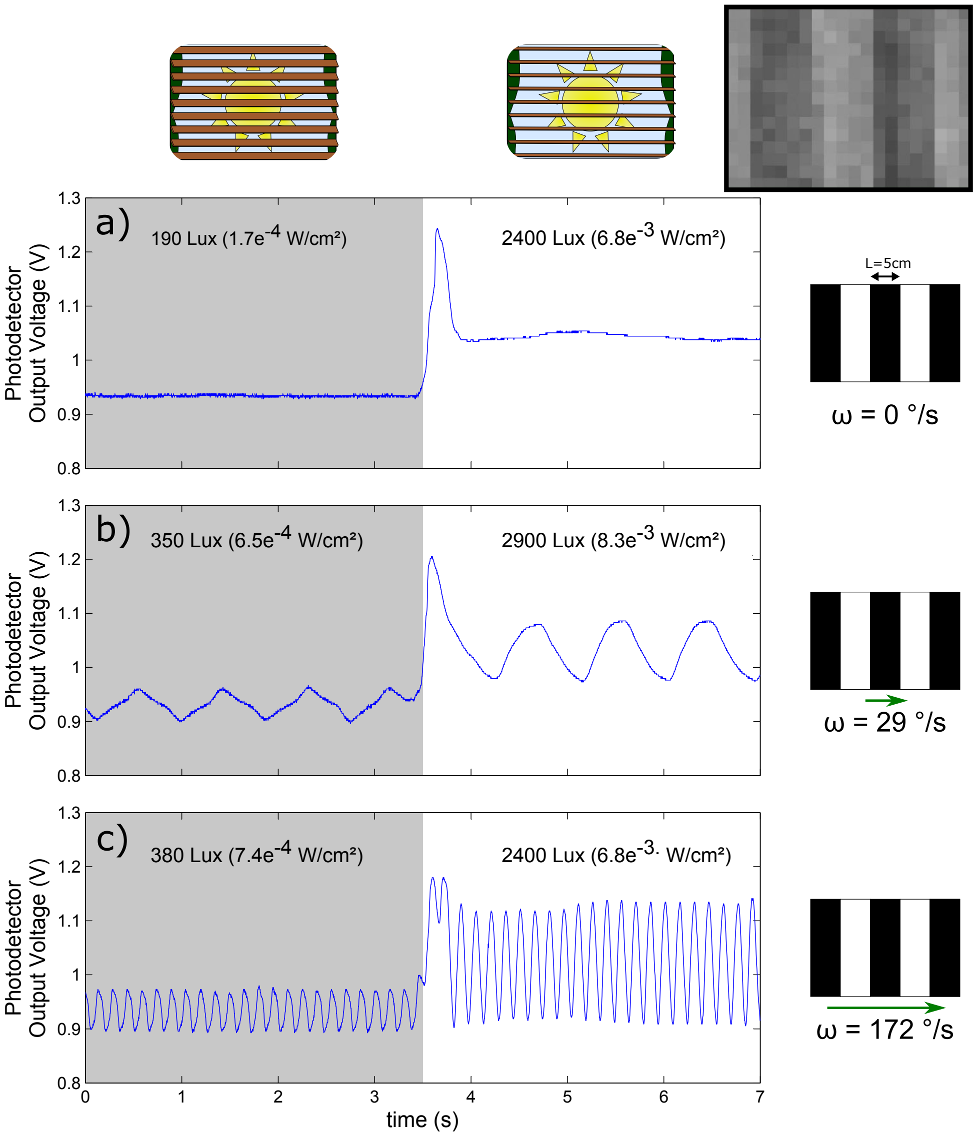


Figure 1.16: Response of one CurvACE element (a photodetector with optics) to sharp changes in the illuminance (obtained by opening a sun-blind) after digitizing and sampling the data at a frequency of $500Hz$. The photodetector output signal was recorded while facing a periodic pattern (a set of stripes with a width of $50mm$ placed $195mm$ from the sensor) exposed to natural lighting conditions) (a) stationary (static state) and translating at a speed of (b) $29°/s$ and (c) $172°/s$. The photodetector compensated quickly (about $0.5s$) for the increase in the illuminance and adapted its gain, as well as amplified the transient signals generated by the moving pattern. The inset (upper right corner) shows the periodic pattern acquired by the CurvACE sensor at a distance of $15cm$ with a region of interest composed of 20 by 15 ommatidia under a lighting of $1500Lux$. (Reprinted from [Viollet et al., 2014])

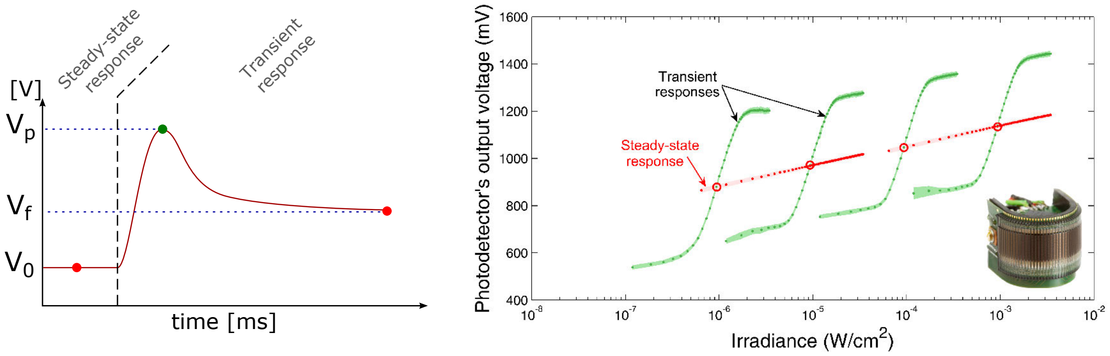


Figure 1.17: a) CurvACE photodetector response to a step variation of illuminance. V_0 and V_f are the steady-state response at the starting illuminance and the end illuminance respectively and are marked with red dots. V_p is the peak response after the step perturbation mark with a green dot. b)CurvACE autoadaptation to ambient light at the single ommatidium level. Steady-state (red dots) and transient (green dots) responses of the adaptive analog VLSI photodetectors [design based on a circuit proposed by Delbrück and Mead [Delbruck and Mead, 1994]]. Each of the four dynamic operating curves (in green) shows the V(log I) response, averaged V_p value over 11 ommatidia (photodetectors with optics) of one column, to step increments and decrements of irradiance about four steady levels V_0 (red circles). (Reprinted from [Floreano et al., 2013])

1.3.5 Neuromorphic vision

The neuromorphic cameras are event-based visual sensors that mimic the neural network behavior of the brain to process the information. It means that there is no image anymore, each pixel handles its own information individually. The concept is to mimic the behavior of the human eye retina which has multiple layers, responsible for capturing light, realizing a first a spatial and temporal filtering before transmitting the information to the brain [Gollisch, 2009].

The first sensor based on this technology to be sold was the "Dynamic Vision Sensor" (DVS) [Lichtsteiner et al., 2006, Lichtsteiner et al., 2008]. It responds to the variation of light intensity of each pixel individually and with a temporal resolution close to the microsecond. Indeed, it is not frame based, meaning that it transmits information completely asynchronously when a change occurs. The main change compared to cameras is that the transmission is "event-based" as it is not a clock that controlled the acquisition. This sensor is well-suited for high speed application, but no real image can be provided by this visual sensor (see figure 1.18 for details).

Another sensor named ATIS, which stands for "Asynchronous, Time-based Image Sensor", was designed to overcome this limit [Posch et al., 2008, Posch et al., 2011]. A change detector triggers a new exposure measurement after a brightness change. The exposure measurement is the duration of the photocurrent integration until it reaches a certain value. The output can therefore provide grayscale imaging as well as change event ON and OFF.

This kind of sensor opens new approaches in computer vision with completely asynchronous systems. Some works has already been done to calculate optic flow with spiking neural networks to have a complete asynchronous computation process [Orchard et al., 2013]. Others have also developed pattern recognition in order to perform feature tracking [Lagorce et al., 2015]. An

Event-based camera have also been used on a mobile robot to track a moving target [Liu et al., 2016].

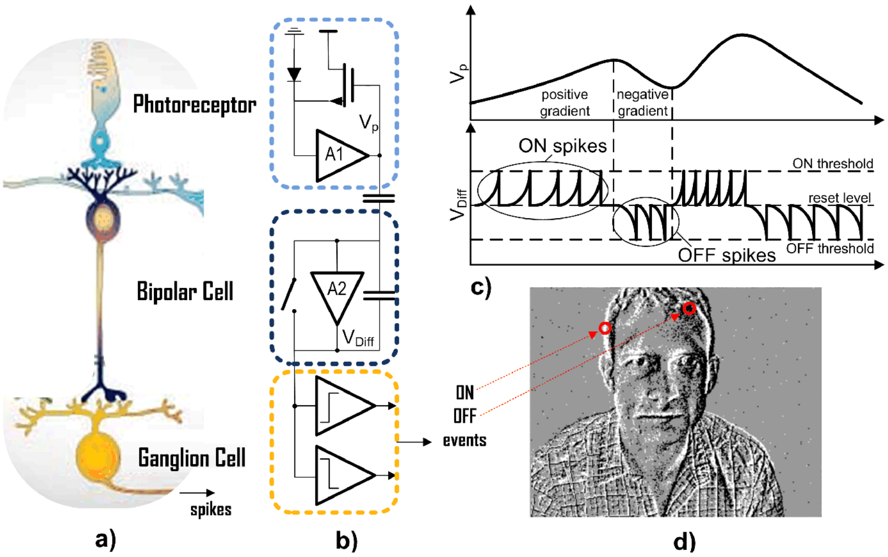


Figure 1.18: a) Simplified three-layer retina model and b) corresponding DVS pixel circuitry; in c) typical signal waveforms of the pixel circuit are shown. The upper trace represents an arbitrary voltage waveform at the node Vp tracking the photocurrent through the photoreceptor. The bipolar cell circuit responds with spike events of different polarity to positive and negative gradients of the photocurrent, while being monitored by the ganglion cell circuit that also transports the spikes to the next processing stage; the rate of change is encoded in inter-event intervals; d) shows the response of a QVGA array of DVS pixels to a natural scene (person moving in the field-of-view of the sensor). Events have been collected for some tens of milliseconds and are displayed as an image with ON (going brighter) and OFF (going darker) events drawn as white and black dots. (from [Posch, 2012])

Chapter 2

Sensing its position according to the environment

Contents

2.1 Summary

[Kerhuel et al., 2012] established that it is possible to localize an edge and a bar with a pair of scanning photosensors. [Juston et al., 2014] designed an edge/bar detector in order to have a similar slope for both patterns. Based on these previous works, a fusion algorithm using multiple pairs was developed. This chapter presents it and an application to visual stabilization of a tethered robot over a textured pattern is made. It endowed a robot with short range odometry and enabled it to return to its hovering position after perturbations. Some limits are observed, mainly because of the non-linearity of the local measurement according to the visual scene. Although this limits is overcome by selecting the best contrasts seen over the whole field of view. It highlights the fact a high number of pairs is required at the input of the fusion algorithm to provide accurate output.

2.2 Article 1: A small-scale hyperacute compound eye featuring active eye tremor: application to visual stabilization, target tracking and short range odometry

Description and Contribution

The following paper was accepted the 9th of January 2015 and published the 25th of February 2015 in the journal *Bioinspiration and Biomimetics* [Colonnier et al., 2015a]. R. Juston realized the demodulation and the fusion algorithm programming. My contribution to this paper is the modification of the FOV of the sensor to fit to endow the robot with a visual odometry ability. I calibrated the sensor to convert the angular measurement described in [Colonnier et al., 2015b] into speed and position measurements. A. Manecy did the rotor and attitude control, as well as the attitude estimation. A. Manecy and I designed the position controller and realized most of the experiments together. S. Viollet worked on the design of the robot and on the gaze stabilization reflex. I wrote the paper with the help of S. Viollet and A. Manecy for the section 2.2.5.

2.2.1 abstract

In this study, a miniature artificial compound eye (15mm in diameter) named CurvACE (Curved Artificial Compound Eye) was endowed for the first time with hyperacuity, using similar micro-movements to those occurring in the fly's compound eye. A periodic micro-scanning movement of only a few degrees enables the vibrating compound eye to locate contrasting objects with a 40-fold greater resolution than that imposed by the interommatidial angle. In this study, we developed a new algorithm merging the output of 35 local processing units consisting of adjacent pairs of artificial ommatidia. The local measurements performed by each pair are processed in parallel with very few computational resources, which makes it possible to reach a high refresh rate of 500Hz. An aerial robotic platform with two degrees of freedom equipped with the active CurvACE placed over naturally textured panels was able to assess its linear position accurately with respect to the environment thanks to its efficient gaze stabilization system. The algorithm was found to perform robustly at different light conditions as well as distance variations relative to the ground and featured small closed-loop positioning errors of the robot in the range of 45mm. In addition, three «tasks of interest»were performed without having to change the algorithm: short range odometry, visual stabilization and tracking contrasting objects (hands) moving over a textured background.

2.2.2 Introduction

According to the definition originally proposed by G. Westheimer in 1975 [Westheimer, 1975] and recently reformulated in 2009 [Westheimer, 2009]: "Hyperacuity refers to sensory capabilities in which the visual sensor transcends the grain imposed by its anatomical structure". In the case of vision, it means that an eye is able to locate visual objects with a greater accuracy than the angular difference between two neighboring photoreceptors $\Delta\varphi$. This study presents the first example of an artificial compound eye which is able to locate the features encountered with a much greater accuracy than that imposed by its optics (i.e. $\Delta\varphi$). Based on findings originally observed in fly vision, we designed and constructed an active version of the previously described artificial compound eye CurvACE [Floreano et al., 2013, Viollet, 2014]. Active CurvACE features two properties that are usually banned by optic sensor designers because they impair the sharpness of the resulting images: optical blurring and vibration. The active visual principle applied here is based on a graded periodic back-and-forth eye rotation of a few degrees scanning the visual environment. Scanning micro-movements of this kind have been observed in human [Rolfs, 2009] and several invertebrates such as crabs [Sandeman, 1978], molluscs [Land, 1969] and arachnids [Land, 1982, Kaps and Schmid, 1996].

The first micro-scanning sensor based on the periodic retinal micro-movements observed in the fly (for a review, see [Viollet et al., 2014]) was presented in [Mura and Franceschini, 1996], whereas recent developments [Viollet and Franceschini, 2010, Kerhuel et al., 2012, Juston et al., 2014] have led to the implementation of bio-inspired vibrating sensors endowed with hyperacuity. However, hyperacuity has also been obtained in artificial retinas without using any retinal micro-scanning processes, based on the overlapping Gaussian fields of view of neighboring photosensors (for a review, see [Wright and Barrett, 2013]). The authors of several studies have assessed the hyperacuity of an artificial compound eye in terms of its ability to locate a point source [Benson et al., 2009], a bar [Luke et al., 2012], a single line [Riley et al., 2008] (the bar and the line both take the form of a stripe in the field of view), an edge [Brückner et al., 2006], and to sense the edge's orientation [Wilcox and Thelen, 1999]. The robustness of these visual sensors' performances with respect to the contrast, lighting conditions and distance from the object targeted (contrasting edges or bars) has never been assessed prior to the present study involving the use of a retinal micro-scanning approach. However, assuming that a priori knowledge is available about the target's and obstacles' contrast, Davis *et al.* [Davis et al., 2009] implemented efficient target tracking and obstacle avoidance behaviour onboard a ground-based vehicle equipped with a bulky apposition eye consisting of an array of 7 ommatidia.

As Floreano *et al.* [Floreano et al., 2013] have shown, an artificial curved compound eye can provide useful optic flow measurements. In addition, we established here that an artificial compound eye performing active periodic micro-scanning movements combined with appropriate visual processing algorithms can also be endowed with *angular position* sensing capabilities. With this visual sensing methods, an aerial robot equipped with active CurvACE was able to perform short range visual odometry and track a target moving over a textured ground. The bioinspired approach used here to obtain hovering behaviour differs completely from those used in studies involving the use of computer vision or optic flow.

In this context, it is worth quoting, for example, two recent studies using binocular vision [Shen et al., 2013] and monocular vision [Forster et al., 2014] to perform hovering without any drift and visual odometry, respectively. However, the latency (about $18ms$) of the embedded visual processing algorithms used by the latter authors still limits the reactivity of the supporting aerial robotic platform. Other strategies combined visual cues with barometric data to obtain a visual odometer [Kendoul et al., 2009] or with an ultrasonic range finder to implement a hovering autopilot [Engel et al., 2012]. In the case of hovering behaviour, many studies have been based on the assumption that the robots in question have previous knowledge of particular features present in the environment. Mkrtchyan *et al.* [Mkrtchyan et al., 2009] enabled a robot to hover using only visual cues by fixating 3 black rectangles, but its altitude was controlled by an operator. Likewise, landing procedures have been devised, which enabled robots equipped with IMUs and visual sensors to detect specific geometrical patterns on the ground [Zhang et al., 2009, Yang et al., 2013]. Bosnak *et al.* [Bosnak et al., 2012] implemented an automatic method of hovering stabilization in a quadrotor equipped with a camera looking downward at a specific pattern. Gomez-Balderas *et al.* [Gomez-Balderas et al., 2014] stabilized a quadrotor by means of an IMU and two cameras, one looking downward and the other one looking forward. In the latter study, the optic flow was computed on the basis of the images perceived when looking downward and the robot's position was determined by using a known rectangular figure placed on a wall, which was detected by the forward-facing camera.

Optic Flow (OF) has also been used along with IMU measurements to perform particular flight maneuvers such as hovering [Lim et al., 2012] and landing [Herissé et al., 2012, Ruffier and Franceschini, 2014]. The robot developed by Carrillo *et al.* [Carrillo et al., 2012] rejected perturbations by integrating the OF with information about the height obtained via an ultrasonic range finder. In their experiments, the goal was to follow a path defined by a contrasting line.

Honegger *et al.* [Honegger et al., 2013] have also developed an optical flow sensor for stabilizing a robotic platform hovering over a flat terrain, but the performances of this sensor over a rugged terrain or a slope were not documented. On similar lines, Bristeau *et al.* [Bristeau et al., 2011] developed a means of estimating the speed of a quadrotor by combining the speed given by an optic flow algorithm with that provided by an accelerometer.
The visual processing algorithm presented here estimates a displacement by measuring the angular *position* of several contrasting features detected by active CurvACE. In this respect, this method differs completely from those used in previous studies based on the use of the OF, which are similar to *speed* estimation methods.

The active version of the CurvACE sensor and the fly's retinal micro-movements are described in section 2.2.3A, and a model for the vibrating eye, including its special optics, is presented in section 2.2.3B. The visual processing algorithms resulting in hyperacuity is described in section 2.2.4. A complete description of the implementation of this sensor on a tethered robot named HyperRob is given in section 2.2.5, and the robot's capability to assess its own linear position relative to the environment and track a target thanks to the active CurvACE, based on the novel sensory fusion algorithm developed here is established in section 2.2.6 (see also section 2.2.4C).

2.2.3 Description of the visual sensor: active CurvACE

A - Inspiration from the fly's visual micro-scanning movements

In this study, visual hyperacuity results from an active process whereby periodic micro-movements are continuously applied to an artificial compound eye. This approach was inspired by the retinal micro-movements observed in the eye of the blowfly *Calliphora* (see figure 2.1a). Unlike the fly's retinal scanning movements, which result from the translation of the photoreceptors (see figure 2.1b) in the focal plane of each individual facet lens (for a review on the fly's retinal micro-movements see [Viollet et al., 2014]), the eye tremor applied here to the active CurvACE by means of a micro-stepper motor (figure 2.1c) results from a periodic rotation of the whole artificial compound eye. The section 2.2.3C shows in details that both scanning processes lead to a rotation of the visual axis.

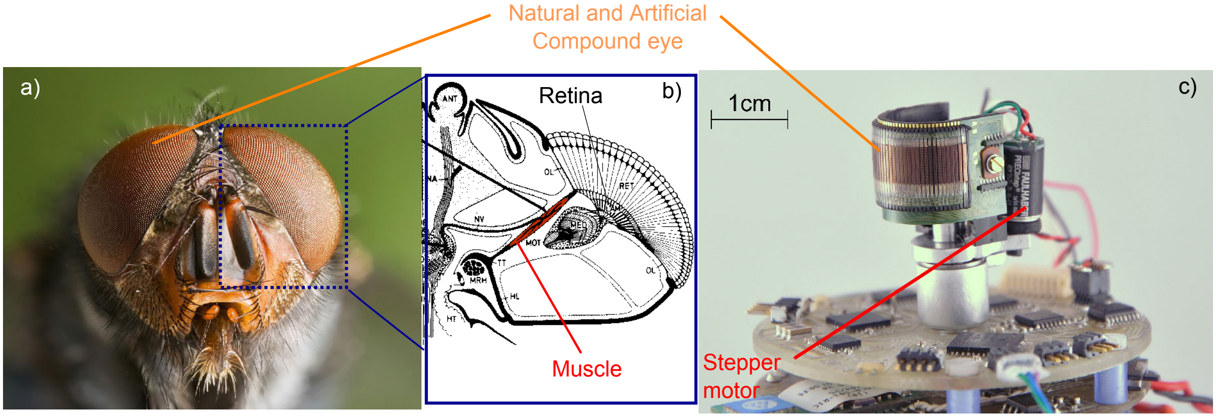


Figure 2.1: (**a**) Head of a *Calliphora vomitaria* (Picture : J.J. Harrison, Wikimedia commons). (**b**) Top view of a fly's head showing the orbito-tentorialis muscle (MOT in red) attached to the back of the head (the fixed part : TT) and the base of the retina (the moving part : RET). The muscle MOT is one of the two muscles responsible for the periodic retinal translation (for a review on the fly's retinal micromovements see [Viollet et al., 2014]). Modified from [Hengstenberg, 1972]. (**c**) The active CurvACE with its vibrating mechanism based on the use of a small stepper motor.

Here we describe in detail the active version of the CurvACE sensor and establish that this artificial compound eye is endowed with hyperacuity, thanks to the active periodic micro-scanning movements applied to the whole eye.

B - Modelling the optics

As described in [Floreano et al., 2013], the CurvACE photosensors array has similar characteristics to those of the fruitfly's eye in terms of the number of ommatidia (630), light adaptation, the interommatidial angle ($\Delta\varphi = 4.2°$ on average) and a similar, Gaussian-shaped Angular Sensitivity Function (ASF). This specific ASF removes "insignificant" contrasts at high spatial frequencies.

To replicate the characteristics of its natural counterpart (see figure 2.2a), CurvACE was designed with a specific optical layer based on the assembly consisting of a chirped microlens array (lenslet diameter: $172\mu m$) and two chirped aperture arrays. In the case of active CurvACE, the optical characteristics remain constant during the scanning process.

Each ASF of active CurvACE (figures 2.2b and 2.2c) can be characterized by the acceptance angle $\Delta\rho$, which is defined as the angular width at half of the maximum ASF. The ASF along 1D $s(\psi)$ of one CurvACE ommatidium can therefore be written as follows:

$$s(\psi) = \frac{2\sqrt{2\ln(2)}}{\pi\Delta\rho} e^{-4\ln(2)\frac{\psi^2}{\Delta\rho^2}} \tag{2.1}$$

where ψ is the angle between the pixel's optical axis and the angular position of a point light source.

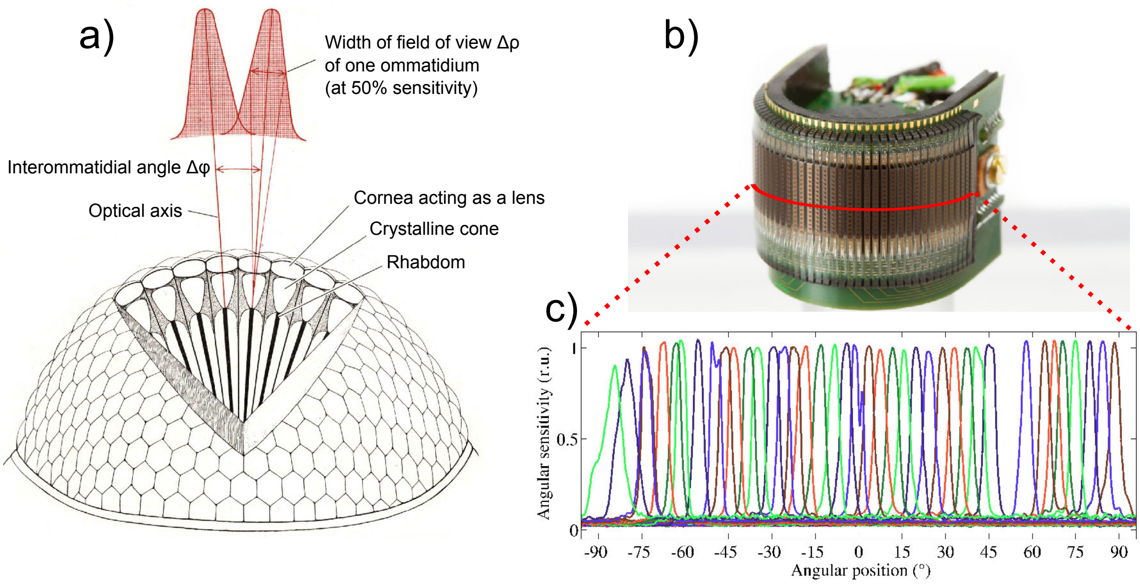


Figure 2.2: (**a**) Schematic view of a compound eye, showing the two main optical parameters of interest: the interommatidial angle $\Delta\varphi$, defined as the angle between optical axes of two adjacent ommatidia, and the acceptance angle $\Delta\rho$, defined as the angle at half width of the Gaussian shaped ASF. This particular shape of ASF results from the combination of the Airy diffraction pattern and the geometrical angular width of the rhabdom at the nodal point of the lens [Land and Nilsson, 2012]. The diameter of the facet lenses in the male blowfly *Calliphora* ranges from $20\mu m$ to $40\mu m$, whereas that of the peripheral rhabdomeres is $1.5 - 2\mu m$ (see [Stavenga, 2003] for review). Adapted with permission from [Horridge, 1977]. (**b**) CurvACE sensor and (**c**) the horizontal angular sensitivity functions (ASFs) measured for each artificial ommatidium along the equatorial row (red line) (see [Floreano et al., 2013] for further details). The mean value of the interommatidial angle $\Delta\varphi$ obtained in the middle row (red line) is $4.2° \pm 0.8°$ (SD) and that of the acceptance angle is $4.2° \pm 0.8°$ (SD).

C - Mathematical description of signals generated by vibrating ommatidia

Figures 2.3a and 2.3b compare the rotation of the visual axes resulting from the translation of the retina behind a fixed lens (e.g., in the case of the fly's compound eye) with the rotation of the visual axes resulting from the rotation of the whole eye (e.g., like the mechanism underlying the micro-saccades in the human's camerular eye [Carpenter, 1988]), respectively. In the active CurvACE, we adopted the second strategy by subjecting the whole eye to an active micro-scanning movement that makes the eye rotate back and forth.

As shown in figure 2.3, the retinal micro scanning movements are performed by a miniature eccentric mechanism based on a small stepper motor (1.7 grams in weight, 6mm in diameter) connected to an off-centered shaft [Juston and Viollet, 2012]. This vibrating mechanism makes it possible to control the scanning frequency by just setting a suitable motor speed.

A general expression for the pixel's output signal is given by the convolution of the ASF $s(\psi)$ with the light intensity I of the 1-D scene as follows:

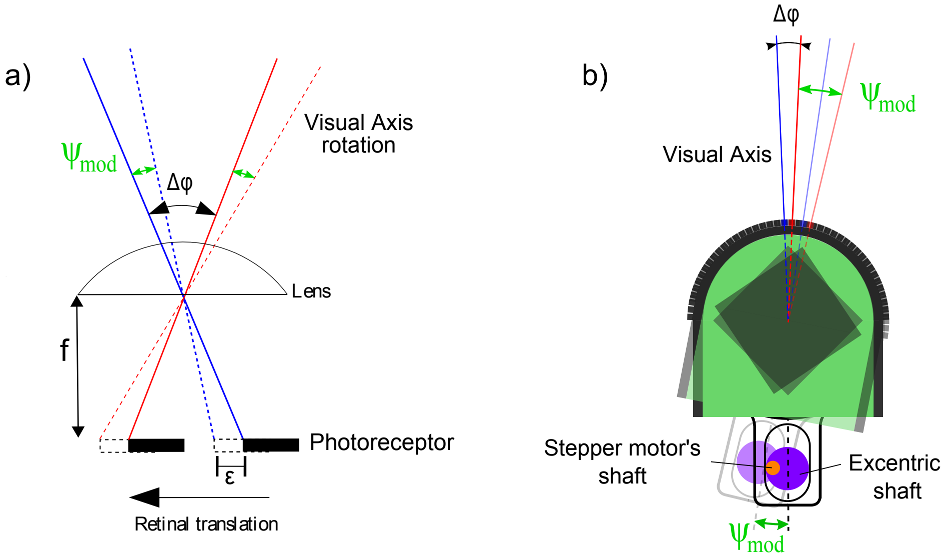


Figure 2.3: Optical axis rotation resulting from (**a**) a micro-displacement ε of the pixels placed behind a fixed lens (e.g., in the case of a compound eye of the fly) or (**b**) a rotation of the whole sensor (e.g., in the case of the active CurvACE sensor). The micro-scanning of active CurvACE is subjected to active periodic rotational movements generated by a miniature eccentric mechanism. The angular vibration ψ_{mod} is generated by a miniature stepper motor represented here by an orange shaft and a purple off-centered shaft, which translates along an elongated hole. The scanning frequency can be easily adjusted by changing the rotational speed of the motor. The scanning amplitude depends on the diameter of the off-centered shaft.

$$Ph(\psi_c) = \int_{-\infty}^{+\infty} s(\psi) \cdot I(\psi - \psi_c)\, d\psi \tag{2.2}$$

where ψ_c is the angular position of a contrasting feature (edge or bar) placed in the sensor's visual field. For example, $I(\psi)$ can be expressed for an edge as follows :

$$I(\psi) = \begin{cases} I_1 & \text{for } \psi < 0 \\ I_2 & \text{for } \psi \geqslant 0 \end{cases} \tag{2.3}$$

and for a bar:

$$I(\psi) = \begin{cases} I_1 & \text{for } \mid \psi \mid < L/2 \\ I_2 & \text{for } \mid \psi \mid \geqslant L/2 \end{cases} \tag{2.4}$$

with L the width of the bar (expressed in rad).

The microscanning movements of the pixels can be modelled in the form of an angular vibration ψ_{mod} of the optical axes added to the static angular position ψ_c of the contrast object (an edge or a bar). The equations for the two pixels (Ph_1 and Ph_2) are therefore:

$$Ph_1(\psi(t)) \quad = Ph\left(\psi_c(t) + \psi_{mod}(t) - \tfrac{\Delta\varphi}{2}\right) \tag{2.5}$$

$$Ph_2(\psi(t)) \quad = Ph\left(\psi_c(t) + \psi_{mod}(t) + \tfrac{\Delta\varphi}{2}\right) \tag{2.6}$$

with ψ_{mod} obeying the following sinusoidal scanning law:

$$\psi_{mod}(t) \quad = A \cdot \sin(2\pi f_{mod} \cdot t) \tag{2.7}$$

With A and f_{mod} describing the amplitude and the frequency of the vibration, respectively. In the case of a whole rotation of the eye, this scanning law is achievable easily by a continuous rotation of the motor with an off-centered shaft. In the case of a translation of the pixels behind a lens, the law should be weighted with the tangent of the ratio between the retinal displacement ε and the focal length f of the lens.
At the end, the photosensor response is a modulated convolution of the light intensity with the gaussian sensitivity function.

2.2.4 Insights into the visual processing algorithms

In this paper, we reuse the Local Processing Unit (denoted LPU) presented in [Juston et al., 2014] and apply the principle to active CurvACE. A LPU is an elementary pair of photosensors endowed with hyperacuity by means of a periodic vibration. The LPU is able to locate very accurately an edge or a bar placed in its small Field Of View (FOV). An artificial compound eye like CurvACE can provide several LPU outputs which can be merged to obtain a bigger FOV and used as a basis for a higher level visual processing algorithm. In the following parts, we describe the different steps of the visual algorithm from the pixel processing to the novel fusion of the LPU's output signals.

A - LPU: from photosensors to an accurate edge and bar location

The Local Processing Unit defined in figure 2.4 is the application of algorithms presented in [Kerhuel et al., 2012] and [Juston et al., 2014]. The first paper ([Kerhuel et al., 2012]) leads to the signal $Output_{Pos}$ resulting from the difference-to-sum ratio of the demodulated pixel output signals described by equation (2.8). The demodulation is realized by means of a peak filter which acts as both a differentiator and a selective filter centered at the scanning frequency ($f_p = f_{mod}$). Then, an absolute value function cascaded with a low-pass filter smooth out the pixel's output signals. The second paper ([Juston et al., 2014]) explains in details the edge/bar detection based on the observed phenomena that the two pixels' output signals are in phase in the presence of an edge and in opposite phase in the presence of a bar. At the output of the LPU, the signal θ_i represents the position of the contrasting feature in the field of view (see equation (2.9)).

$$Output_{Pos} \quad = \frac{Ph_{1_{demod}} - Ph_{2_{demod}}}{Ph_{1_{demod}} + Ph_{2_{demod}}} \tag{2.8}$$

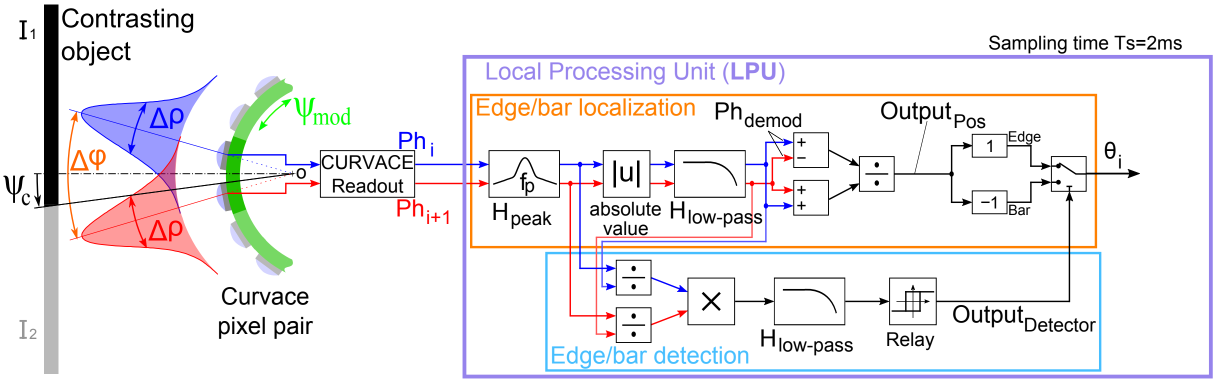


Figure 2.4: Block diagram of the elementary 2-artificial ommatidia local processing unit (LPU) integrated into the active CurvACE for locating edges and bars with great accuracy. The stepper motor (see figures 2.1c and 2.3b) generates a periodic rotation (green double arrows) of the overall visual sensor, resulting in the angular microscanning of their visual axes, in keeping with a sinusoidal law $\psi_{mod}(t)$ (scanning frequency $50Hz$, amplitude about 5° peak to peak with $\Delta\varphi = 4.2°$ and $\Delta\rho = 4.2°$). Two parallel processing pathways (one for edge/bar localization and one for edge/bar detection) were implemented. The edge/bar localization block gives the local angular position θ_i of an edge or bar placed in the visual field of two adjacent photosensors. The edge/bar detection block detects the presence of a bar or edge and triggers the appropriate gain: +1 for edges and -1 for bars. The principle of this detector is described in [Juston et al., 2014]. The central frequency f_p of the peak filter is equal to the scanning frequency ($50Hz$), whereas the cut-off frequency of the second order digital low-pass filter is equal to $10Hz$. Adapted from [Juston et al., 2014].

$$\theta_i(\psi_c) = Output_{Detector}.Output_{Pos} \tag{2.9}$$

With $Output_{Detector}$ equal to (-1) or (1) and θ_i the output signal of an LPU (see figure 2.4).

B - Hyperacute localization of contrasting bars and edges

The characteristic static curves of the active CurvACE obtained with a contrasting edge and a black bar $2.5cm$ in width subtending an angle of 2.86° are presented in figure 2.5.

The curve in figure 2.5a has a tangent hyperbolic profile with respect to the angular position of the edge. It can be clearly seen by comparing the two curves plotted in figure 2.5 that the slopes of the characteristic static curves obtained with a bar and an edge are inverted. A theoretical explanation for the inversion of the slopes is given in [Juston et al., 2014]. This inversion justifies the use of an edge/bar detector in the LPU (see figure 2.4) to compensate for it and still be able to distinguish the direction of the movement.

Moreover, the characteristic curves are independent of the ambient lighting condition. Figure 2.6 shows that the $Output_{Pos}$ signal remains constant even if the ambient light varies over about one decade (from 180 to 1280Lux). A peak is visible at each light change and corresponds to the transient phase during light adaptation of the CurvACE photosensors which lasts only $250ms$. In addition, figure 2.6 shows that active CurvACE is a genuine angular position sensing device

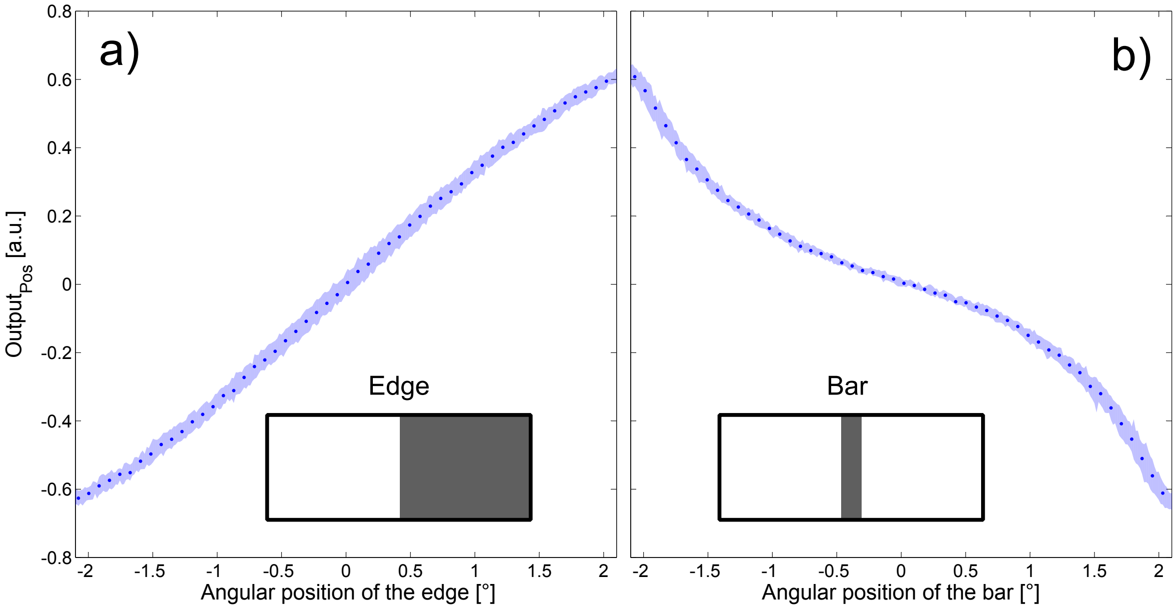


Figure 2.5: Characteristic static curves of the signal $Output_{Pos}$ (see figure 2.4). The $Output_{Pos}$ signal is plotted versus the angular position of a) an edge or b) a bar of $2.5cm$ width placed $50cm$ in front of active CurvACE rotating in 0.016° steps, each lasting $80ms$. The blue points represents the mean response of a LPU, and the blue shaded area represents the standard deviation (STD) of the output. The characteristic static curve obtained with a bar is inverted in comparison with that obtained with an edge. Bars therefore have to be distinguished from edges in order to select the appropriate sign of the $Output_{Pos}$(see figure 2.4).

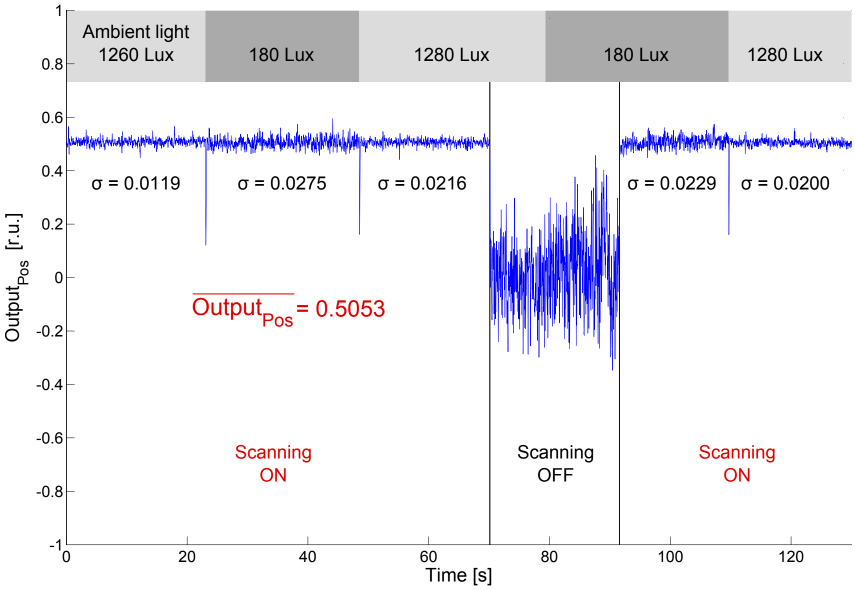


Figure 2.6: The $Output_{Pos}$ signal, corresponding to the angular position of a contrasting edge, plotted versus time. Active CurvACE was subjected to a variation of ambient lighting condition over about one decade and a forced interruption of the visual micro-scanning.

able to provide the angular position of a contrasting object placed in its FOV. Indeed, when the scanning is turned off and on again, the value of the measurement remained the same. This experiment shows that the micro-scanning movement allows us to measure the position of a contrasting object without any drift. A limitation comes here from the contrast, it cannot be theoretically higher than 81.8% for an edge because the auto-adaptative pixel output signal is no longer log linear for changes of illuminance (in W/m^2) greater than one decade (see [Floreano et al., 2013]).

C - Merging the output of local pairs of processing units

To endow a robot with the capability to sense its linear speed and position, a novel sensory fusion algorithm was developed using several LPU in parallel. In this article, a 2D region of interest (ROI) composed of 8 × 5 artificial ommatidia in the central visual field was used in order to expose several pixels to the same kind of movements (figures 2.9a and 2.9c). In other words, the pattern seen by the sensor during a translation of the robot is a succession of edges and bars. The algorithm used here and depicted in figure 2.7 implements the connection between the 8 × 5 photosensors' output signals to an array of 7 × 5 LPUs in order to provide local measurements of edge and bar angular positions. Then a selection is performed by computing the local sum S of two demodulated signals Ph_{demod} obtained from two adjacent photosensors:

$$S_{n,n+1} = Ph_{(n)_{demod}} + Ph_{(n+1)_{demod}} \tag{2.10}$$

Indeed, as a signal $Output_{Pos}$ is pure noise when no feature is in the field of view, an indicator of the presence of a contrast was required. The sum of the demodulated signals was used here to give this feedback, because we observed that the contrast is positively correlated with the sum and the Signal-to-Noise Ratio. Therefore, at each sampling step, each local sum S is thresholded in order to select the best LPU's outputs to use. All the sums above the threshold value are kept and the others are rejected. The threshold is then increased or decreased by a certain amount until 10 local sums have been selected. The threshold therefore evolves dynamically at each sampling time step. Lastly, the index i of each selected sum S gives the index of the pixel pair to process. Thus, the computational burden is dramatically reduced. Moreover, this selection helps to reduce the data processing because only the data provided by the 10 selected LPUs are actually processed by the micro-controller.

In a nutshell, the sensory fusion algorithm presented here selects the 10 highest contrasts available in the FOV. As a result, the active CurvACE is able to assess its relative linear position regarding its initial one and its speed with respect to the visual environment.

It is worth noting that the selection process acts like a strong non-linearity. The output signal θ_{fused} is therefore not directly equal to the sum of all the local angular positions θ_i. The parallel differentiation coupled to a single integrator via a non-linear selecting function merges all the local angular positions θ_i, giving a reliable measurement of the angular orientation of the visual sensor within an infinite range. The active CurvACE can therefore be said to serve as a visual odometer once it has been subjected to a purely translational movement (see section

2.2.6A). Mathematically, the position is given through the 3 equations as follows:

$$\begin{cases} \Delta P_{i_{sel}} & = \theta_{i_{sel}}(t) - \theta_{i_{sel}}(t-1) \\ \theta_{fused}(t) & = \theta_{fused}(t-1) + \frac{1}{10}\sum_{i=1}^{10} \Delta P_{i_{sel}} \\ V_x(t) & = \frac{K}{Ts}(\theta_{fused}(t) - \theta_{fused}(t-1)) \end{cases} \tag{2.11}$$

As shown in figure 2.7, the robot's speed is determined by applying a low-pass filter to the fused output signal S_{fused} (which is the normalized sum of the local displacement error ΔP_{isel}), whereas the robot's position is determined in the same way as θ_{fused}, with the gain K.

To sum up, the algorithm developed here sums the local variation of contrast angular positions in the sensor field of view to be able to give the distance flown with the assumption that the ground height is known.

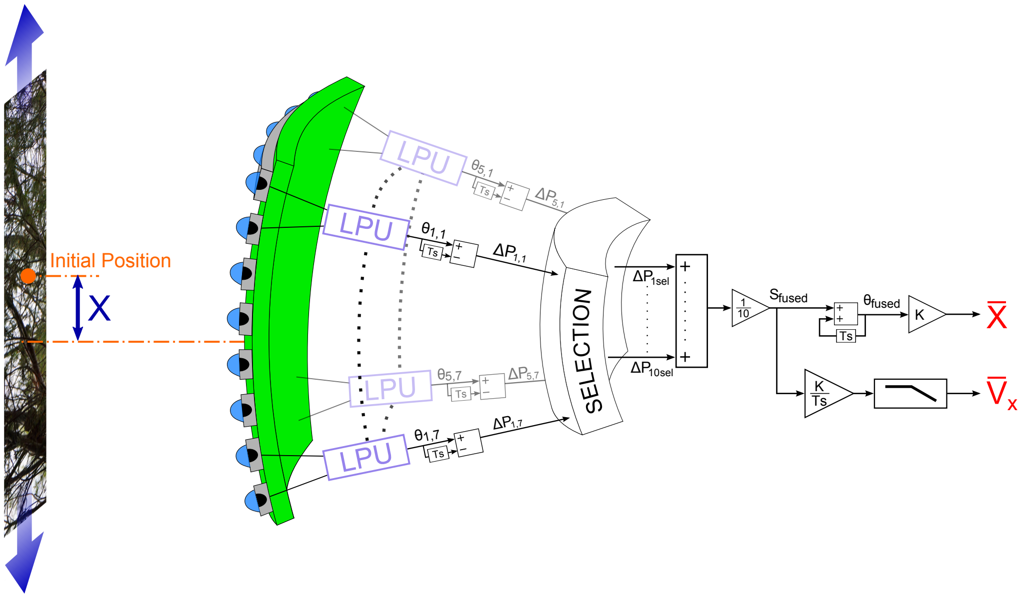


Figure 2.7: Visual processing algorithm
Description of the sensory fusion algorithm to assess the robot's speed $\bar{V_x}$ as well as its position $\bar{X}$ resulting here from a translation of the textured panel with respect to the arbitrary reference position (i.e. the initial position if not resetting during the flight). The 35 (7×5) LPU output signals corresponding to the ROI (8×5 photosensors) of the active CurvACE (see figure 2.9a) were processed by the 35 Local Processing Units (LPUs) presented in figure 2.4. Here, the signal obtained before reaching the discrete integrator, denoted S_{fused}, was used to compute the linear speed. This procedure involved scaling the angular data to millimetric data (gain K) and normalizing the time $\frac{1}{T_s}$, with T_s equal to the sample time of the system. A first order low-pass filter with a cut-off frequency of $1.6Hz$ limited the noise. The robot's position $\bar{X}$ was scaled in millimeters by means of the gain K. The visual processing algorithm presented here provides $\bar{V_x}$ and $\bar{X}$ to the robot's autopilot (see figure 2.13).

2.2.5 HyperRob: an aerial robot equipped with an hyperacute compound eye

The objective of this part of the study was to endow a visually controlled robot, named HyperRob, with the capability to:

- stay at a desired position (reference position) with respect to the visual environment (a textured panel, see figure 2.8)
- return to the reference position even in the presence of perturbation applied to the robot (lateral disturbance) or the textured panel over which the robot is flying.
- track visual target placed between the robot and a textured background environment.

Figure 2.8 summarizes the 4 scenarios used to show the visual stabilization capabilities of HyperRob.

Sections 2.2.3 and 2.2.4 presented the visual sensor and the algorithm we implemented onto HyperRob. It is a twin-rotor robot tethered at the tip of a rotating arm. The robot was free to rotate around its roll axis and could therefore make the arm rotate around its vertical axis (the azimuth). The robot therefore travelled along a circular path with a radius of curvature equal to the length of the arm ($1m$). Figure 2.9 shows the robot equipped with active CurvACE placed on the experimental testbench.

This section introduces HyperRob and we will see in section 2.2.6B that the robot will be able to stay at its initial position thanks to the vibrating active CurvACE estimating its linear speed and position, assuming that its gaze is stabilized.

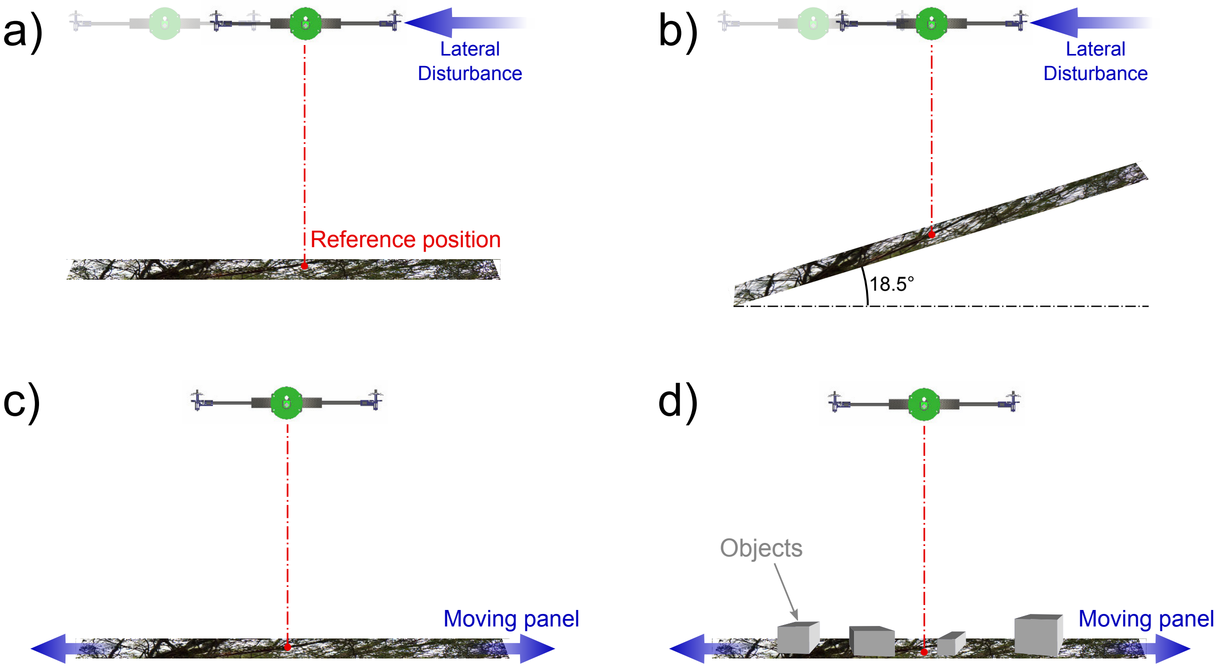


Figure 2.8: The robot HyperRob flies over a textured panel and stays automatically at a programmed reference position despite lateral disturbance (a and b), changes in the ground height (b), movement of the ground (c) or small relief introduced by several objects placed onto the ground (d).

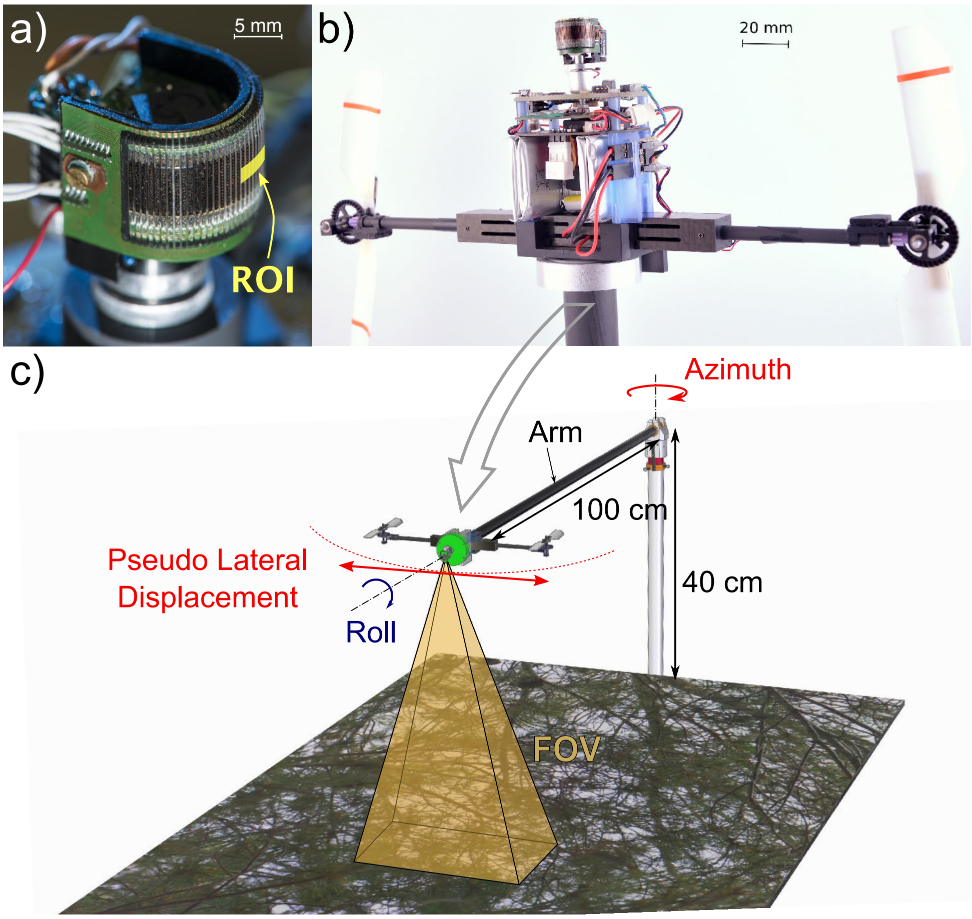


Figure 2.9: Experimental setup of the robot HyperRob
a) Active CurvACE with a region of interest (inset) composed of only 40 artificial ommatidia (8×5), each photosensor is composed of one pixel and one lens. The field of view (FOV) covers about 33.6° by 20.2°. (Picture provided by courtesy of P. Psaïla)
b) The robot HyperRob and its active CurvACE sensor.
c) The complete setup consisted of a twin-propeller robot attached to the tip of a rotating arm. The robot was free to rotate around its roll axis. Arm rotations around the azimuth were perceived by the robot as lateral displacements.

A - Gaze stabilization

In order to determine the robot's speed and position accurately, the gaze direction should be orthogonal to the terrain. But as a simplification, we chose to aligned it with the vertical, assuming that the ground is mostly horizontal. Therefore, the eye has to compensate for the robot's roll angle. To this end, the eye is decoupled from the robot's body by means of a servo motor with a rotational axis aligned with the robot's roll axis. The gaze control system composed of an inertial feedforward control makes the eye looking always in a perpendicular direction to the movement during flight (figure 2.10). The rotational component introduced by the rotating arm supporting the robot can be neglected in this study.

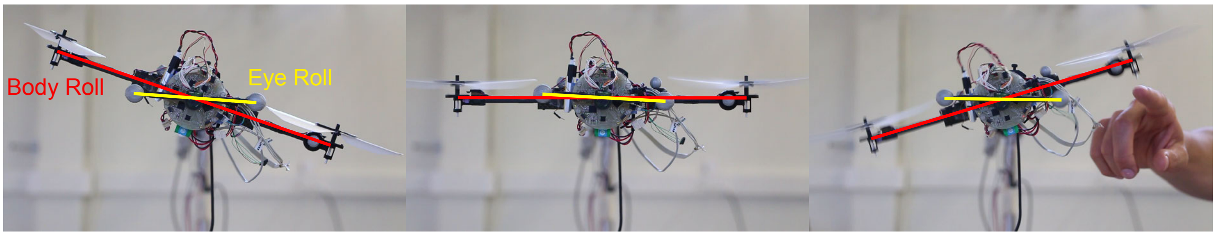


Figure 2.10: Decoupled eye on the roll axis.
Three examples of gaze stabilization. Despite the strong roll disturbances applied by hand to the body, the gaze orientation was kept vertically aligned by the mechanical decoupling provided by a fast servomotor between the robot's body and the visual sensor. It can be clearly seen from the sequence of pictures that the yellow line remained horizontal regardless of the robot's roll angle (red line).

B - Details of the robot HyperRob

HyperRob is an aerial robot with two propellers and a carbon fiber frame, which includes 2 DC motors. Each motor transmits its power to the corresponding propeller (diameter $13cm$) via a 8-cm-long steel shaft rotating on microball bearings in the hollow beam, ending in a crown gear (with a reduction ratio of $1/5$). Two Hall effect sensors were used to measure the rotational speed of each propeller, and hence its thrust, via four magnets glued to each crown gear. Based on the differential thrust setpoints adopted, HyperRob can control its attitude around the roll axis, which is sensed by a 6-axis inertial sensor (here an InvenSense MPU 6000). The robot's position in the azimuthal plane is controlled by adjusting the roll angle. In the robot, where only the roll rotation is free, only 1 axis of the accelerometer and 1 axis of the rate gyro are used. As shown in figure 2.11, active CurvACE is mounted on a fast micro-servomotor (MKS) which makes it possible to control the gaze with great accuracy (0.1°) and fast dynamics (60° within $70ms$, *i.e.* $860°/s$). This configuration enables the visual sensor to be mechanically decoupled from the body (see section 2.2.5A). The robot is fully autonomous in terms of its computational resources and its power supply (both of which are provided onboard). The robot alone weighs about $145g$ and the robot plus the arm weigh about $390g$.

All the computational resources required for the visual processing and the autopilot are implemented on two power lean micro-controllers embedded onboard the robot. The first micro-controller (Microchip dsPIC 33FJ128GP802) deals with the visual processing, whereas the second one (Microchip dsPIC 33FJ128GP804) is responsible for stabilizing the robot. The two micro-controllers have a sampling frequency of $500Hz$. The robot's hardware architecture is presented in detail in figure 2.12.

The micro-controller (μC) Vision communicates with CurvACE via a SPI bus and collects the pixel values of the ROI. The signal S_{fused} is computed and sent to the μC Control via another SPI bus. The latter completes the computation of the position $\bar{X}$ and the speed $\bar{V}_x$. This solution was chosen in order to keep the number of data sent via the SPI bus to a minimum. The μC Control estimates the robot's attitude on the roll axis via a reduced complementary filter (inspired by [Mahony et al., 2008]) and controls the robot's linear position on the basis of the two visual measurements (X and V_x see figure 2.13). The μC Control then sends the propeller speed setpoints to a custom-made driver controlling the rotational speed of each propeller in a

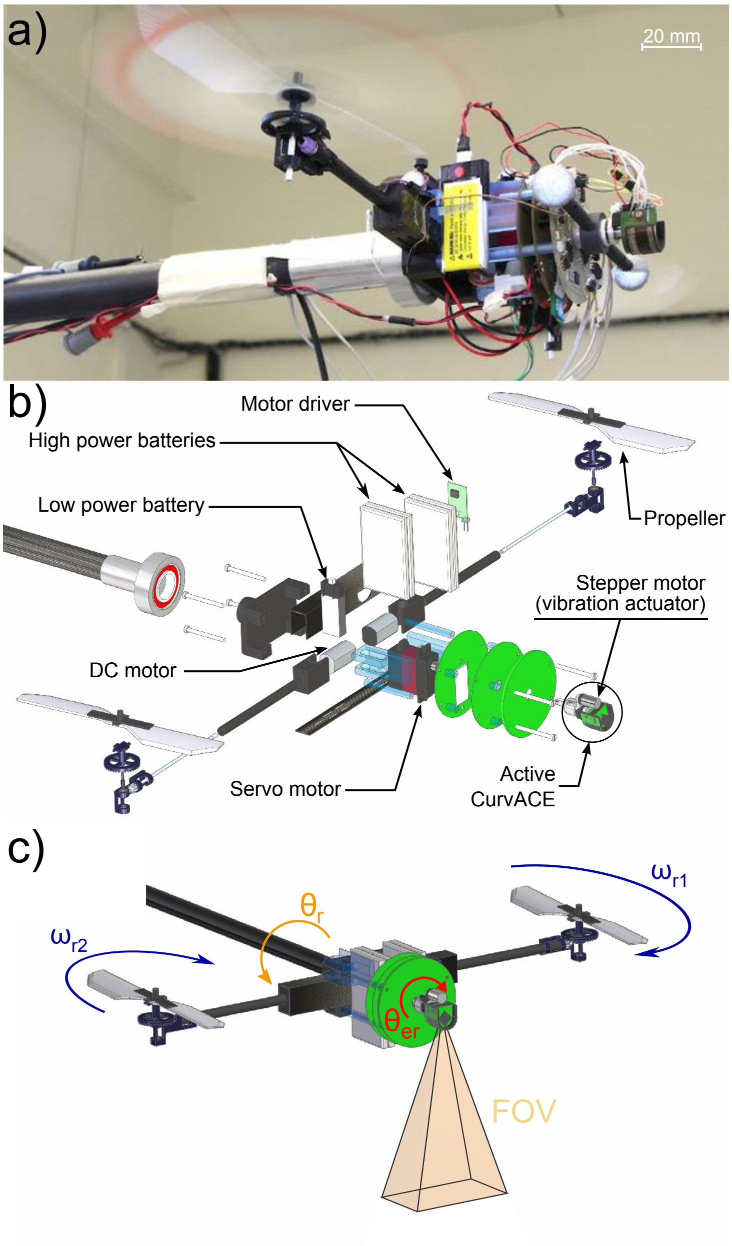


Figure 2.11: Robot description
(a) HyperRob mounted at the tip of the arm, leaving it free to rotate around the roll axis and move along a circular path in the azimuthal plane.
(b) Schematic view of the robot. Four actuators were mounted onboard the robot: two DC motors driving the propeller's rotation are set in the carbon fiber body, one servomotor is used to stabilize the gaze and one stepper motor is used to produce the eye tremor (vibration).
(c) CAD view showing the robot equipped with its visual sensor with a FOV of 33.6° by 20.2°. ω_{r1} and ω_{r2} are the two propellers' rotation speeds, θ_r is the robot's roll angle, and θ_{er} is the eye-in-robot angle.

closed-loop mode.

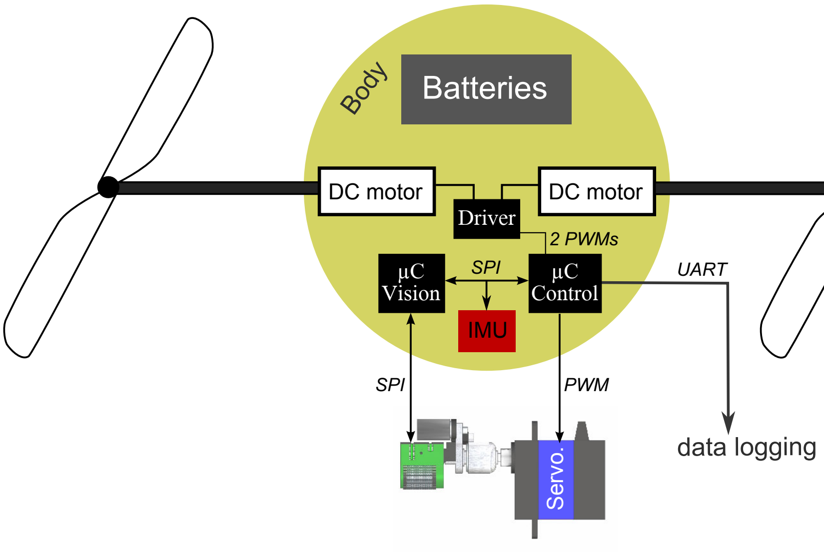


Figure 2.12: Electronic architecture
The autopilot's electronics board is composed of 2 micro-controllers. The first one is involved in the visual processing algorithm (dsPIC33FJ128GP802) and the second one (dsPIC33FJ128GP804) controls the robot's roll and its linear position. The micro-controller denoted μC Vision communicates with CurvACE and receives the digitized pixel output values via a SPI bus. The second micro-controller, denoted μC Control (dsPIC33FJ128GP804), receives the visual sensor's output data from the micro-controller μC Vision via an additional SPI bus. The μC Control then sends the propellers' setpoints to a custom-made driver including a closed-loop control of the rotational speed of each propeller.

C - Control

Assuming the robot to be a rigid body and simplifying the dynamic model for a quad-rotor presented in [Bouabdallah et al., 2004] in the case of a single roll axis, the robot's dynamics can be written as follows:

$$\begin{cases} \dot{X} &= V_x \\ \dot{V}_x &= -\frac{T_{nom}}{m} sin(\theta_r) \\ \dot{\theta}_r &= \Omega_r \\ \dot{\Omega}_r &= \Gamma_\theta = \frac{2l}{I}\delta \end{cases} \tag{2.12}$$

Where X is the robot's lateral position, V_x is its lateral speed, θ_r is the roll angle, Ω_r is the rotational roll speed, l is the robot's half span, I is the moment of inertia, δ is the differential thrust and T_{nom} is the nominal thrust.

$$\begin{cases} 2\delta &= T_1 - T_2 = c_T(\omega_{r1}^2 - \omega_{r2}^2) \\ T_1 &= T_{nom} + \delta^\star \\ T_2 &= T_{nom} - \delta^\star \end{cases} \Leftrightarrow \begin{cases} \omega_{r1} &= \sqrt{\frac{(T_{nom}+\delta^\star)}{C_T}} \\ \omega_{r2} &= \sqrt{\frac{(T_{nom}-\delta^\star)}{C_T}} \end{cases} \tag{2.13}$$

where c_T is the thrust coefficient, and ω_{r1} and ω_{r2} are the right and left propeller speeds, respectively, and $\delta = f_{prop}(s)\delta^\star$, where $f_{prop}(s)$ corresponds to the closed-loop transfer function of the propellers' speed.

As described in figure 2.13, the autopilot controlling both the robot's roll and its position is composed of four nested feedback loops:

- the first feedback loop controls the robot's rotational speed by directly adjusting the differential thrust, and hence the roll torque.
- the second feedback loop yields setpoints on the previous one for tracking the robot's reference roll angle
- the third feedback loop adjusts the robot's linear speed by providing roll angle setpoints
- the fourth feedback loop controls the robot's linear position and yields the reference speed.

In the first and second feedback loops, the roll angle's estimation is obtained by means of a reduced version of a complementary filter described in [Mahony et al., 2008]. In the case of HyperRob, since only a 1-D filtering method is required, the attitude estimator becomes:

$$\left\{\begin{array}{rl} \bar{\theta}_r & = arcsin\left(\frac{Y_{acc}}{g}\right) \\ \dot{\hat{b}} & = -k_b(\bar{\theta}_r - \hat{\theta}_r) \\ \hat{\Omega}_r & = \bar{\Omega}_r - \hat{b} \\ \dot{\hat{\theta}}_r & = \hat{\Omega}_r + k_a(\bar{\theta}_r - \hat{\theta}_r) \end{array}\right. \tag{2.14}$$

where $\bar{\theta}_r$ is the roll angle calculated from the accelerometer measurement Y_{acc}, $\hat{b}$ is the estimated rate gyro's bias, $\bar{\Omega}_r$ is the rate gyro's output measurement, $\hat{\theta}_r$ is the estimated roll angle, and $\hat{\Omega}_r$ is the unbiased rotational speed. Here k_a and k_b are positive gains which were selected so as to obtain a convergence time of 3 seconds and 30 seconds for the estimated angle and the estimated rate gyro's bias, respectively.

The complementary filter therefore yields the values of the rate gyro bias, the unbiased roll rotational speed and the roll angle.

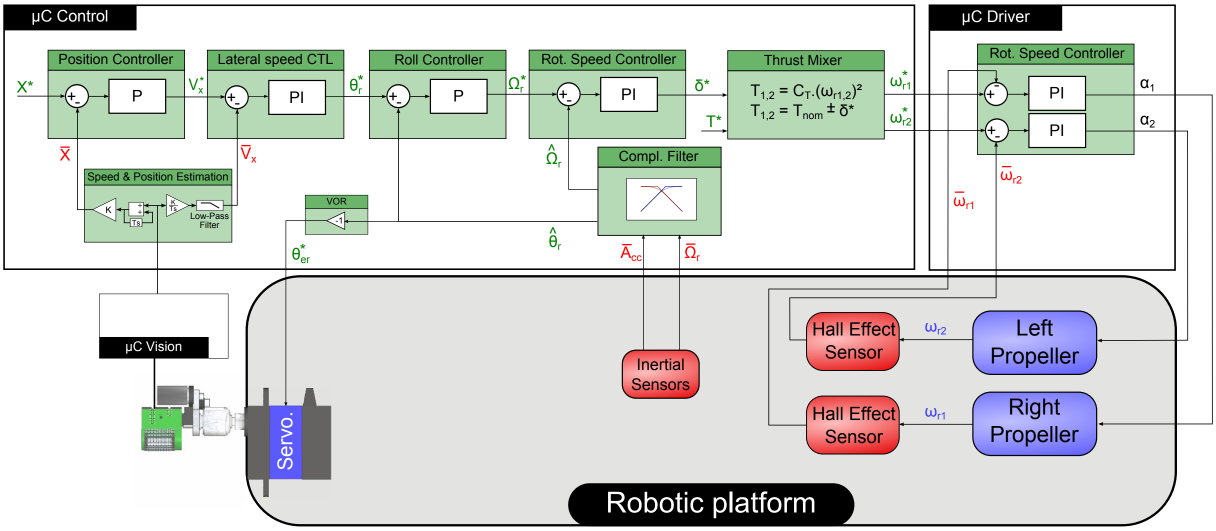


Figure 2.13: Description of the control loops
The robot's control system consists mainly of four nested feedback loops: one for the lateral position, one for the lateral speed V_X, one for the roll angle θ_r and one for the rotational roll speed Ω_r. The propellers' speeds $\omega_{r1,2}$ are controlled via an additional local feedback loop. All the controller are detailed in table 2.1. Two Hall effect sensors are used to measure the propeller speed used in the feedback loop controlling the effective thrust. The inertial sensors give a biased rotational speed $\bar{\Omega}_r$ and the acceleration $\bar{A}_{cc}$. The active CurvACE sensor produces two visual measurements, corresponding to the robot's relative position $\bar{X}$ and its linear speed $\bar{V}_X$.

2.2.6 Application to short range odometry, visual stabilization and tracking

In the various experiments performed in this study, a serial communication with a ground station was used to record the data provided by the robot. A VICON motion tracking system was run at the same time to obtain the ground truth data. A textured panel was placed $39cm$ below the robot. During the experiment involving the translation of the panel, as the sensor can only sense the movement along one direction and the robot travels along a circular path, the data were projected in order to obtain a comparable dataset. Thus, at each time step, we projected the position vector into the robot frame and took only the tangential components of the displacement.

In this section, we report on several experiments which were carried out in order to test the robot's capability to perform various tasks. In all these experiments, thanks to the efficient gaze control system compensating for the robot's roll, the visually controlled robot experienced a quasi translational optic flow elicited by the perturbations applied either to the robot itself or to the textured panel. In the first experiment, the sensor played the role of an odometer. The robot achieved accurate hovering performances despite the lateral disturbances, as well as an efficient tracking capability. All these experiments confirmed that the robot was able to perform robust short-range visual odometry, i.e., to estimate its linear position before returning automatically to an arbitrary reference position adopted.

A - Short-range visual odometry

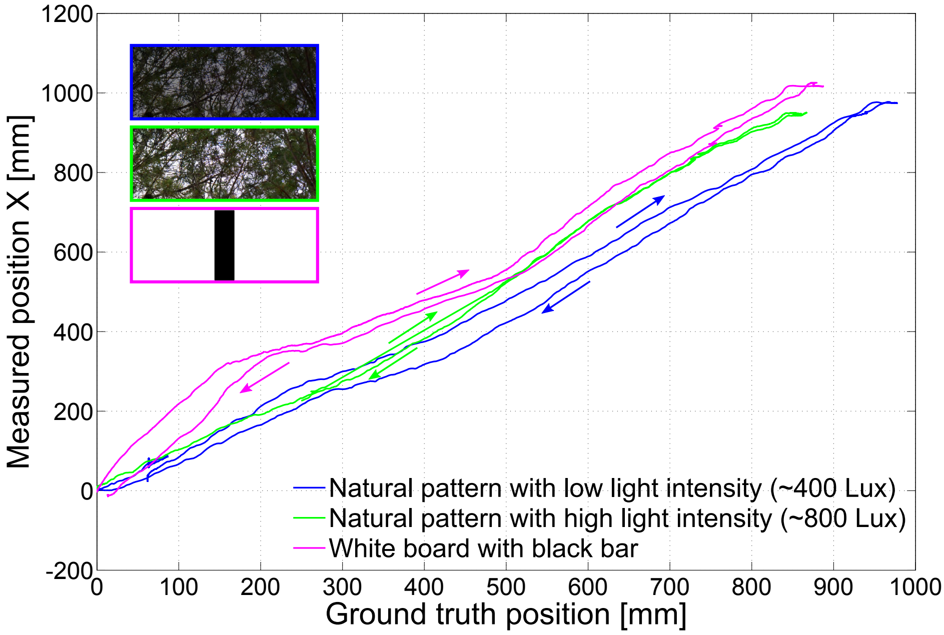


Figure 2.14: Robustness of the sensor
Comparison between the active CurvACE sensor's measurements and the ground truth position given by the VICON system when the robot made a lateral movement and returned to its initial position. The sensor's output remained fairly stable regardless the lighting conditions and structure of the pattern, giving a standard error ranging from 2.3 to 7.8% ($Deviation = \frac{Std_{error}(y_i - x_i)}{max(x_i) - min(x_i)}$).

The fused visual output signal S_{fused} provided by the active CurvACE depends on the visual environment: if there are no contrasts, the sensor will not detect any visual cues and will therefore not be able to specify the robot's position accurately. The richer the visual environment is (in terms of contrasts), the better the position measurement. In order to compare the output of the sensor with the ground-truth measurements, three experiments were conducted under different visual conditions.

In these experiments, the robot was moved manually over two different panels under two different ambient lighting conditions from right to left and back to the initial position with its gaze stabilization activated. The results obtained, which are presented in figure 2.14, are quite similar for each round trip, giving a maximum error of $174mm$. Figure 2.14 shows that the output in response to a textured panel and one composed of a single $5cm$-wide black bar was similar. Therefore, assuming the distance to the ground to be known, the active CurvACE was able to serve as a visual odometer by measuring the robot's position accurately in the neighbourhood of its initial position.

B - Lateral disturbance rejection

Above a horizontal textured panel

Lateral disturbances were applied by pushing the arm in both directions simulating gusts of wind. In figure 2.15, it can be seen that all the lateral disturbances were completely rejected within about 5 seconds, including even those as large as $40cm$. The dynamics of the robot could be largely improved by using a robot with a higher thrust or a lighter arm in order to reduce the oscillation. Figure 2.15b and 2.15c show that the robot was always able to return to its initial position. With its active eye, the robot can compensate for lateral disturbance as large as $359mm$ applied to its reference position with a maximum error of only $25mm$, i.e. 3% of the flown distance. This error is presumably due to the selection process which does not ensure to select the same features in the outward path and in the way back to the reference position or maybe the assumption of a linear approximation of the inverse tangent function does not hold entirely within the entire field of view. As a consequence, thanks to the active visual sensor and its capability to measure the *angular position* of contrasting features, the robot HyperRob is highly sensitive to any motion and thus can compensate for very slow perturbation ranging here from 0 to $391mm/s$.

Above an evenly sloping ground

In the previous experiment, it was assumed that the ground height must be known to be able to use the conversion gain in the fused output signal S_{fused}. The same experiment was repeated here above a sloping ground (see figure 2.16). The robot's height increased sharply in comparison with the calibration height. The robot's height varied in the range of $+/-74mm$ and starts with an offset of $+96mm$ compared to the calibration height. As shown in figure 2.16b, the estimation of the traveled distance was always underestimate because the robot was always higher than the calibration height. However, the robot was still able to return to its starting position with a maximum error of only $45mm$ (time $t = 36.5s$) for a disturbance of $210mm$ (*i.e.* 10.5% of the flown distance).

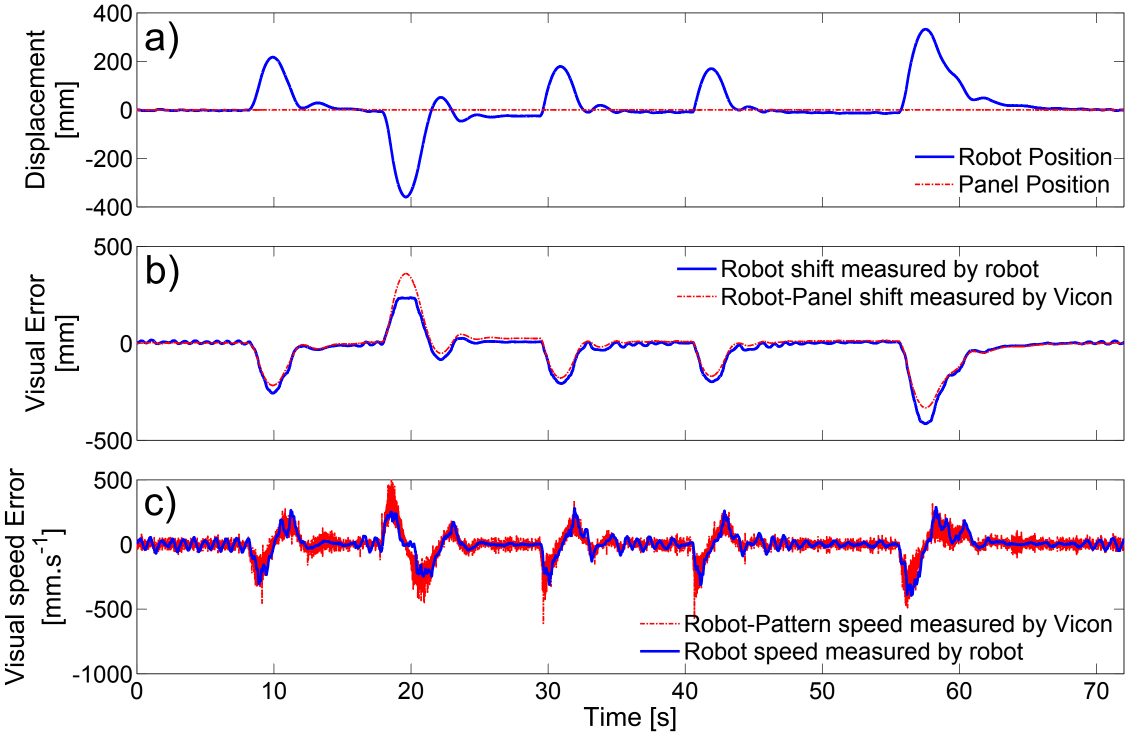


Figure 2.15: Lateral disturbance rejection over a naturally textured Panel
a) Robot's position (in blue) superimposed on the panel's position (in red), both measured by the VICON system. The robot rejected the series of disturbances and returned to its initial position with a maximum error of 25mm in less than 5 seconds.
b) The visual errors measured by the robot (red) and the VICON (blue) were very similar. With large disturbances, small errors occurred in the visual estimation of the robot's position without noticeably affecting the robot's capability to return automatically to its starting position.
c) Ground-truth measurement of the robot speed error (red curve) and the visual speed error (blue curve) measured by the robot thanks to active CurvACE. These two curves show that the robot was able to compensate for maximum lateral speed of 391mm/s.

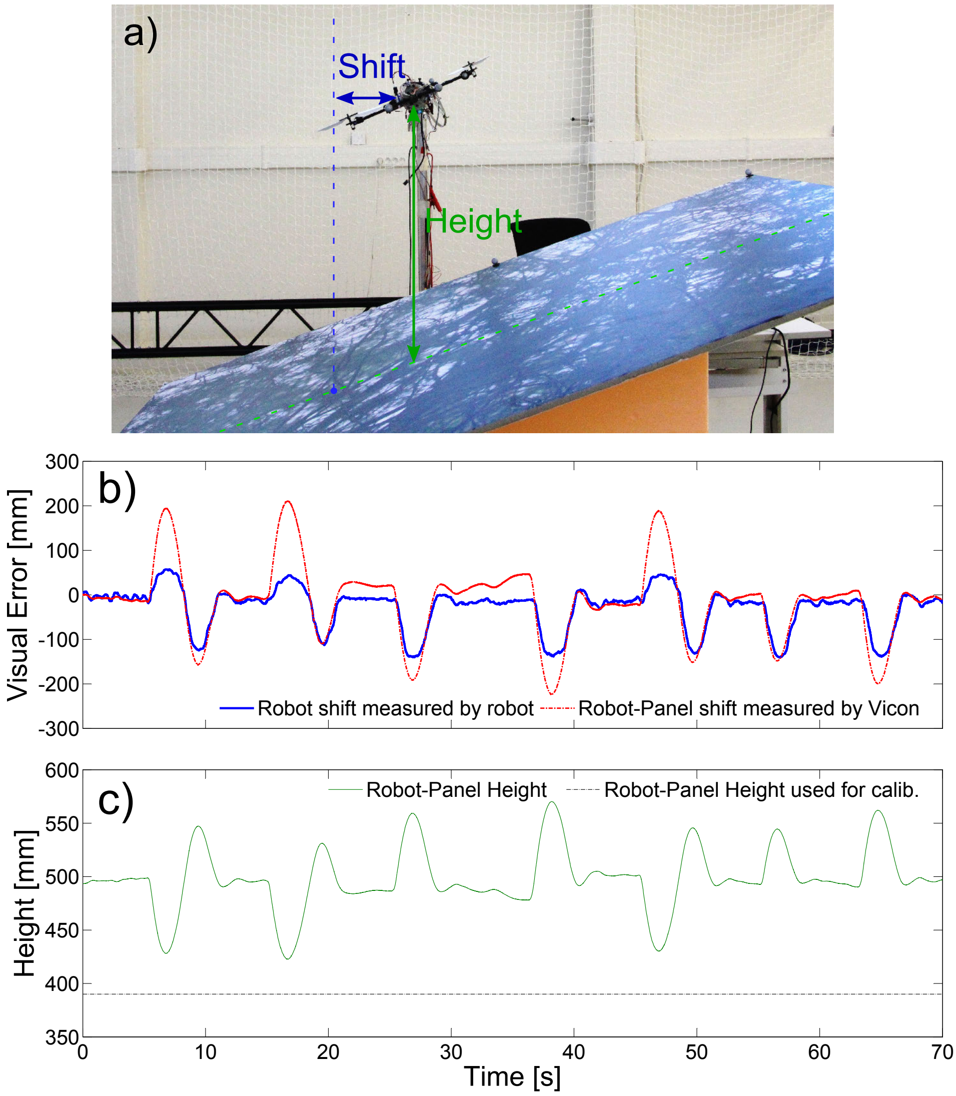


Figure 2.16: Disturbances above a sloping ground
a) Picture of the robot above the sloping ground at an angle of 18.5° with respect to the horizontal. In the case of a $300mm$ horizontal displacement, the height increased by $100mm$. Definitions of the terms Height and Shift are also displayed. The robot is subjected to a series of disturbances with a maximum amplitude of $200mm$.
b) Horizontal shift measured by the sensor (blue) and the theoretical one calculated from VICON data (red).
c) Vertical distance from the robot to the panel in comparison with the calibration height of 390mm.

C - Tracking

The robot's tracking performances are presented in this subsection. Three different tracking tasks were tested:

- tracking a moving textured panel
- tracking a moving textured panel with 3-D objects placed on it
- tracking moving hands perceived above a stationary textured panel

Panel Tracking

In this experiment, the panel was moved manually and the robot's reference position setpoint $X^\star$ was kept at zero. The robot faithfully followed the movements imposed on the panel. The few oscillations which occurred were mainly due to the robot's dynamics rather than to visual measurement errors. Each of the panel's movements was clearly detected, as shown in figure 2.17b, although a proportional error in the measurements was sometimes observed, as explained above.

Tracking a moving rugged ground

In the second test, some objects were placed on the previously used panel to create an uneven surface. The robot's performances on this new surface were similar to those observed on the flat one, as depicted in figure 2.18. The visual error was not as accurately measured as previously over the flat terrain because of the changes in the height of the ground. But the robot's position in the steady-state was very similar to that of the panel. The maximum steady-state error at $t = 19s$ was only $32mm$.

Toward figure-ground discrimination: hand tracking

The last experiment consisted of placing two moving hands between the robot and the panel. Markers were also placed on one of the hands in order to compare the hand and robot positions: the results of this experiment are presented in figure 2.19. As shown in the video in the Supplementary Data and in figure 2.19, the robot faithfully followed the hands when they were moving together in the same direction. By comparing the robot position error seen by the active CurvACE with the ground-truth error, it was established that the robot tracked the moving hands accurately with a maximum estimation error of $129mm$.

Our visual algorithm selects the greatest contrasts in order to determine its linear position with respect to an arbitrary reference position. Therefore, when the hands were moving above the panel, some stronger contrasts than those of the hands were detected by the visual sensor, which decreased the accuracy of its tracking performances. However, this experiment showed that the robot is still able to track an object when a non-uniform background is moving differently, without having to change the control strategy or the visual algorithm. The robot simply continues to track the objects featuring the greatest contrasts.

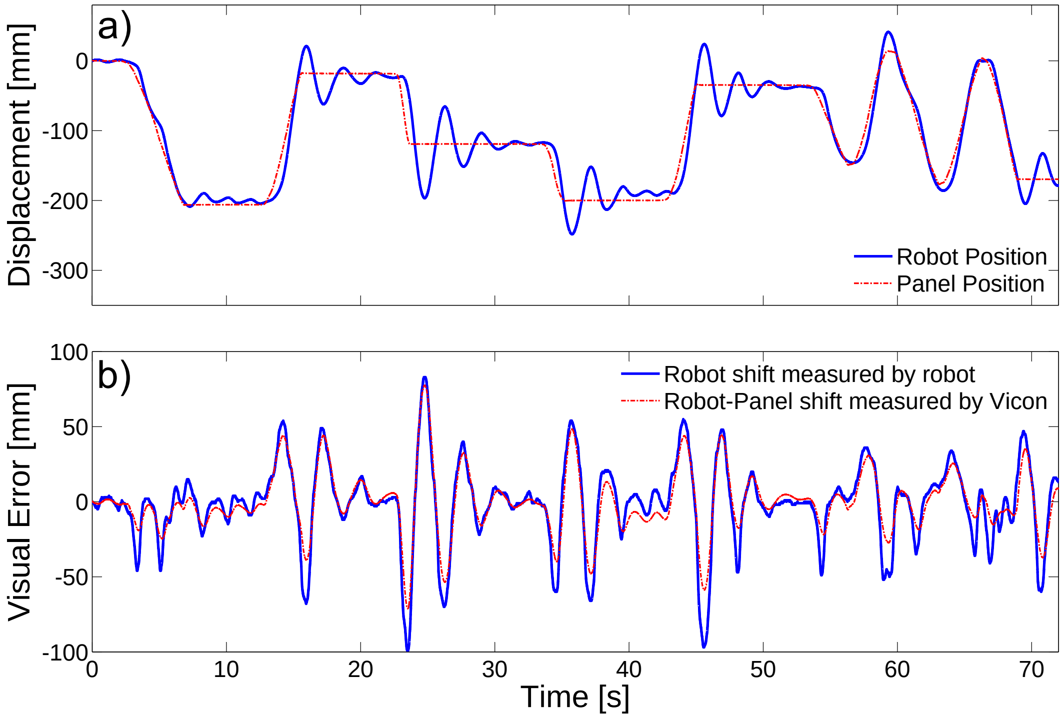


Figure 2.17: Tracking a naturally textured Panel.
When the textured panel was moved manually below the robot, HyperRob automatically followed the movement imposed by the panel.
a) Tracking of the panel by the robot. The red line corresponds to the panel's position and the blue line to the robot's position, both measured by the VICON system.
b) Comparison between the position measurement error given by the visual system (in blue) and the ground truth data (in red) given by the VICON system. The results show that the robot tracked the moving panel accurately with a maximal position estimation error of 39mm for a panel translation of 150mm.

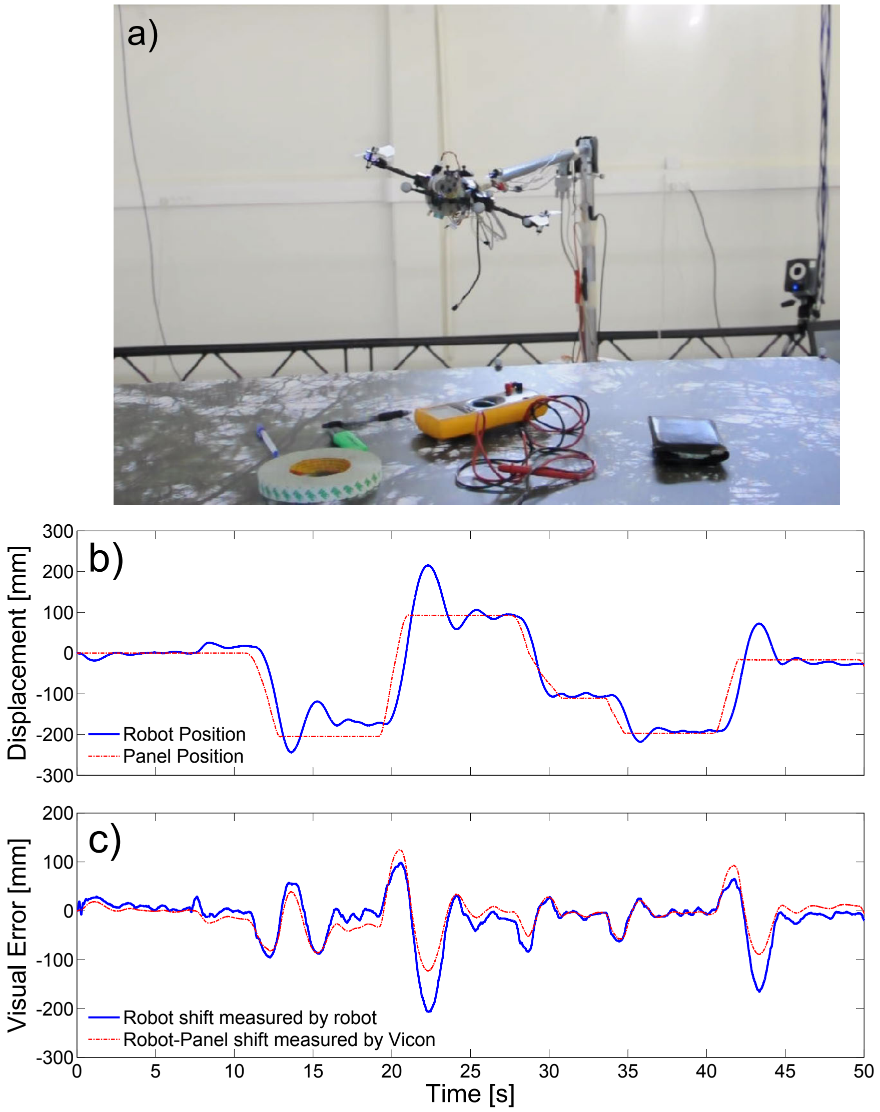


Figure 2.18: Tracking a rugged ground with height variations formed by objects.
A rugged surface including several objects was moved below the robot, which had to follow the movements imposed on the ground.
a) Picture showing the robot's visual environment during the test.
b) Tracking of the panel by the robot. The red line corresponds to the panel position and the blue line to the robot's position, both measured by the VICON system.
c) Comparison between the position measurement error given by the visual system (in blue) and the ground-truth error (in red) given by the VICON measurements.

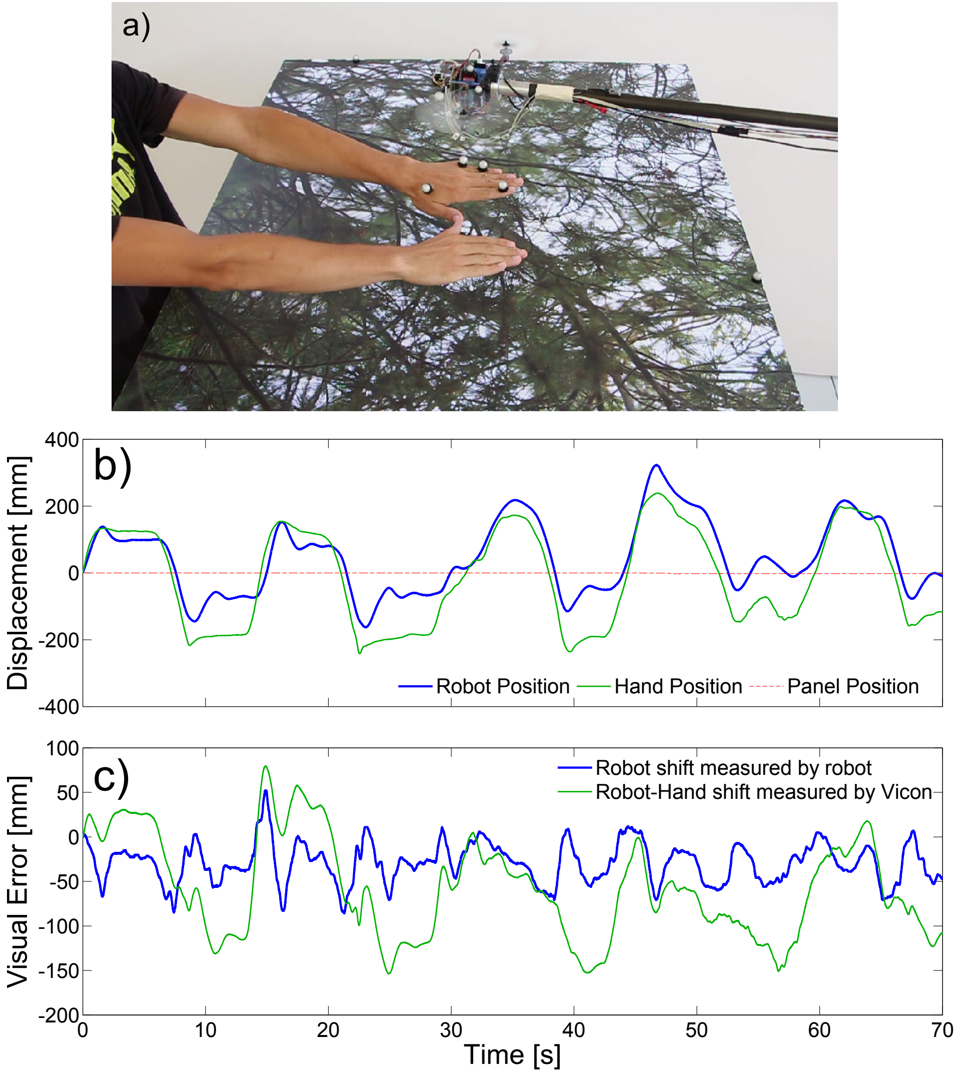


Figure 2.19: Tracking moving hands above a textured panel.
In this experiment, the textured panel was kept stationary while two hands were moving horizontally together between the robot and the panel.
a) Picture of the hands during the experiment conducted with VICON. Markers are only required to monitor the hands' position. In the video provided in the Supplementary Data, we showed that the robot's performances are similar without those markers.
b) Plots of the textured panel's position (red), the robot's position (blue) and the hands' position (green), all measured by VICON. The robot followed the moving hands faithfully over the ground.
c) Comparison between the error measured by the eye (blue), and the ground-truth error provided by the VICON system (green). The latter is equal to the (Hand_Position)-(Robot_Position).

2.2.7 Conclusion

In this paper, we describe the development and the performances of a vibrating small-scale cylindrical curved compound eye, named active CurvACE. The active process referred to here means that miniature periodic movements have been added in order to improve CurvACE's spatial resolution in terms of the localization of visual objects encountered in the surroundings. By imposing oscillatory movements (with a frequency of 50Hz) with an amplitude of a few degrees (5°) on this artificial compound eye, it was endowed with hyperacuity, *i.e.*, the ability to *locate* an object with a much greater accuracy than that achieved so far because of the restrictions imposed by the interommatidial angle. Hyperacuity was achieved here by 35 Local Processing Units applying the same local visual processing algorithm across a ROI of active CurvACE consisting of 8×5 artificial ommatidia. The novel sensory fusion algorithm used for this purpose, which was based on the selection of the 10 highest contrasts, enables the active eye (2D-FOV: 32° by 20°) to assess its displacement with respect to a textured environments. We even established that this new visual processing algorithm is a first step toward endowing robots with the ability to perform figure/ground discrimination tasks. By applying miniature movements to a stand-alone artificial compound eye, we developed a visual odometer yielding a standard error of 7.8% when it was subjected to quasi translational movements of $1m$. Moreover, active CurvACE enabled a robot to hover and return to a position after perturbations with a maximal error of $2.5cm$ for experiments based on a flat terrain, which is state-of-the-art performance in aerial robotics (see table 2.2), although our study is about a tethered robot flying indoor.
All the solutions adopted in this study in terms of practical hardware and computational resources are perfectly compatible with the stringent specifications applying to low-power, small sized, low-cost micro-aerial vehicles (MAVs). Indeed, thanks to active CurvACE, we achieved very accurate hovering flight with few computational resources (only two 16-bit micro controller and few pixels (only 8×5). However, the 1D visual scanning presented here should be extended to a 2D scanning so as to enable free flight, which, however, would require a completely new mechanical design. In addition, the architecture of the 2-D visual processing algorithm will have to be revised to make it compatible with low computational overheads. It is worth noting that the gaze stabilization reflex implemented onboard the present robot requires very few computational resources and allows CurvACE to process visual information resulting from purely translational movements. In addition, recent robotic studies have shown that gaze stabilization can be a useful means of achieving automatic heading [Kerhuel et al., 2010] and vision-based hovering [Manecy, 2014].
The MAVs of the future (e.g., [Ma et al., 2013]) will certainly require very few computational resources to perform demanding tasks such as obstacle avoidance, visual stabilization, target tracking in cluttered environments and autonomous navigation. Developing airborne vehicles capable of performing these highly demanding tasks will certainly involve the use of the latest cutting-edge technologies and bio-inspired approaches of the kind used here.

Acknowledgements

The authors would like to thank Marc Boyron and Julien Diperi for the robot and the test bench realization; Nicolas Franceschini for helping in the design of the robot, Franck Ruffier for the fruitful discussions and the help for implementing the flying arena. We would like to thank the referees for carefully reading our manuscript and for giving a constructive feedback which improved significantly the quality of the paper. We also acknowledge the financial support of the Future and Emerging Technologies (FET) program within the Seventh Framework Programme for Research of the European Commission, under FET-Open Grant 237940. This work was supported by CNRS, Aix-Marseille University, Provence-Alpes-Cote d'Azur region and the French National Research Agency (ANR) with the EVA, IRIS and Equipex/Robotex projects (EVA project and IRIS project under ANR grants' number ANR608-CORD-007-04 and ANR-12-INSE- 0009, respectively).

Appendix

Table 2.1: Controllers' parameters

Controller	**Transfer Functions**	**Parameters value**
Position controller	K_x	$K_x = 0.6s^{-1}$
Lateral speed Controller	$K_V.\frac{\tau_V s+1}{s}$	$K_V = 0.7$ $\tau_V = \frac{1}{0.7}$
Roll Controller	K_θ	$K_\theta = 0.6$
Robot Rot. Speed Controller	$K_\Omega.\frac{\tau_\Omega s+1}{s}$	$K_\Omega = 0.06$ $\tau_\Omega = 0.45$
Motor Rot. Speed Controller	$K_\omega.\frac{\tau_\omega s+1}{s}$	$K_v = 0.9$ $\tau_\omega = 0.05$

Table 2.2: Hovering Performance of different aerial robots

Reference	**Visual sensor**	**Resolution** (pixels)	**Root Mean Square Error [mm]**	**Altitude [m]**
[Kendoul et al., 2009]	*RangeVideo* KX-171	320×240	3D max error $\pm 2m$	5000 (outdoor)
[Zhang et al., 2009]	Firefly MV	320×240 (FOV 56 × 38°)	$ErrorMax_x = 200$ $ErrorMax_y = 200$ $ErrorMax_z = 200$	1.2
[Engel et al., 2012]	2 cameras Parrot AR.Drone	320×240 (FOV 73.5 × 58.5°) 176×144 (FOV 47.5 × 36.5°)	$\sigma_{3D} = 49$	?
[Lim et al., 2012]	Optical Mouse Sensor	30×30	$\sigma_x = 91$ $\sigma_y = 84$ $\sigma_z = ?$	≈ 0.3
[Yang et al., 2013]	Firefly MV	680×480	$\sigma_{x,y} = 38.8$ $\sigma_z = 5.7$	1
[Shen et al., 2013]	2 uEye 1220SE	752×480	$\sigma_x = 31.3$ $\sigma_y = 52.9$ $\sigma_z = 27.2$	0.9

2.3 More insights on the algorithm and its performance

2.3.1 Identified limits of the algorithm

This algorithm relies on several assumptions:

- the altitude of the robot is considered known and constant, in order to convert the angular measurement into lateral displacement.
- the whole field of view must experience pure translational movements thanks to a gaze stabilization reflex that compensates for body's rotations, or at least the same movement if a derotation is applied afterwards thanks to gyrometer measurements.

A limit of the algorithm is the non linearity of the response, caused mostly by the non linearity of the output characteristic and sometimes a lack of contrasts.

Moreover, during some tests, the responses of the robot were different to the expected behavior. It was mainly due to undesired shadows on the panel and when the latter was moved, the shadow did not and the robot was selecting this high contrast, makes it steady although it should follow the panel. The shadow problem is very complicated to be overcome without any object recognition. It is similar to track a partially masked object.

2.3.2 A practical test with a smaller FOV

The objective was to test the possibility to measure a displacement with a small FOV, in order to apply an odometry algorithm in an unknown environment with obstacles at different distances. Moreover, in the previous paper, the FOV was quite large, which makes it difficult to measure accurately movements close to the focus of expansion. Therefore, an experiment was conducted with another sensor called μEye presented in [Juston, 2013]. The sensor was based on an off-the-shelf photodiode array (iC-LSC from iCHaus Company, http://www.ichaus.de) consisting of 2 rows of 6 photodiodes. To improve the Signal-to-Noise Ratio, the photosensors in each column were paired in order to increase the sensitive surface. The photodiode array was mounted behind a defocused lens, to create the Gaussian shape angular sensitivity. The electronic of the sensor is close to the motion sensor presented in [Roubieu et al., 2011]. The advantage of this sensor compared to CurvACE is its higher sensitivity, but it is not endowed with light adaptation at the pixel level.

Figure 2.20 shows the visual sensor and the setup of the experiment. The sensor is placed at a certain distance of a naturally textured pattern. The picture was kindly provided by R.S.A. Brinkworth and D.C. O'Carroll and printed on a strip. A motor was able to make it rotate to stimulate the visual sensor with translational movements. The test bed was previously used to test a motion sensor [Roubieu et al., 2011].

An experiments was made placing the sensor at a distance of $49cm$ from the strip which was moving at a speed of $8cm.s^{-1}$, leading to an optic flow of $9.3°.s^{-1}$. The results of the modified odometry algorithm (see figure 2.21) are depicted in figure 2.22. First, the experiment was realized by taking all the pair signals into account for the odometry calculation. Then, a threshold was applied on the sum of the demodulated photosensor signals for each pair. The results are different because some low contrasts are not taken into account in the output signal anymore.

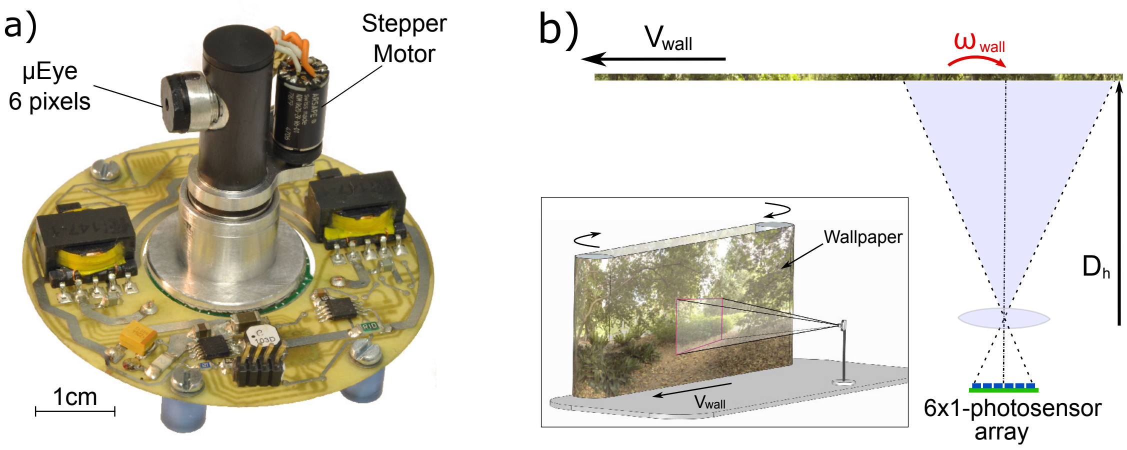


Figure 2.20: a) Picture of the μEye visual sensor. It is composed of 6 horizontal pixels placed behind a lens with a miniature stepper motor (diameter $6mm$) to apply the micro-vibrations to the eye via a miniature crank mechanism (adapted from [Juston and Viollet, 2012]). b) Setup of the experiments. The visual sensor is placed at a distance D_h from a rotating at a speed V_{wall}, equivalent to an optic flow ω_{wall}. The setup is similar to the one used in [Roubieu et al., 2011].

The output signal Θ_i, which is the integration over time of ΔP_i, is sometimes completely flat. It can also be seen by comparing figure 2.22a) and b) that the integration is not as low in both cases after $80s$. However, the Θ_{fused} output signals with and without threshold reach a close value during the same time. These results show that low contrasts are closed to noise signals but still provide information in some situations. The averaging effect is also highlighted. It should also be stressed that the output signal of the visual processing is highly sensitive to contrasts, their level and spatial frequency. Hence, the threshold value should be chosen carefully. As a reminder, in the robotic experiment, it was the ten pairs that see the highest contrasts, which were selected for the fused signal.

In the figure 2.22a, the different slope of the Θ_i signals of different pairs was observed. It is not very clear why, but I suppose it is due to the orientation of the light source which was not perpendicular to the strip. It could also be due to a misalignment of the sensor, which is not exactly horizontal and oriented at 90° of the strip. The difference of the sensitivity of the photodiodes could not be the source of this asymmetry because as shown in figure 2.23, they have similar shapes and amplitudes.

As it can be seen in figure 2.22, the visual processing algorithm when combined with a 6-pixel sensor provide a non-linear and even non-monotonous output signal resulting from the merge between the 5 LPUs. On the fused value the influence of the pair measurement is reduced, that is why, with a higher number of pair used in the fused output, the average is giving a reliable information. It explains that in the robotic experiment, the use of the 35 pairs are giving a sufficiently good measurement to enable the robot to stabilize itself.

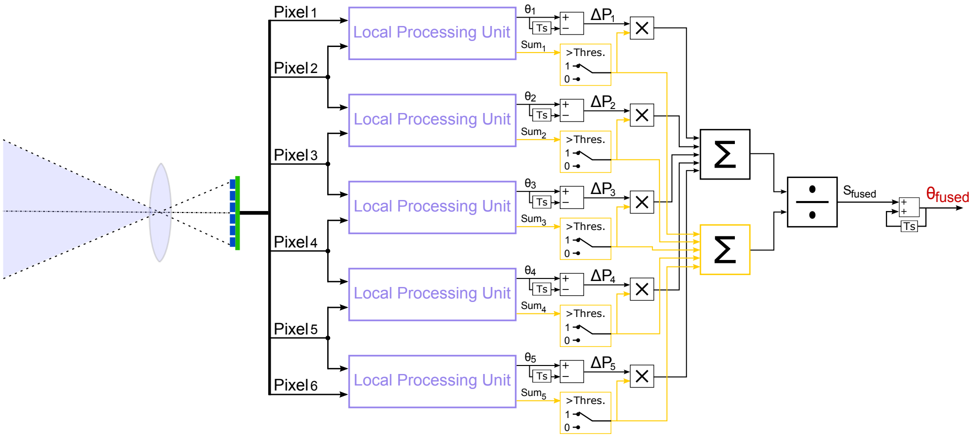


Figure 2.21: Algorithm used for the experiments. Two values were used for the threshold on the Sum. First, it was equal to 0 and then to 80000. This value is chosen after a test to have a knowledge of the amplitude of the demodulated signals. In the case, the threshold is 0, the denominator of the division giving S_{fused} is always 5 because all the pairs are contributing to the output.

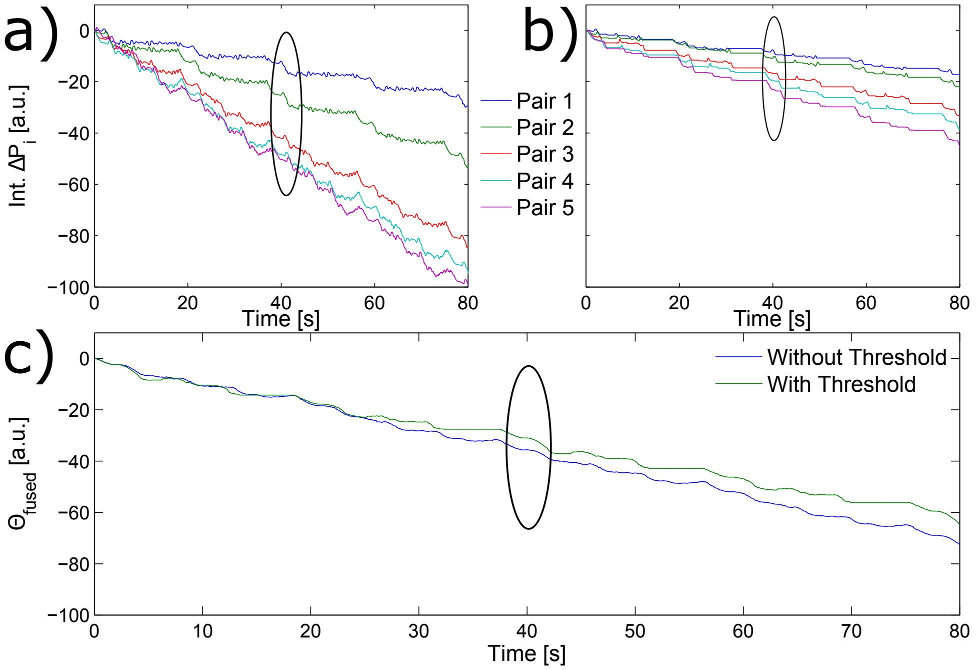


Figure 2.22: a) Θ_i is the integration over time of the respective ΔP_i signals without threshold processing b) Θ_i with threshold equal to 80000 c) Comparison of the Θ_{fused} signal of both cases. The circled areas, between $t = 38s$ and $t = 42s$, show an example of a non-monotonous measurement. It is clearly visible on a single pair output, but smooth out on the Θ_{fused} measurement. (see appendix B.2 for a zoom on this part)

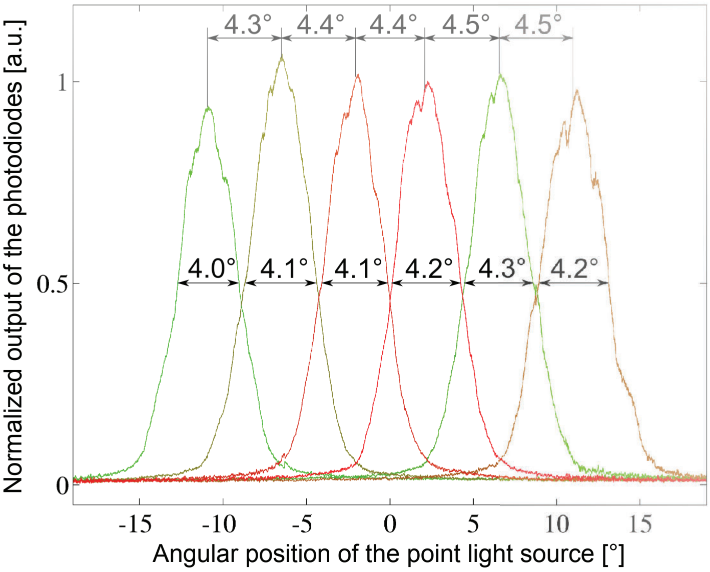


Figure 2.23: Angular sensitivity distribution measured for each photodiode of the μEye sensor, obtained by rotating the sensor in front of a point light source. The interommatidial angle $\Delta\varphi$ and the acceptance angle $\Delta\rho$ are calculated from a Gaussian approximation of the curves. From [Juston, 2013]

2.4 To go further

2.4.1 Expansion measurement

An interesting point would be to test if our hyperacute visual sensor could be used to detect an expansion and thus a possible frontal collision. This could be done by separating the FOV view into different ROI with some overlapping between the direction of motion and enough measurements for the fusion to ensure linearity, as seen in figure 2.24. Indeed, the FOV of each side should contain a sufficient number of pairs to provide a reliable measurement.

2.4.2 Odometry

The state-of-the-art nowadays for ego-motion estimation are mainly of two kinds. First, the visual odometry techniques enables to measure the travelled distance [Scaramuzza and Fraundorfer, 2011]. It consists of using features extraction to compute the motion realized between 2 frames. SLAM (Simulatneous Localization and Mapping) is using similar basis but with an increased complexity. The goal here is also to map these features acquired along the path into a depth map using either binocular cameras or a monocular camera. Usually the map obtained in the second case should be scaled to be usable. This scaling process can be realized with a motion estimation thanks to an IMU (called Visual Inertial Measurement Unit). Different approaches allows to tackle high speed motion estimation [Shen et al., 2015, Ling and Shen, 2015] or poorly textured environment [Holzmann et al., 2016].

If we want to achieve correct ego-motion estimation with the active CurvACE sensor, as done today with SLAM and visual odometry techniques, a solution should be found to compensate the drift, due to the noise. A mean of detecting a pattern should be applied to recognize the location where the robot has already been. But, the complexity would greatly increased with

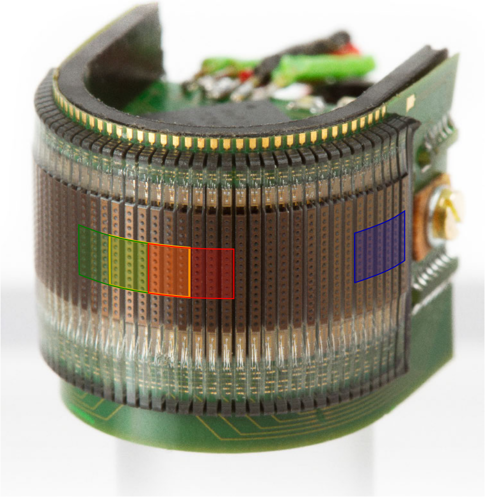


Figure 2.24: Example of different ROIs which could be used to detect translation in front of the sensor and expansion. The translation would be measured by the yellow ROI (6×7 photosensors) and the expansion would be measured by the green and red ones (6×7 both). The blue ROI (6×7) with another on the opposite side could provide a measurement of a rotation of the sensor. The addition of the yellow ROI could even provide robustness in case of complex movements.

such solution. Moreover, our current sensor disparity is also a limitation to overcome, in order to have the same response in front of the same pattern seen by 2 different photosensors. The hardware could be improved, but the software could also plays a key role to provide solutions.

2.4.3 Bio-inspired flight

In this experiment, a rate gyro and an accelerometer were used to estimate the attitude of the robot and therefore, in the VOR to ensure that the gaze direction is always vertical. Thanks to the halteres, it is possible for the diptera to sense the rotational speed, like rate gyro does [Taylor and Krapp, 2007]. However, the insect does not seem to be endowed with accelerometer, it should be done differently to be closer to the insect flight. It is suggested that the light is used as a cue for the orientation [Goulard et al., 2015].

Another solution would be to use the same visual algorithm on both side (see figure 2.24), it should be possible to have a measurement of the eye roll angle with respect to the environment. By combining the roll measured optically to the rate gyro measurement, it could be possible to use these cues in the complementary filter and replace the accelerometer in the VOR control loop. However, it is difficult to foresee the effect of a combined rotation and translation with this principle. The work of Manecy *et al.* [Manecy et al., 2015] could also be inspiring for this topic if a specific target is identified and tracked. It is quite similar to the behavior described in hoverfly [Collett and Land, 1975], which can hover when it sees a female or lands on a flower. Keep tracking a target on a textured ground is also a big challenge to overcome to be able to stabilize the robot.

But it should be kept in mind that the insects are using data fusion involving several sensors of different kind, like the compound eye, but also the ocelli which seems to be involved in gaze control and orientation detection using the dorsal light response, and antennea and hairs as airflow sensors [Taylor and Krapp, 2007].

Chapter 3

Subtended angle estimation and linear angular positioning of a bar

Contents

3.1 Summary

From the previous chapter, it has been shown that it is possible to have the angular position of the sensor compared to its environment [Juston, 2013, Colonnier et al., 2015b], because all the contrasts will move the same way in the case of a rotation. This hypothesis still hold in the case of a plane if it is sufficiently smooth to measure translation, as established in [Colonnier et al., 2015a]. However, this is no longer true for non-smooth surfaces or a forest of obstacles located at various distances from the sensor. The angular speed of the features in the FOV will not move at the same speed. An average is therefore not possible. It is also difficult to have a reliable measurement if the visual environment is not textured enough in the sense that if a few numbers of LPUs see a contrast, the displacement estimation is not very accurate. Therefore, the ability to accurately measure the displacement of features in a small area of the FOV is necessary. For example, in the case of detecting a frontal obstacle like a tree, the displacement of the contrasts between the trunk and the background should be measured. The use of vibration could be biologically plausible in order to track a moving target [Viollet, 2014] in the frontal part of the field of view.

It has been established that the measurement of an edge position does not vary according to the distance. A single pair with an appropriate Look-Up Table (LUT) are enough to localize edges according to the $Output_{Pos}$ response (see figure 2.4 for the definition). However, for a bar with a small width, it is no longer true because the $Output_{Pos}$ response does vary with the distance.

The aim of the work developed in this chapter is to describe an algorithm to assess the azimuthal position of a bar and its size in the field of view of the sensor, *i.e.* its subtended angle. It would be applied for a small bar in diameter with a subtended angle inferior to $2\Delta\varphi$. The idea is to use the measurements of 2 neighboring pairs, in addition to a proper calibration of the $Output_{Pos}$ according to the position and subtended angle of a bar. An algorithm can therefore be designed to estimate these parameters in a bar localization task. Compared to previous studies carried out at the laboratory, the proposed algorithm dramatically improves the linearity of the sensor in terms of bar localization. The results obtained in this chapter were good in steady conditions. But the subtended angle measurement was not reliable enough in a tracking situation, though the angular position was well-measured.

3.2 Observation

Based on the observation made and described by L. Kerhuel [Kerhuel, 2009] (see figure 3.1), it was observed that the response of the difference over the sum of two neighboring photosensors submitted to a vibration and demodulated can lead to two different responses according to the pattern in the field of view. This output was previously named the S_{VODKA} response; $VODKA$ stands for Vibrating Optical Device for the Kontrol of Autonomous-robots [Kerhuel, 2009] and referred to $Output_{Pos}$ in this thesis according to figure 2.4.

It can be seen that the $Output_{Pos}$ response, in the case of a bar in the FOV (figure 3.1a), can be divided in 3 parts. On the middle part where ψ_c is between $[-\frac{\Delta\varphi}{2};\frac{\Delta\varphi}{2}]$. The response is monotonously decreasing for subtended angle bigger than 3° in the example. On the left and the right, on the spacing $]-\infty;-\frac{\Delta\varphi}{2}]$ and $[\frac{\Delta\varphi}{2};+\infty[$ respectively, the response is monotonously increasing in every case. The shape, however, is evolving, from a hyperbolic tangent function for large subtended angle to hyperbolic function for small ones.

The idea that will be developed here is to use 4 photosensors, linked in 3 pairs, to compute both the subtended angle and the position at the same time. Indeed with 4 photosensors it is possible to have a value on the 3 parts of the characteristics of the $Output_{Pos}$ response. The left, central and right pairs are providing the right, the central and the left parts of the characteristic, respectively. Using lookup tables, it could be possible to compute a solution for the bar localization.

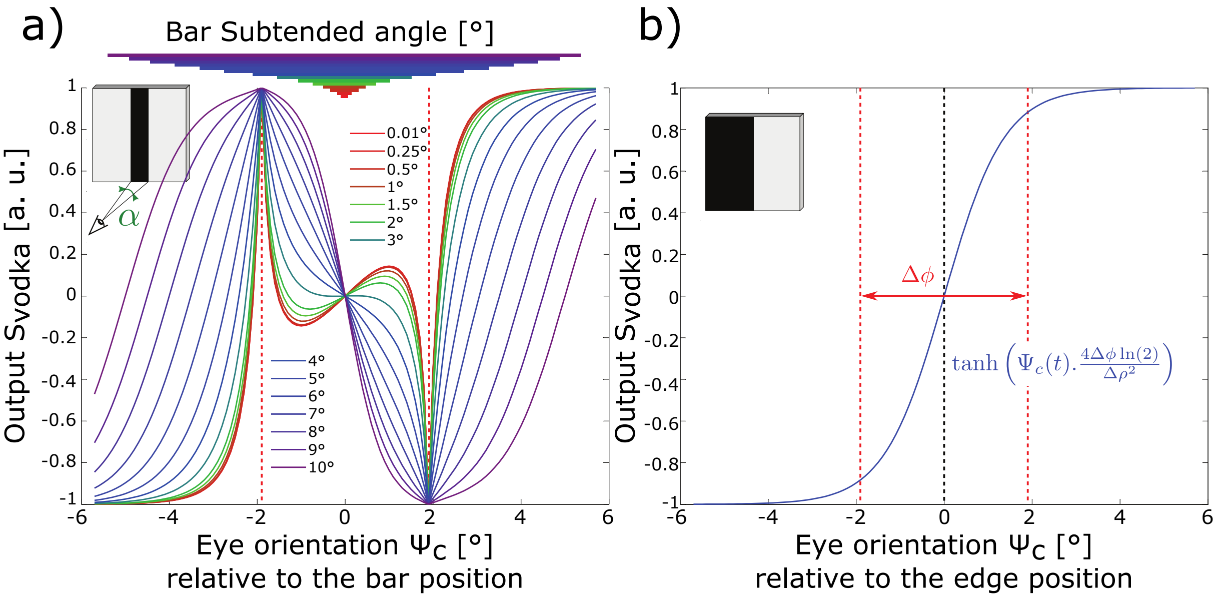


Figure 3.1: a) The simulated response of the VODKA sensor in front of a bar of width L (here $\Delta\varphi = \Delta\rho = 3.8$). The response varies according to the position of the bar and its subtended angle in the sensor FOV. It can be noticed that for very small subtended angles, the shape of the response becomes non-monotonic. A position of the bar can therefore not be measured at the output. Adapted from [Kerhuel, 2009]. b) The simulated response of the VODKA sensor rotated in front of a contrasted edge. On both figures, the red dashed lines represent the spacing $[-\frac{\Delta\varphi}{2}; \frac{\Delta\varphi}{2}]$ between the axis of 2 photosensors. These simulated response are calculated with the approximation that the demodulation is a derivative. Reprinted from [Manecy, 2015]

3.3 The algorithm

3.3.1 Calibration for the simulation

As mentioned in the previous section 3.2, the $Output_{Pos}$ can be divided into 3 parts that are described by 3 functions $f_l(\psi,\alpha)$, $f_m(\psi,\alpha)$ and $f_r(\psi,\alpha)$. A calibration is possible to identify these 3 functions for each parts of the characteristics defined in figure 3.2.

To plot the characteristics shown in figure 3.2, the responses of 2 photosensors were simulated by rotating the sensor in front of a bar. The bar's width was changed in different simulations to get the different datasets. The simulated vibration was made at a frequency of $50Hz$ with an amplitude of 2.1°. The demodulation process is the same as in the previous chapter, except that an envelop detector was added to increase the bandwidth, as done in [Manecy, 2015]. It is therefore possible to have the outputs S_{left}, S_{mid} and S_{right} of each respective function f_l, f_m and f_r and identify them.

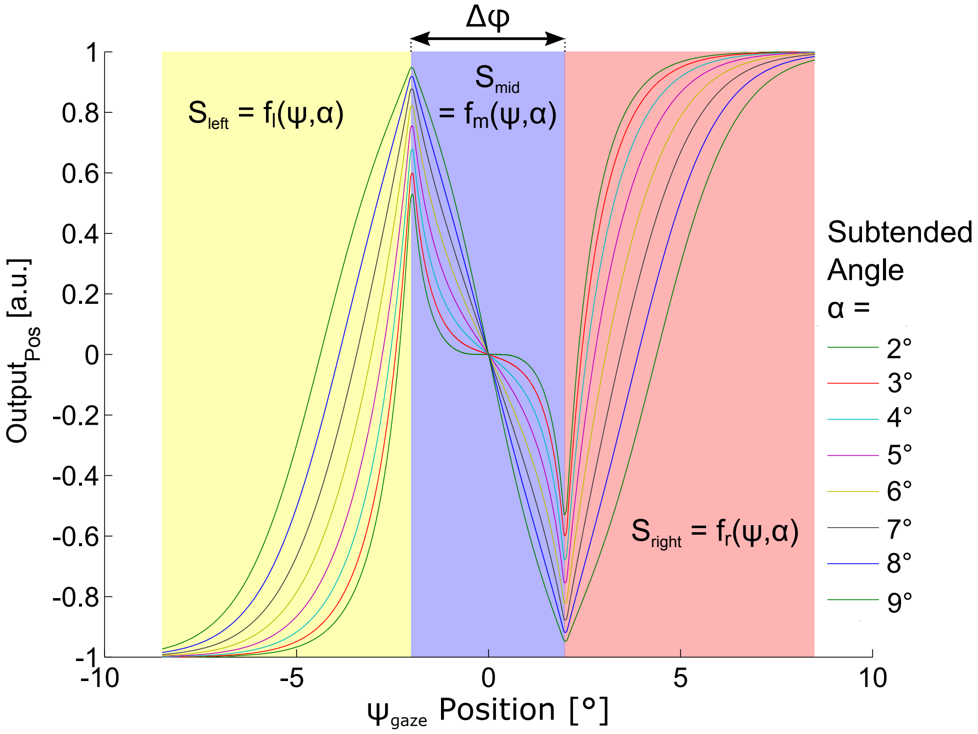


Figure 3.2: The $Output_{Pos}$ simulated response of a sensor in front of a bar of width L (here $\Delta\varphi = \Delta\rho = 4°$). The response varies according to the position of the bar ψ_{gaze} and its subtended angle α in the visual sensor FOV. Compared to the simulated results presented in figure 3.1, the assumption of the derivative is not made. Instead, the simulation uses the photosensors response with a vibration of amplitude $A = 2°$. The demodulation process is the same that would be used on the real platform. The characteristics can be divided in 3 functions $f_l(\psi, \alpha)$ (in yellow), $f_m(\psi, \alpha)$ (in blue) and $f_r(\psi, \alpha)$ (in red). These 3 functions are monotonous for $\alpha \geqslant \frac{\Delta\varphi}{2}$ on their definition domains $]-\infty; -\frac{\Delta\varphi}{2}[$, $[-\frac{\Delta\varphi}{2}; \frac{\Delta\varphi}{2}]$ and $]\frac{\Delta\varphi}{2}; +\infty[$ respectively.

3.3.2 Processing of the Output Signals to provide Angular position and Subtended Angle

With 4 photosensors, it is possible to have 3 pairs separated by an angle of $\Delta\varphi$. The 3 pairs' $Output_{Pos}$ can be associated to the value S_{right}, S_{mid} and S_{left} respectively. The pair which sees the bar in between its two photosensors can be identified thanks to the edge/bar detector described in [Juston et al., 2014]. Indeed, the bar is detected when the filtered signals of the paired photosensors are out of phase, which happens when the center of the bar is in the spacing $[-\frac{\Delta\varphi}{2};\frac{\Delta\varphi}{2}]$. The left pair would then see the bar at its right, so it gives the S_{right} signals and the same way, the right pair would provide the S_{left} signal.

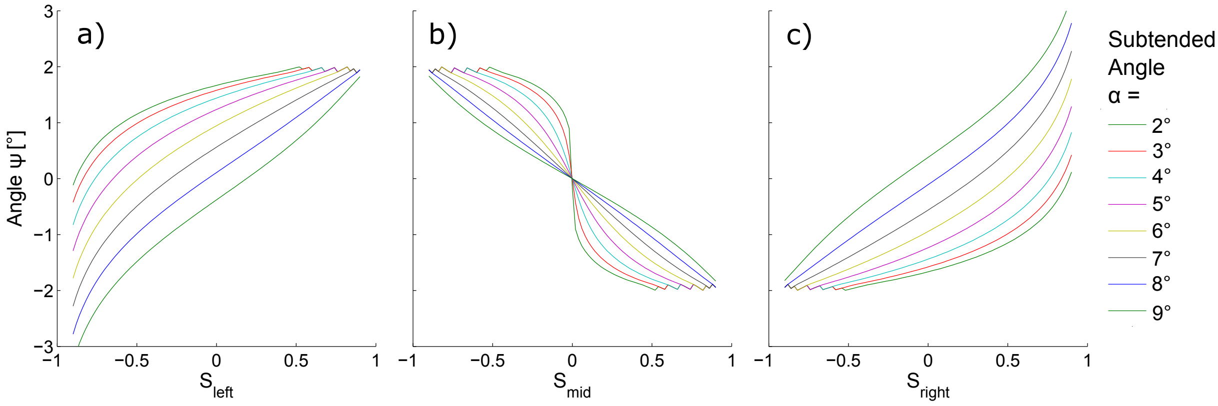


Figure 3.3: Look-Up-Tables used for the computation of the functions $h_l(S_{left},\alpha)$, $h_m(S_{mid},\alpha)$ and $h_r(S_{right},\alpha)$ in (a), (b) and (c) respectively.

After having identified upstream the invert functions $h_l(S_{left},\alpha)$, $h_m(S_{mid},\alpha)$ and $h_r(S_{right},\alpha)$ of f_l, f_m and f_r respectively (see figure 3.3), it is therefore possible to identify the parameters of the bar localization, its orientation ψ and its subtended angle α image of the distance. It should be mentioned that the angle at the output of the h_l and h_r is shifted, compared to the characteristics presented in figure 3.2, with offsets of $+\Delta\varphi$ and $-\Delta\varphi$ respectively. Indeed, the ψ angle is given according to the middle pair.

The algorithm proposed here is using the response of the middle pair and one of its neighbor. A selection is made between the left and the right pair according to the sign of the S_{mid} signal. If it is positive, the left pair is used and the right one otherwise. This selection process was chosen to use the pairs that are closer to the bar.

Then, to compute the solution, an error function Err_fcn, which is the difference of the 2 selected functions, is calculated. The Err_fcn is only depending on the α value. Here S_{left}, S_{right} and S_{mid} are the measured inputs and are therefore known. The solution of the equation $Err\_fcn = 0$ should be found to obtain α and then the value for ψ can be calculated. This algorithm is explained in the Algorithm 1.

A limitation appeared, because for some positions and because of the noise, the error function never crosses zero or sometimes does so several times. Figure 3.4 shows all the possibilities for the values of S_{mid} combined with S_{left} and S_{right} respectively. It also explains the choice of using the left characteristics when S_{mid} is above 0 and the right one in the other case. In the case where no solution are found, it was chosen to keep the old value of α and compute the ψ position with the h_m function. In the case where 2 or more solutions were found, the closest to

Algorithm 1 Pseudocode of the bar localization process

```
acquisition $S_{left}$, $S_{mid}$ and $S_{right}$
if $S_{Mid} > 0$ then
    $Err\_fcn(\alpha) = h_m(S_{mid}, \alpha) - h_l(S_{left}, \alpha)$
else
    $Err\_fcn(\alpha) = h_m(S_{mid}, \alpha) - h_r(S_{right}, \alpha)$
end if
Solve $Err\_fcn(\alpha) = 0 \rightarrow \alpha_{output}$
$\psi = h_m(S_{mid}, \alpha_{output})$
```

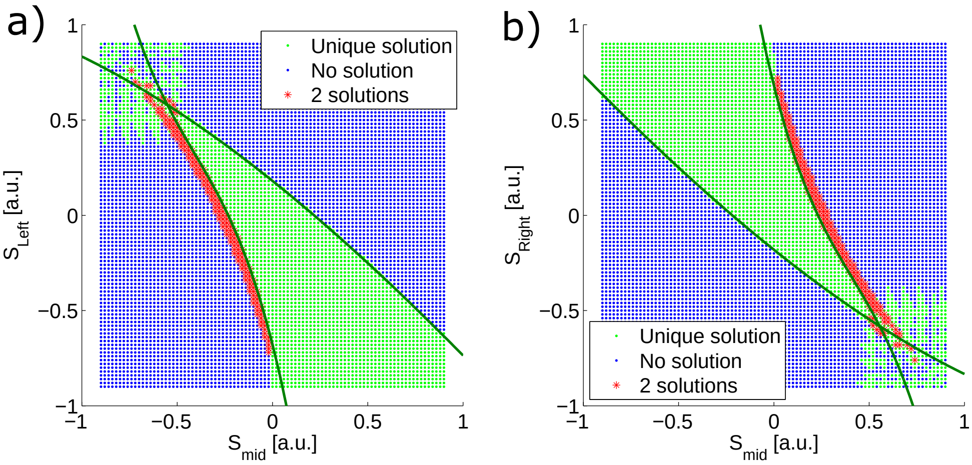


Figure 3.4: Computation of all the possibilities of combination for (a) S_{mid} and S_{left} and (b) S_{mid} and S_{right}. For each combination the Error function is calculated and the number of times the function crosses 0 is checked. In principle, a unique solution would be perfect and indicate a unique couple (ψ, α). The green line are polynomial interpolation of the boundaries between the green and the other areas when S_{mid} is higher than -0.5 for (a) and lower than 0.5 for (b).

the old α value was selected.

Remark 1 *A thought of using the interpolate boundaries occurred (see the green lines in figure 3.4), in order to check if the two measurement coordinates, (S_{mid}, S_{left}) or (S_{mid}, S_{right}), are inside the green zone. It would have perhaps enabled more robust results. But the realization was not so simple for the experimental characteristics. The green area is not as well-shaped as in the simulated one.*

3.4 Simulation Results

A simulation was made in order to test this algorithm. Some noises were added to the photosensors' signals to have relevant data and proved the possibilities to use such an algorithm, which relies on the quality of the calibration process. Figure 3.5 shows the results corresponding to two experiments: one with a constant subtended angle and a variation of the eye orientation and

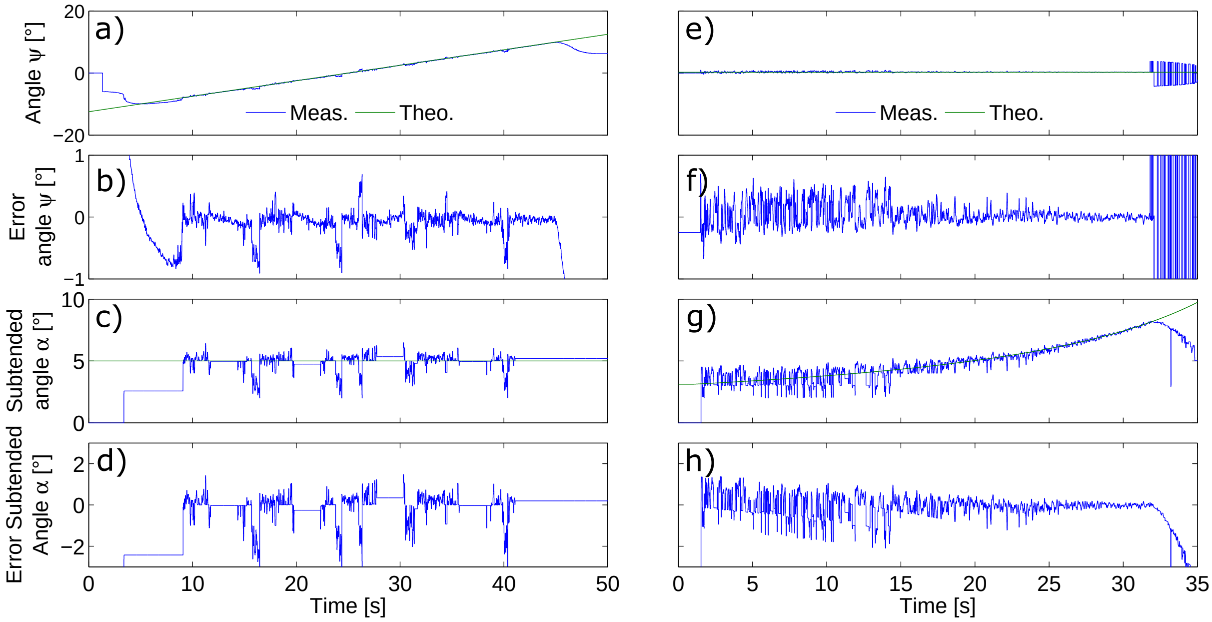


Figure 3.5: a-d) Simulation of a 6-photosensors eye rotating around its vertical axis at a constant distance of $1.5m$ from a bar. e-h) Simulation of the same eye seeing a bar at a constant ψ position equal to 0.25°, but the distance varies from $2.5m$ to $78cm$. a) and e) shows the measured angular position ψ in the sensor FOV and the theoretical one, in blue and green respectively.

another with a constant eye orientation and a varying subtended angle. The first one simulates the rotation of the eye in front of a bar and the second one the translation of the bar that goes away from the eye. The results show that the orientation is always quite well estimated with a maximal error close to 1°. However, the error of the subtended angle is a bit higher than 2°. It can also be noticed that the value is conserved between 2 pairs (figure 3.5c) because the old value is kept when no better one is available; the old one is conserved to compute the orientation value. Moreover, on figure 3.5g, it can be seen that the estimation is noisy and inaccurate because the true subtended angle is under the smaller calibrated one.

3.5 Results with Active CurvACE sensor

3.5.1 Experimental Setup

In order to assess the algorithm in a real experimental setup, a text bench was created. We used an Active CurvACE sensor mounted on a servomotor MKS DS92a+, placed in front of a vertical bar. This bar was $25mm$ in diameter and painted grey. The color was chosen to have a contrast around 70% with respect to the background, in order to stay in the linear slope of the CurvACE response during the tracking of the object (see the figure 1.17 to remember the S-Shape). The bar is also mounted on an off-centered shaft on a stepper motor. The distance between the motor axis and the center of the bar is $82mm$. The motor is able to do very small step 0.45° and a maximal speed of $270°.s^{-1}$.

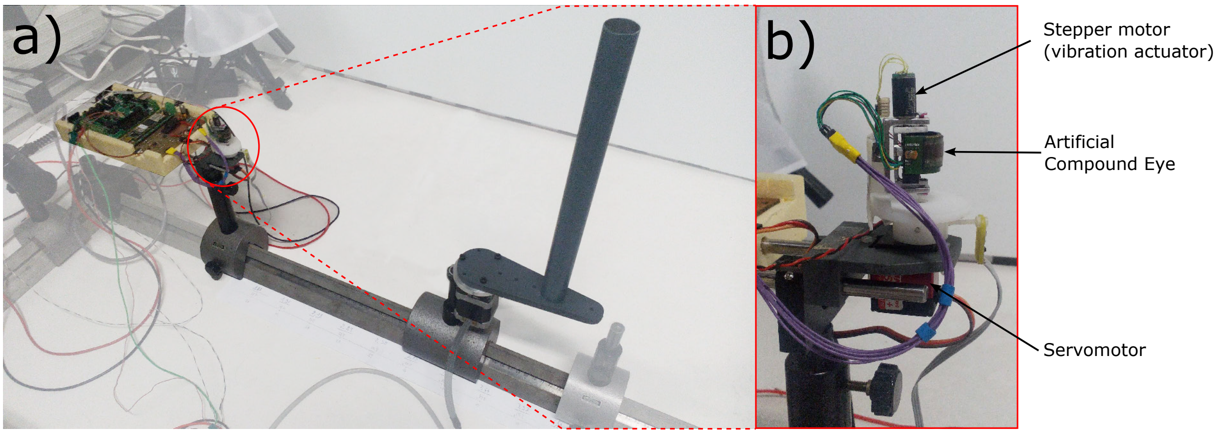


Figure 3.6: a) Experimental setup showing the Active CurvACE sensor placed in front of the rotating vertical bar. The bar can be moved and place at different distance from the motor axis to change the variation of the subtended and orientation angles. The distance between the motor and therefore the bar, and the visual sensor can be adjusted on the rail. b) Zoom on the Active CurvACE sensor with its stepper motor mounted on a servomotor.

3.5.2 Calibration

The calibration process was realized by making 2 rotations of the eye placed in front of the bar, back and forth. The bar was displaced on the rail to get several subtended angle measurement. As each pair of photosensors are not exactly identical, the identification of the response must be done for each, as shown in figure 3.7a and b). Compared to the theoretical value obtained in simulation (see figure 3.2), it can be seen that the middle part is more linear, even if the S-curve can be observed. However, with these 8 different subtended angles, it appeared that some characteristics are crossing each other and creating a singularity. Indeed at these particular points, it is not possible to distinguish the subtended angle. Moreover, it makes the response non-monotonous according to the variation of the subtended angle. A potential explanation of this phenomenon is that the bar is too small and the signal generated is not good enough, which altered the response, and that is why the small angle were removed from the calibration data. The results used are seen in figure 3.7c and d, for 2 different pairs of photosensors.

3.5.3 Localization performance

After the calibration, another tests was made to assess the precision of the algorithm. The bar used was the same (*i.e.* $24mm$) but the distances tested were different. It could be seen that the orientation measurements are close to the theoretical value. The error is always bounded in the interval $[-1;1]$. The results are also slightly different according to the movement's direction. This is mainly due to the delay in the filtering processing. However, the measurements of the subtended angle were not as good. Indeed, the precision seems usually good, except for the sixth pair when α is small. Moreover, the accuracy could be different between the pairs of photosensors used.

Figure 3.9 shows that for different ambient light conditions, the orientation measurement is quite accurate with a mean error of less than 1°. However, the subtended angle measurement is not as robust to the light variations. The left side was usually more robust to the perturbation

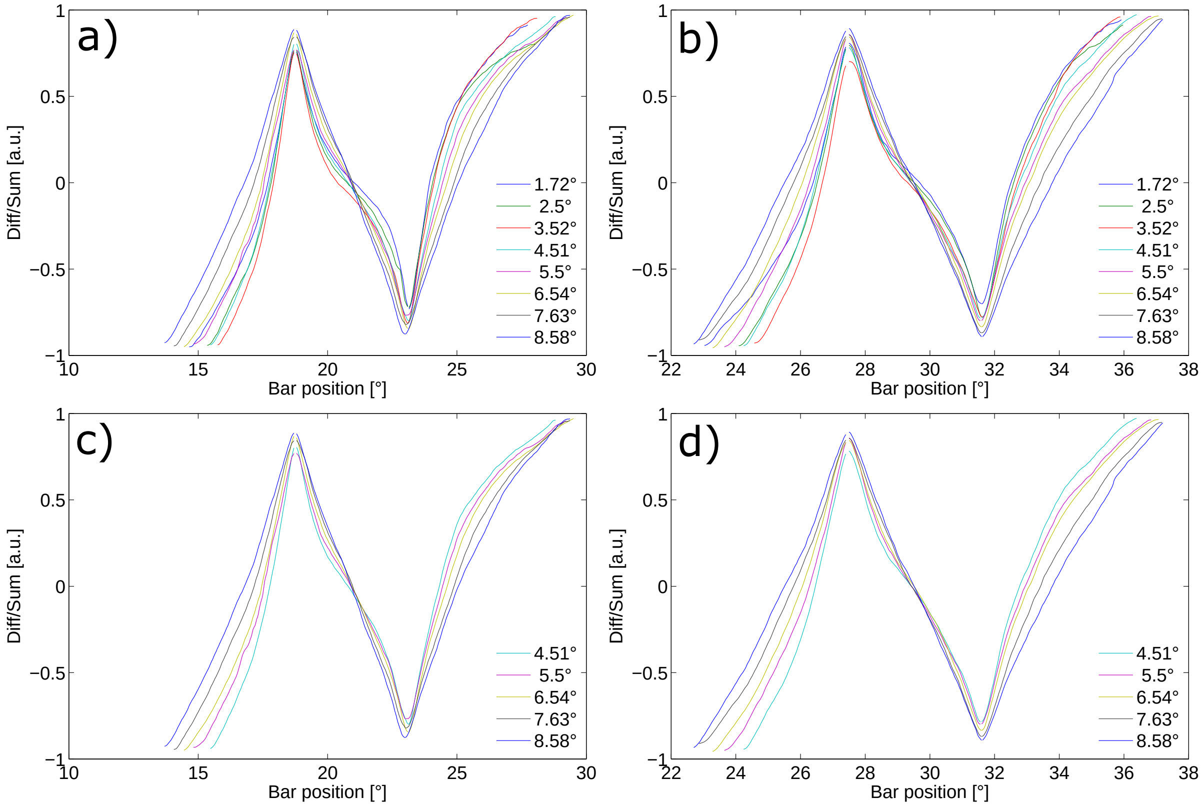


Figure 3.7: a and b) are the full calibration of 2 different pairs of photosensors. Each color corresponds to the calibrated subtended angle. It can be seen that the variation of the slope is not identical to the results obtained in the simulation, especially for the middle part which is less curved (see figure 3.2). One main problem which appeared in those characteristics is that they sometimes cross each other. It would be a problem in the resolution because it makes the whole function not monotonous according to the subtended angle. c and d) are the only calibrated data used to avoid the crossing of 2 characteristics, therefore, less subtended angle were used form the initial calibration which reduced the measurable subtended angles.

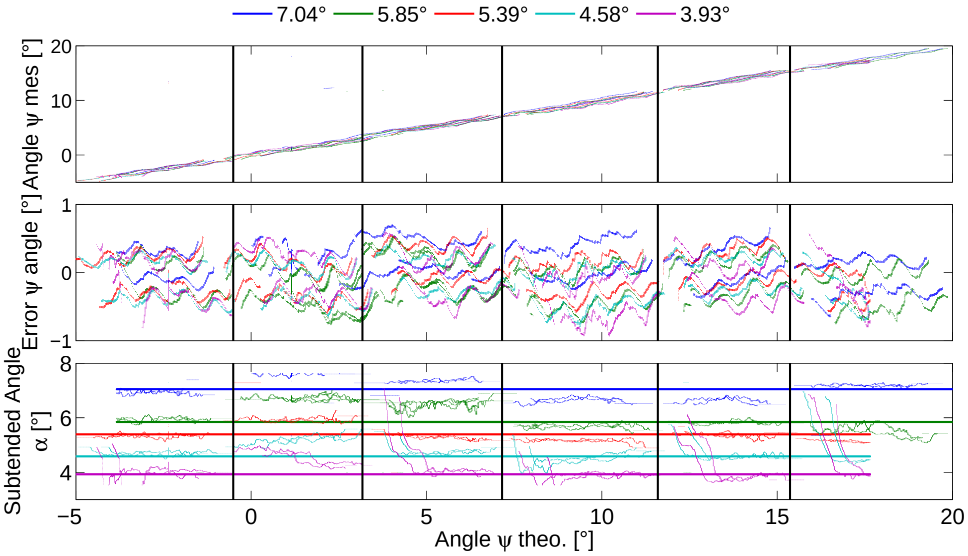


Figure 3.8: Localization performance for different subtended angles, varying from 3.93° to 7.04°. The experiment was made with one bar of 24mm width viewed at different distance. The sensor was rotating on its yaw axis to change its orientation, once back and forth for each distance. The measurements showed are made when one single couple (ψ, α) was found. a) Measurements of the orientation of the bar ψ in the FOV. b) The orientation error between the measurement and the ground truth. c) The measured subtended angle α and the theoretical in thick line. All the data are plotted according to the ground truth value. The black vertical lines depicts the mean position where the central pair is selected to compute the ψ and α values

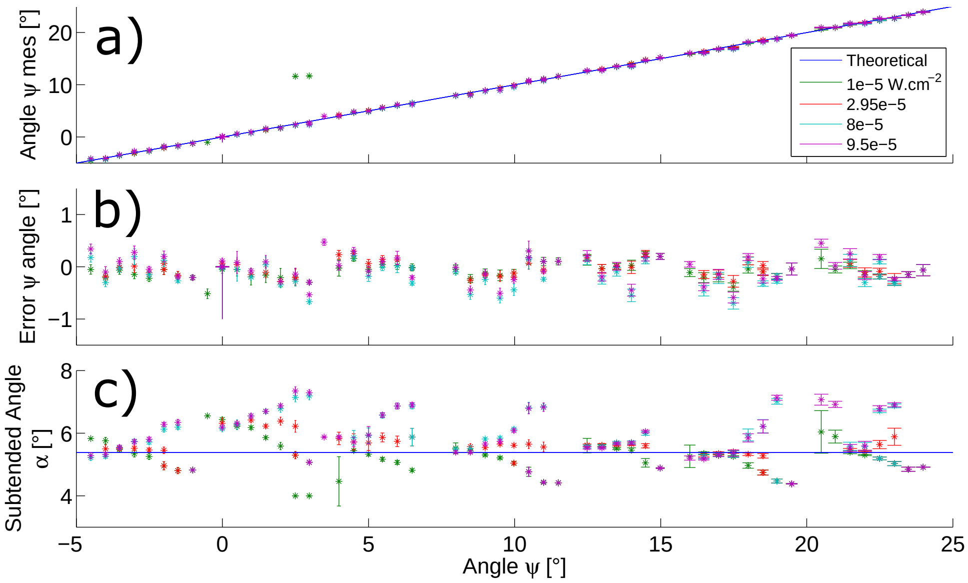


Figure 3.9: Localization performance with four different ambient light intensities, varying over one decade. The sensor is at a distance of 255mm of a 24mm wide bar. a) Estimation of the orientation of the bar ψ in the FOV. b) Orientation error compared to the theoretical values with standard deviation for a 5s steady measurements, without the two outliers. c) α estimation of the bar's subtended angle with the standard deviation.

than the right one. The results are also dependent on the pairs used for the computation. As the visual environment was the same as the one used for the calibration, it is perhaps due to a shadow that appeared behind the bar that had an influence on the measurement.

3.5.4 Tracking performance

It has been shown in the previous section that a measurement of the bar position can be done with this algorithm. From the experiments made, it was noticed that the measurements are more robust when it is close to the center of a photodiode's pair. In this situation, whatever the bar width, the S_{mid} value should be equal to 0. Therefore, the neighboring pairs are providing a value, enabling the deduction of the subtended angle.

We wanted to test if the use of a closed loop control based on the orientation of the bar could be possible and include a distance measurement as well. As presented in figure 3.6, the bar is placed on an off-centered shaft and is describing a circular trajectory when the motor is activated. The speed can be adjusted.

The results of the tracking of the bar is provided in figure 3.10. It is shown that the eye follows the bar quite well but the distance estimation is not as accurate. The errors seen in the section 3.5.3 have an impact on the measurement. Due to the delay in the measurement, the controller cannot ensure that the bar is kept in the middle of a pair. Indeed, the angular position error is higher than one degree. Moreover, when the tracking error is close to 0, the subtended angle estimation is not always accurate (see figure in appendix C). Therefore, providing a reliable measurement is not only a matter of the area. It can be also noticed that the subtended angle measurement is the worst when it changes rapidly in the active CurvACE FOV. The speed of

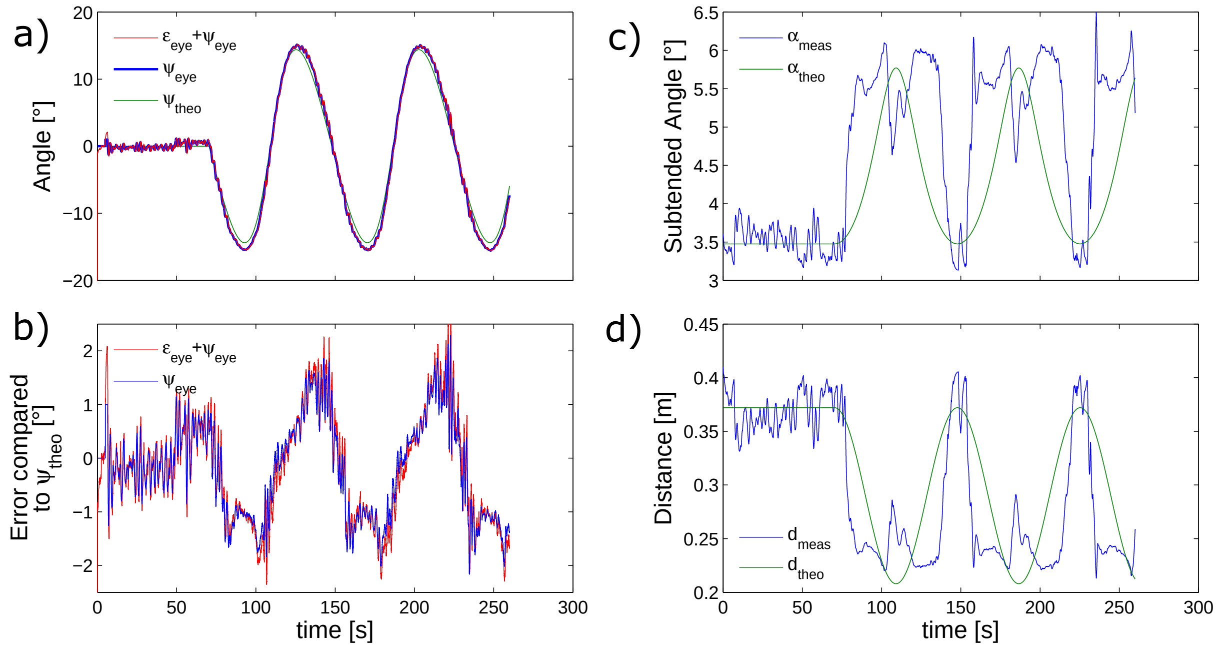


Figure 3.10: a) Estimation of the orientation of the bar relative to the eye. The green curve is the theoretical position of the bar calculated from the stepper motor command and the distance to the sensor. The blue curve is the position setpoint given to the servomotor and in red is the addition the same setpoint and the measured position provided by the sensor. b) Error of the estimated orientation and error of the eye orientation, both compared to the theoretical value, in red and blue respectively. c) Measurement of the subtended angle α of the bar in the FOV compared to the theoretical value, in blue and green respectively. d) Distance estimation from the subtended angle measurement compared the the theoretical value, in blue and green respectively.

the target is therefore a limitation in the processing. It will be later seen (see section 5.3) that the light adaptation has some drawbacks when it comes to contrasts at high speed.

3.6 Conclusion

The use of a Look-Up Table seems to be a good solution to provide more linearity and information when it comes to localize a bar with an active eye. However, the calibration process should be done carefully in a situation similar to the one where the future measurements will occur. Indeed, the visual environment and the light intensity should not be too different from the calibration ones. If a shadow appears at the back of the bar, it can modify the sensor response and see a larger bar than it is in actuality. Moreover, if the contrast on the left and on the right of the bar is different, it leads to an asymmetry of the pair's characteristic response. If this was the case during the calibration, it can be taken into account, but it can lead to measurement errors C.2.

The calibration process can be noisy and not provide monotonous characteristics which is the condition to obtain bijective functions $f_l(\psi,\alpha)$, $f_m(\psi,\alpha)$ and $f_r(\psi,\alpha)$. These results should be checked before any further computations. It should be also highlighted that the $f_m(\psi,\alpha)$ function is quasi linear in the experimental calibration compared to the various response according to the subtended angle obtained in simulation. It explains mostly the performance on the angular position measurement. The sensitivity and the quality of the sensor are also important in order to have reproducible measurements to fit to the calibration maps.

Another difficulty is that the speeds of the contrasts in the FOV have an impact on the demodulation and therefore the response. It could lead to identify another couple (ψ, α) compared to a steady measurement, at the same position and conditions.

To summarize, in a completely controlled visual environment and with a sensitive sensor, the proposed algorithm can provide good results. For some applications where the ambient light is known and constant, a sensor other than CurvACE can provide good results. In this case, the use of a vibrating sensor could be questioned, in regards to the work of Luke *et al.* [Luke et al., 2012].

Chapter 4

An alternative edge localization: application to target tracking

Contents

4.1 Summary

The previous chapter shows that localizing a small target is a difficult challenge with a compound eye with a coarse resolution. The localization of edges over a wide field of view with the VODKA algorithm requires a calibration process in order to have a linear output. The response of each pair should be assess and the offset with its neighbors need to be identified as well in order to ensure smooth measurement over the whole FOV. This chapter presents a new visual algorithm using 3 photosensors in order to localize an edge with precision without calibration. It is inspired from the Heiligenberg works on the electric fish [Heiligenberg, 1987], which compute a linear output with an array of sensor with Gaussian sensitivity as input. An application to target tracking was made to demonstrate its performance. The task is to follow a known cylinder by localizing the edges of both sides. These measurements enable to know the orientation and the distance of the target.

4.2 Introduction

The work on the subtended angle rose the following question. Does the VODKA algorithm really provide the best solution in the localization of a contrast and a bar? In the case of a contrasting edge, the response of the algorithm is well described by an invert tangent function. But for the bar, it is still not very well mathematically described and directly usable [Kerhuel, 2009]. The experiment made by Augustin Manecy [Manecy et al., 2016] in localizing a cross shows that with a knowledge of the height and a complex calibration, a look up table could provide a localization of a bar with a great linearity and resolution, though the distance to the bar should be known. His experiment shows also that its calibration is robust with respect to some height variations.

The work of R. Juston [Juston, 2013] with the selection of the largest detected contrasts, enabled to track a moving target over a textured pattern [Colonnier et al., 2015a] (work also presented in in chapter 2). But the target should be wide enough in order to have a sufficient number of good contrasts. It is a first step but could not really be used in pursuit task or obstacle avoidance, in the case of frontal objects, because the distance to objects is also considered known.

The previous chapter shows that it is possible to have a relatively reliable localization of a bar. But the estimation of its distance is really the challenge for small bar. Investigations are still needed for the localization of this kind of target. In this chapter, it was chosen to localize a wider target to reduce the challenge. We tried to change the localization algorithm to increase robustness and avoiding the need of calibration and identification process, even in the edge case. Moreover, the idea here is to avoid the drift observed during the hand tracking experiment in [Colonnier et al., 2015a], by selecting fewer contrasts.

The reading of the Heiligenberg's work was very inspiring to achieve this goal and we tried to go further in this direction by applying his model to the visual localization of an edge. Moreover, with 2 edges sufficiently separated from each other, it becomes possible to estimate the subtended angle of an object and thus to estimate its distance and maintain automatically a constant distance to the target.

4.3 Using the weighted sum process introduced by Heiligenberg

4.3.1 About the Heiligenberg algorithm

A - Presentation

During his study on the electric fish *Eigenmannia*, Heiligenberg makes the observation that an individual neuron had a general response and do not reveal the outside situation entirely. However, having a large amount of these neurons mapped together can provide an information about the stimulus with a better resolution than the one provided by the array itself [Heiligenberg, 1987]. It reveals a hyperacuity [Westheimer, 1981]. Therefore, in the same paper, he provided a model which fuses all the output of the receptive fields into one single output. His calculation is based on the hypothesis that each receptive field have a Gaussian sensibility noted $f_k(x,d)$ according to the stimuli position and defined as follows:

$$f_k(x,d) = e^{-(\frac{x-\delta \cdot k}{d})^2} \tag{4.1}$$

$f_k(x,d)$ describes mathematically the receptive field response to a stimuli according to its spatial position, where δ the spacing between receptors, d the standard deviation of the Gaussian function.

If the hypothesis of a Gaussian sensibility is verified and the standard deviation of the Gaussian function is sufficiently large compared to the space between receptors, the sum $G(x,d)$ of all the Gaussian functions weighted with their rank number in the array is approximately linear, expressed in equation (4.2) and shown in figure 4.1. To ensure a relative linearity error under 5%, the parameter d defining the Gaussian function should be greater than 7.16, when the spacing $\delta = 10$. It should be highlighted that this value is purely theoretical and was computed without any noise, neither on the response and nor on the position of each receptive field.

$$G(x,d) = \sum_{k=-N_b}^{N_b} k \cdot f_k(x,d) \tag{4.2}$$

where N_b is the number of receptive fields taken into account in the computation on both left and right side of the center position chosen.

Baldi and Heiligenberg observed the following paradox: "the wider the tuning curves, i.e. the less precise the single units, the more robust and precise is the overall computation" ([Baldi and Heiligenberg, 1988]). This is a key point but it does not take into account the noise in the measurement or the quantification problem. Indeed, the change between 2 positions (in other term, the derivative) should be sufficiently large to be able to detect the change and also to distinguish this change from the noise. Therefore, a best compromise should be found. This is also mentioned in [Heiligenberg, 1987] with the fact that a higher standard deviation value does not improve the resolution. It has a constant resolution for d between 10 and 200, but decrease for higher d values.

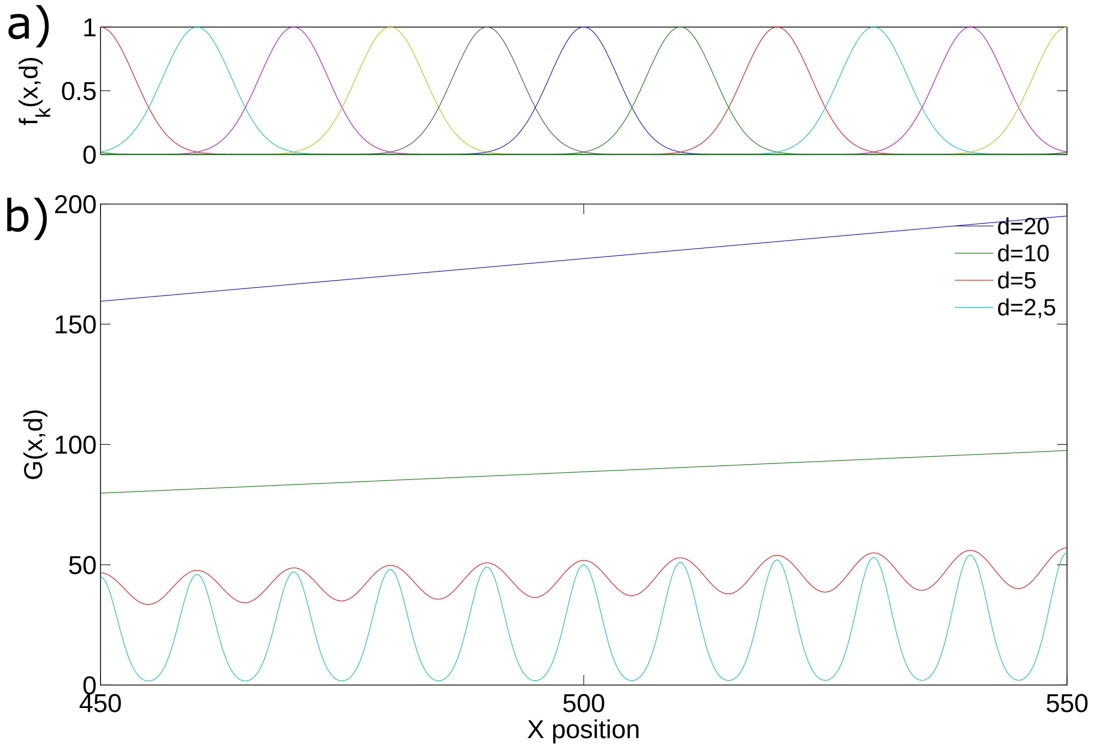


Figure 4.1: a) Simulated array of 100 receptor cells with featuring periodic Gaussian receptive fields for $d = 5$ and $\delta = 10$. b) Weighted sum calculated for 4 arrays with different standard deviations. It is clearly seen that with a large value of d, the function G is very close to a linear function. (modified from [Heiligenberg, 1987])

B - The mathematical proof

As previously proved by Baldi and Heiligenberg [Baldi and Heiligenberg, 1988], the function $G(x,d)$ could be approximated by a linear function $h(x,d) = \sqrt{\pi} \cdot d \cdot x$ in the case of $\delta = 1$. This result could be extended to $h(x,d) = \sqrt{\pi} \cdot \frac{d}{\delta^2} \cdot x$.

Remark 2 *If for a function* $f_{1k}(X,A) = e^{-(\frac{(X-k)^2}{A})}$*, the linear approximation is* $h_1(x,A) = \sqrt{\pi A} \cdot X$*. Then, by changing the variables* $X = \frac{x}{\delta}$ *and* $A = \left(\frac{d}{\delta}\right)^2$*, the result could be extended to* $h_2(x,d) = \sqrt{\pi} \cdot \frac{d}{\delta^2} \cdot x$

Although the function is never monotone, in the case of a finite number of element, the error $\mid \epsilon \mid$ is crudely bounded to:

$$\mid \epsilon \mid \leqslant \frac{2e^{\frac{-c^2}{2}}}{1 - e^{-\frac{c}{2}d}} \tag{4.3}$$

if the function $G(x,d)$ take in account only the response of $c \cdot d$ sensors on each side of x, *i.e.*:

$$G(x,d) = \sum_{k=x-c\cdot d}^{x+c\cdot d} k \cdot f_k(x,d) \tag{4.4}$$

For example, with an array of 1200 sensors, for $d = 20$ and $c = 5$, the error is under 10^{-4}

in the case of $x \in]100, 1100[$ with $\delta = 1$. This means also that 200 sensors are used for the localization calculation.

4.3.2 Adaptation to visual processing algorithms

Heiligenberg suggested its model for the electric fish. But other animals have similar sensitivity for other sensory cues. Like it has been shown previously in section 1.2.2, the fly has a Gaussian Angular Sensitivity Function (ASF) for each ommatidium. Therefore, the same principle can be applied to vision.

As in the fly's eye, the parameters used to described the layout of the eye are the interommatidial angle $\Delta\varphi$ and the angle at the half height of the ASF $\Delta\rho$ also called acceptance angle. At this step, a mathematical equivalence to the previous notation is required. It should be highlighted that the visual spatial response to an edge of the CurvACE sensor with the vibration is Gaussian according to its position.

Adding a tremor on the visual sensor is an advantage in the visual processing, as it makes the sensor the robust to the light level changes and more sensitive to contrasts thanks to the high-pass filtering [Kerhuel, 2009, Juston, 2013]. In the case of vision, the vibration is needed because of the adaptation to the illuminance and because of the high-pass filtering step. The neural response of a fly's ommatidia shows that its signal is high-pass filtered (as seen in section 1.2.2). Such response results from a movement of a contrast, the periodic vibration applied to the sensor creates such movement. Indeed, even if the sensor and the target are steady, a scanning movement of the sensor generates variation of the photosensor signals. In the following section, this response will be shown to be Gaussian according to the position of an edge, it will be called the "Edge receptive field". Therefore, the Heiligenberg algorithm could be applied.

A - Response of the Photosensors

Let consider that the ASF of the photosensors (which combine the lens and the photodiode) can be modeled by a 2D Gaussian function:

$$S(\psi, \theta) = K \cdot \exp\left(-\frac{4\ln 2}{\Delta\rho^2}(\psi^2 + \theta^2)\right) \tag{4.5}$$

where K is the amplitude, ψ, the azimuth and θ the pitch.

In order to have the volume of the 2D Gaussian function equal to one, the value of K should be:

$$K = \frac{4\ln(2)}{\pi\Delta\rho^2} \tag{4.6}$$

Thus, the photosensor response could be written as follows:

$$Ph(\psi, \theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(p, g).S(p, g)dp.dg \tag{4.7}$$

where $I(p, g)$ is the function of the light intensity received by the photosensor under an horizontal angle p and a vertical angle g.

Hence:

$$Ph(\psi,\theta) = K \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(p,g).\exp\left(-\frac{4\ln 2}{\Delta\rho^2}(p^2+g^2)\right)dp.dg \tag{4.8}$$

B - Hypothesis of one dimension only

If the visual scene is considered as one dimension, it means the light intensity from an azimuth is constant whatever the pitch. It is supposed that all the elements in the FOV are infinite in height.

This implies $I(\psi,\theta) = I(\psi)$. Therefore, the equation 4.8 becomes:

$$\begin{cases} Ph(\psi,\theta) = & K\int_{-\infty}^{+\infty} I(p).\exp\left(-\frac{4\ln 2}{\Delta\rho^2}p^2\right)dp\int_{-\infty}^{+\infty}\exp\left(-\frac{4\ln 2}{\Delta\rho^2}g^2\right)dg \\ K = & \frac{4\ln(2)}{\pi\Delta\rho^2} \end{cases} \tag{4.9}$$

But knowing that:

$$\begin{aligned} \int_{-\infty}^{+\infty}\exp\left(-\frac{4\ln 2}{\Delta\rho^2}t^2\right)dt = & \int_{-\infty}^{0}\exp\left(-\eta^2\right)d\eta + \int_{0}^{\infty}\exp\left(-\eta^2\right)d\eta \\ \text{where } \eta = & \frac{2\sqrt{\ln 2}}{\Delta\rho}t \\ = & 2.\int_{0}^{+\infty}\exp\left(-\eta^2\right)d\eta \\ = & \sqrt{\pi}\left(erf(+\infty) - erf(0)\right) \\ = & \sqrt{\pi} \end{aligned} \tag{4.10}$$

Then:

$$\begin{cases} Ph(\psi) = & K\sqrt{\pi}\int_{-\infty}^{+\infty} I(p).\exp\left(-\frac{4\ln 2}{\Delta\rho^2}p^2\right)dp \\ K = & \frac{4\ln(2)}{\Delta\rho^2\pi} \end{cases} \tag{4.11}$$

C - With an Edge in the Field of View

Mathematically, the function I expressing the light intensity, in the case of one dimension with only an edge in the field of view can be expressed as follows (as seen in equation (2.3)):

$$I(\psi) = \begin{cases} I_1 & \text{for } \psi < \psi_e \\ I_2 & \text{for } \psi \geqslant \psi_e \end{cases} \tag{4.12}$$

where ψ_e is the angular position of the edge in the field of view.

In this particular case, the photosensor response becomes:

$$\begin{cases} Ph(\psi) = & K\sqrt{\pi}. \left[I_1 \int\limits_{-\infty}^{\psi-\psi_e} \exp\left(-\frac{4\ln 2}{\Delta\rho^2}p^2\right)dp + I_2 \int\limits_{\psi-\psi_e}^{+\infty} \exp\left(-\frac{4\ln 2}{\Delta\rho^2}p^2\right)dp \right] \\ K = & \frac{4\ln(2)}{\Delta\rho^2\pi} \end{cases} \tag{4.13}$$

Knowing that:

$$\frac{1}{\sqrt{2\pi}} \int\limits_{-\infty}^{x} \exp\left(-\frac{t^2}{2}\right)dt = \frac{1}{2} \cdot \left(erf\left(\frac{x}{\sqrt{2}}\right) + 1\right) \tag{4.14}$$

where erf is the so-called error function.

and assuming the equivalence $t = \frac{2\sqrt{2\ln 2}}{\Delta\rho}p$, we have:

$$\begin{cases} Ph(\psi) = & K_2. \left[I_1. \left(1 + erf\left(\frac{2\sqrt{\ln(2)}}{\Delta\rho}.(\psi - \psi_e)\right)\right) + I_2. \left(1 - erf\left(\frac{2\sqrt{\ln(2)}}{\Delta\rho}.(\psi - \psi_e)\right)\right)\right] \\ K_2 = & K\sqrt{\pi}\frac{\sqrt{2\pi}}{2} \cdot \frac{\Delta\rho}{2\sqrt{2\ln 2}} = \frac{\sqrt{\ln 2}}{\Delta\rho} \end{cases} \tag{4.15}$$

$$\begin{cases} Ph(\psi) = & K_2 \cdot (I_1 + I_2) \left[1 + \frac{I_1 - I_2}{I_1 + I_2} \cdot erf\left(\frac{2\sqrt{\ln(2)}}{\Delta\rho}.(\psi - \psi_e)\right)\right] \\ K_2 = & \frac{\sqrt{\ln 2}}{\Delta\rho} \end{cases} \tag{4.16}$$

As it can be noticed in equation (4.16), the addition of both light intensity, I_1 and I_2, is multiplied to the whole response. It represents the amount of light received by the sensor and its effect is seen as a linear response relative to the light in the case of a linear sensor. In the case of an adaptive one and in steady state situation, it is either completely canceled [Mafrica et al., 2015] or very reduced in the case of CurvACE for example (see section 1.3.4 and especially figure 1.17). The Michelson contrast, defined as $\frac{I_1-I_2}{I_1+I_2}$, appears also in this equation. As it is multiplied by the sum of I_1 and I_2, it shows that the response is increased for identical contrast when light increased. This is highlighted in the experiment presented in figure 1.16 of section 1.3.4.

D - Modeling the photosensor response after adding tremor and demodulation process

As we are working with a sensor that can adapt to the ambient illuminance, it can be seen as a high-pass filter (see [Viollet et al., 2014] for the bode diagram of the CurvACE sensor response). But as the steady state response still does vary slightly according to illuminance (see figure 1.17), in the signal processing, a high-pass filter is also added to have a constant response relative to illuminance. It is included in the peak filter, as seen in figure 4.6

Considering that the high-pass filter of the demodulation process acts as a temporal derivative (hypothesis already used in [Kerhuel, 2009]), the photosensor response is:

$$\begin{cases} Ph_D(\psi(t)) = & Ph'(\psi(t)) = K_3 \cdot (I_1 - I_2) \cdot \dfrac{d\psi}{dt} \cdot \exp\left(- \left(\dfrac{2\sqrt{\ln(2)}}{\Delta\rho}(\psi - \psi_e) \right)^2 \right) \\ K_3 = & K_2 \cdot \dfrac{2}{\sqrt{\pi}} \cdot \dfrac{2\sqrt{\ln(2)}}{\Delta\rho} = \dfrac{4\ln 2}{\Delta\rho\sqrt{\pi}} \end{cases} \tag{4.17}$$

If a tremor ψ_{mod} is imposed to the photosensor, then, its angular position can be noted:

$$\psi(t) = \psi_{mod}(t) + \psi_c(t) \tag{4.18}$$

with ψ_c the mean angular position of the photosensor in the absolute frame. So, when $\psi_c = \psi_e$, the photosensor is in front of the edge.

First, as in our case, the tremor is added mechanically with an eccentric shaft, the modulation can be considered as sinusoidal noted $\psi_{mod} = A\sin(2\pi ft)$. Therefore, the angular position of the photosensor can be noted:

$$\psi(t) = A\sin(2\pi ft) + \psi_c(t) \tag{4.19}$$

where ψ_c is the mean angular position of the photosensor in the absolute frame. So, when $\psi_c = \psi_e$, the photosensor is in front of the edge. A and f are the amplitude and the frequency of the scanning, respectively.

The derivative of the angular position is:

$$\frac{d\psi(t)}{dt} = A \cdot 2\pi f \cos(2\pi ft) \tag{4.20}$$

In the case of the demodulation process [Kerhuel et al., 2012], only the envelop is kept. It should be highlighted that this processing is accurate only under the assumption that neither the sensor nor the environment does move during a period of scanning.

Finally, the response of the demodulated signal could be expressed as follows:

$$\begin{cases} Ph_D(\psi(t)) = & Ph'(\psi(t)) = K_3 \cdot (I_1 - I_2) \cdot A \cdot 2\pi f \cdot \exp\left(- \left(\dfrac{2\sqrt{\ln(2)}}{\Delta\rho}(\psi - \psi_e) \right)^2 \right) \\ K_3 = & K_2 \cdot \dfrac{2}{\sqrt{\pi}} \cdot \dfrac{2\sqrt{\ln(2)}}{\Delta\rho} = \dfrac{4\ln 2}{\Delta\rho\sqrt{\pi}} \end{cases} \tag{4.21}$$

The equation (4.21) shows that the faster the modulation is, the larger is the amplitude of the response. Increasing both the frequency and the amplitude lead to an increase of the rotational speed of the eye. It also highlights that the response of a demodulated signal is therefore a Gaussian response (as it will be observed in figure 4.6) and the Weighted Sum computation can therefore be applied.

Remark 3 *The demodulated signal of a photosensor submitted to a periodic vibration seeing an edge in its FOV (also later called the "Edge receptive field") is depending on the amplitude of vibration. Figure 4.2 displays this dependency. It shows that when the amplitude A of the vibration is less or equal to the half of $\frac{\Delta\rho}{2}$, the response is a Gaussian function with a Full Width* *at Half Maximum (noted d) value similar to the initial one, i.e. here* 4.2° *(see table 4.1), but with different maximum values. For an amplitude of vibration higher than* $\frac{\Delta\rho}{2}$*, the Gaussian approximation is not completely maintained and the shape changed. It is due to the fact that the edge is not any more in the FOV of the photosensor for some positions, so the modulated signals is constant at these positions. It should be highlighted that doing the assumption that the output of the modulated signal has a d value equal to* $\Delta\rho$*, is the worst case that could happen. For an amplitude under* $\frac{\Delta\rho}{2}$*, d is near* $\Delta\rho$ *and higher for higher amplitude. The only remaining requirement is to have the closest approximation as possible. Therefore, the best is to have a scanning amplitude between* $\frac{\Delta\rho}{2}$ *and* $\Delta\rho$*. The simulated effect of the vibration amplitude on the NWS calculation is described in Appendice D.1.*

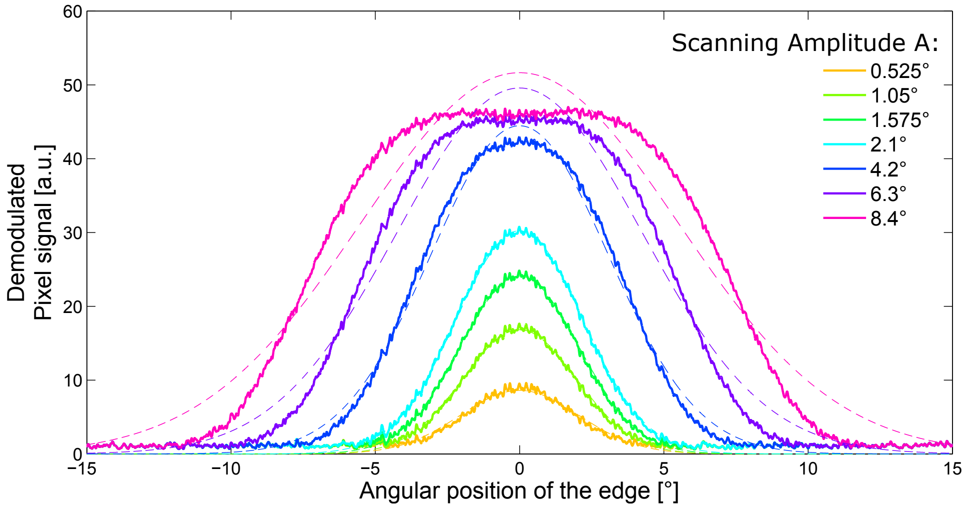


Figure 4.2: Simulated photosensor responses according to the position of an edge for different amplitudes of vibration. For the simulation, the value of $\Delta\rho$ is constant and equal to 4.2°. In continuous line, the simulated response and in dashed line, the Gaussian approximation. The legend is the amplitude value A (referenced in equation (4.19)).

Table 4.1: Estimation of the Full Width at Half Maximum of the "Edge receptive field" according to the scanning amplitude calculated from the data of figure 4.2

Vibration Amplitude A [°]	d value [°]	Coefficient of Determination R^2
0.525	4.86	0.73
1.05	4.63	0.94
1.575	4.81	0.97
2.1	5.14	0.98
4.2	7.24	0.99
6.3	9.98	0.98
8.4	12.93	0.97

E - The photosensor demodulated signals as inputs for the Weighted Sum algorithm

Heiligenberg Calculation To use the Weighted Sum algorithm, it is required to have multiple photosensors. In this case, each are separated with an angle of $\Delta\varphi$. Therefore, the response of each photosensor demodulated signal is:

$$\begin{cases} Ph_{D_i}(\psi(t)) = & K_3 \cdot (I_1 - I_2) \cdot A \cdot 2\pi f \cdot \exp\left(-\left(\dfrac{2\sqrt{\ln(2)}}{\Delta\rho}(\psi + i.\Delta\varphi - \psi_e)\right)^2\right) \\ K_3 = & \dfrac{4\ln 2}{\Delta\rho\sqrt{\pi}} \end{cases} \tag{4.22}$$

From the definition, $\Delta\rho$ is the width at half maximum of the gaussian function. It means that d and δ from the equation (4.1) are equal to $\frac{\Delta\rho}{2\sqrt{\ln(2)}}$ and $\Delta\varphi$, respectively. To make Ph_{D_i} look closer to the function $f_k(x, d)$, let Z be equal to $\psi - \psi_e$, the equation (4.22) becomes:

$$\begin{cases} Ph_{D_i}(\psi(t)) = & K_3 \cdot (I_1 - I_2) \cdot A \cdot 2\pi f \cdot \exp\left(-\left(\dfrac{Z + i.\Delta\varphi}{d}\right)^2\right) \\ K_3 = & \dfrac{4\ln 2}{\Delta\rho\sqrt{\pi}} \end{cases} \tag{4.23}$$

With this expression, it is proved that the response of a scanning photosensors' array in front of an edge have a Gaussian response. The signal amplitude is varying with the contrast and the scanning function. It can therefore be used as an input of a Weighted Sum algorithm to localize edges.

4.4 Application to the tracking of a cylindrical target

The choice of a cylinder was interesting because the image formed in the pursuer FOV does not change with rotation. As our algorithm localizes edges, only the two edges of the cylinder are tracked. They should have a sufficient contrast with respect to the background, as seen in figure 4.3. The localization of the 2 edges can lead to the estimation of the orientation and distance of the target relative to the pursuer as it will be seen in the following section.
The cylinder is also contrasted with a black and green pattern to show the robustness of the edges localization. It could be uniform with no change in the processing. The only requirement is to have a higher spatial frequency ($cycle.deg^{-1}$) than the spatial cutoff frequency of the "pixels + optic" assembly. The use of such pattern highlights also that our localization process does not rely on optic flow, because there is no change in the visual response during a rotation of the target.

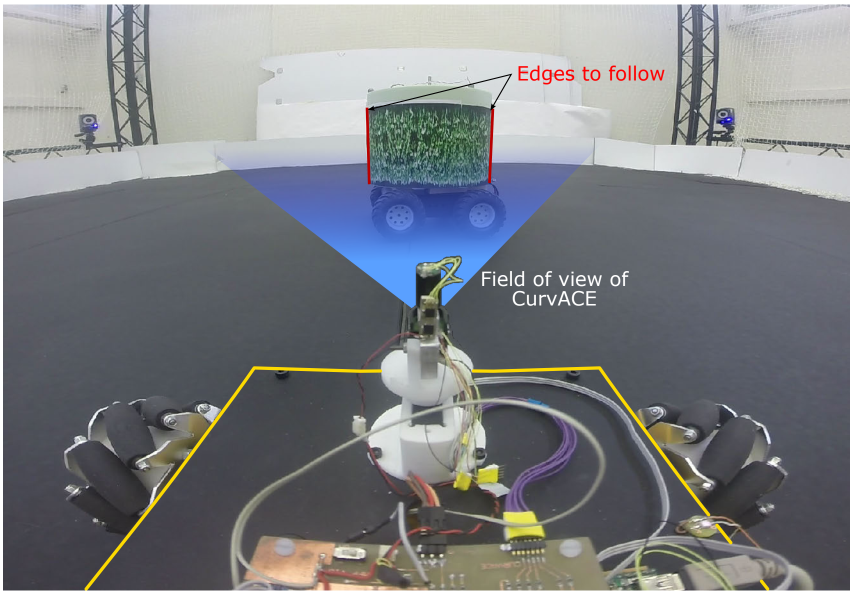


Figure 4.3: Experimental setup from the pursuer point of view. It shows the two edges of the target with the visual sensor Field of View

4.5 Article 2: Visual Micro-Scanning Makes a Robot Pursuer Capable of Robust Target Following Behavior with Hyperacuity

Description and Contribution

The material presented in this section is close to submission in a journal [Colonnier et al., 2017]. The idea of the visual algorithm comes from a discussion with S. Viollet about the Heiligenberg's observation. I pointed out that only one edge should be in the Visual Field to have an accurate response. I made and tested the solution of using only 3 photosensors to avoid this drawback. It was observed that the slope changed according to the lighting conditions and S. Viollet suggested the idea to divide by the Sum to balance this problem. Then, I made the mathematical demonstration to find the appropriate scaling. S. Ramirez-Martinez designed the control law of the robot supervised by F. Ruffier, S. Viollet and myself. I managed all the experiments. Finally, I wrote the paper with the help of S. Viollet and F. Ruffier.

4.5.1 abstract

Here we present a novel bio-inspired visual processing system, which enables a robot to locate and follow a target, using an artificial compound eye called CurvACE. In the present study, only 23 pixels were used, representing a wide Field of View of (around 97°). This artificial compound eye actively scanned the visual environment at an imposed frequency ($50Hz$) with an angular scanning amplitude of a few degrees (4.2°) and succeeded in locating a textured cylindrical target with hyperacuity, *i.e.*, with greater accuracy than that normally imposed by the pixel pitch. By placing this small, lightweight visual scanning sensor on a Mecanum-wheeled mobile robot named ACEbot, we established that the robotic pursuer was able to follow a target at a constant distance with a high level of repeatability. Thanks to the CurvACE auto-adaptive pixels, ACEbot consistently achieved similar pursuit performances under various lighting conditions.

4.5.2 Introduction

In this study, we focused on a pursuer scenario in which a holonomic mobile robot (the pursuer) tracks a mobile target, maintaining its visual contact with the target at all times and keeping a constant distance. It was assumed that the pursuer knows the size of the target, and that the speeds of the pursuer and target are bounded. Previous authors have tackled target following challenges using an omni-directional camera without any communication between the pursuer and the target, and by planning a suitable trajectory for the pursuer [Mariottini et al., 2009, Das et al., 2002, Cowan et al., 2003]. Follow-the-leader scenarios can be said to be a particular case of target tracking, as the leader is identified and can provide the follower(s) with relevant information. Some authors of robotic studies have used infrared (IR) beacons [Roberts et al., 2012, Wenzel et al., 2012, Faessler et al., 2014, Shishika et al., 2015], visual pattern recognition processes [Saska, 2015] based on the open-source code Whycon [Nitsche et al., 2015], and acoustic signals either alone [Basiri et al., 2016] or combined with radio signals to estimate the distance from a moving target [Maxim et al., 2008]. The problem of target tracking has been previously addressed in the field of computer vision (see [Wu et al., 2013] for a review). Bio-inspired approaches have involved the use of either an Event-based camera to track a moving target with a mobile robot [Liu et al., 2016] or insect-based Elementary Small Target Motion Detectors (ESTMDs) [Bagheri et al., 2015]. Many animals are capable of performing highly efficient target tracking tasks [Land, 1992]. Small insects are capable of catching prey or mates, for example, in order to survive in their ecological environment: the housefly (*Fannia canicularis*) tracks the female by consistently flying towards it, making no assumptions about its quarry's future position [Land and Collett, 1974]. Likewise, *Syritta pipiens L.* hoverflies follow their potential mates, keeping the same distance away before trying to catch them. Depending on the position of the female in its field of view, the male will perform either body saccades and sway movements when the target is located on the side, or adopt a smooth pursuit strategy [Collett and Land, 1975] when she is upfront. The male *Lucilia* can also follow its target for a while before deciding whether she will be a suitable mate. Based on field observations, flies seem to use the size of their target in their Field-Of-View (FOV) and hold its position on the frontal midline of the head [Boeddeker et al., 2003].

In this study, we used a bio-inspired artificial compound eye called CurvACE featuring an optical resolution of only a few degrees imposed by the angle between two adjacent ommatidia (photoreceptor + lenslet) [Floreano et al., 2013]. The relatively coarse optical resolution and its low sensitivity (see [Viollet et al., 2014], fig. 15) was greatly improved by applying small periodic mechanical vibrations to the whole eye, which resulted in a visual micro-scanning of the environment. Many visual sensors based on active retinal micro-movements (see [Viollet, 2014] for a review) have been used for various robotic purposes, such as enhancing edge detection [Prokopowicz and Cooper, 1995, Hongler et al., 2003] and improving obstacle avoidance [Mura and Shimoyama, 1998]. However, few studies have focused so far on the use of retinal vibrations to enhance the visual acuity of artificial sensors. Recent studies featuring hyperacuity have focused on the processing of the amplitude of the photosensor's output signals to locate an edge/bar using only 2 pixels [Kerhuel et al., 2012], to locate and track a white cross using complex calibration process [Manecy et al., 2016] or to measure robot 1D egomotion [Colonnier et al., 2015a]. The visual sensor called active CurvACE [Colonnier et al., 2015a] (i.e. the scanning version of the CurvACE sensor) was developed to locate features with hyperacuity, *i.e.* "with a greater accuracy than that corresponding to the resolution imposed by the photoreceptor's limited pitch" as defined by [Westheimer, 1981]. An eccentric mechanism was used to impose bio-inspired micro-scanning movements on our artificial eye [Juston and Viollet, 2012, Colonnier et al., 2015a]. The Active CurvACE sensor [Colonnier et al., 2015a] is characterized by 4 specific bio-inspired principles:

- local light adaptation at the pixel level,
- a similar inter-ommatidial angle $\Delta\varphi$ to that observed from the fruitfly's eye,
- a Gaussian shaped angular sensitivity function in the case of each artificial ommatidium,
- a micro-scanning movement with its visual signal processing providing hyperacuity.

Here, we introduced a new fusion algorithm which greatly improved the linearity of the angular position of the target placed in the FOV of the active CurvACE sensor. This new linear characteristic was used to make a pursuer robot follow a target. We then established that the pursuer equipped with a bio-inspired non-emissive visual sensor was capable of achieving smooth pursuit when following a mobile cylindrical target at a constant distance. Section 4.5.3 presents the experimental setup and the objectives. Section 4.5.4 describes the bio-inspired visual fusion method used to locate the cylindrical target over a large angular position range within the horizontal FOV of active CurvACE. Section 4.5.5 describes the control system and the kinematics of the pursuer robot. Section 4.5.6 shows the highly reproducible results obtained under 3 different lighting conditions in the “pursuer scenario" framework. Section 4.5.7 draws some conclusions and suggests some perspectives.

4.5.3 Experimental setup and objectives

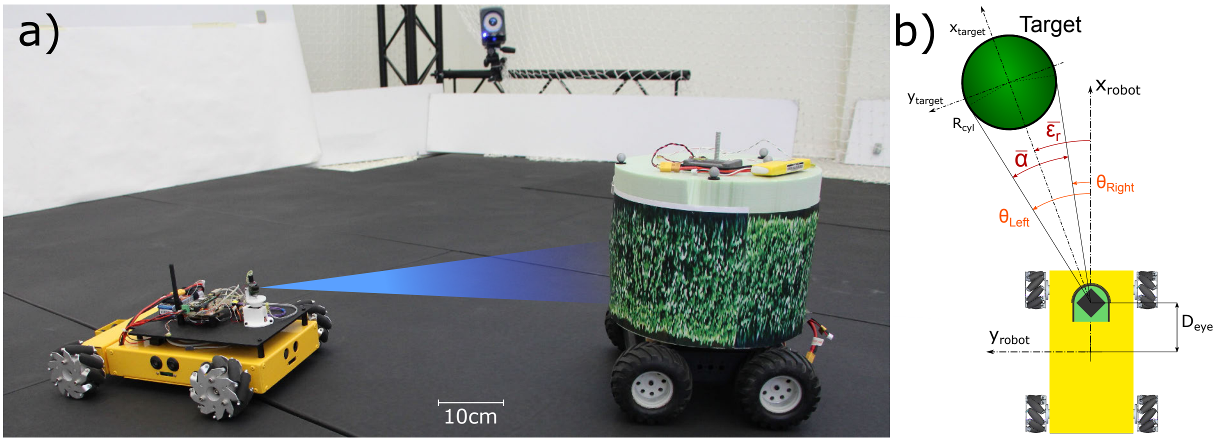


Figure 4.4: a) An experimental situation where the pursuer (*i.e.*, the yellow robot ACEbot) is following the target (a textured cylinder mounted on a moving wheeled rover) at a constant distance. Thanks to its hyperacute visual sensor, the robot ACEbot is able to lock its heading onto the target and keep a constant distance from the target. b) Scheme of the various parameters and angles used in the control strategy implemented here.

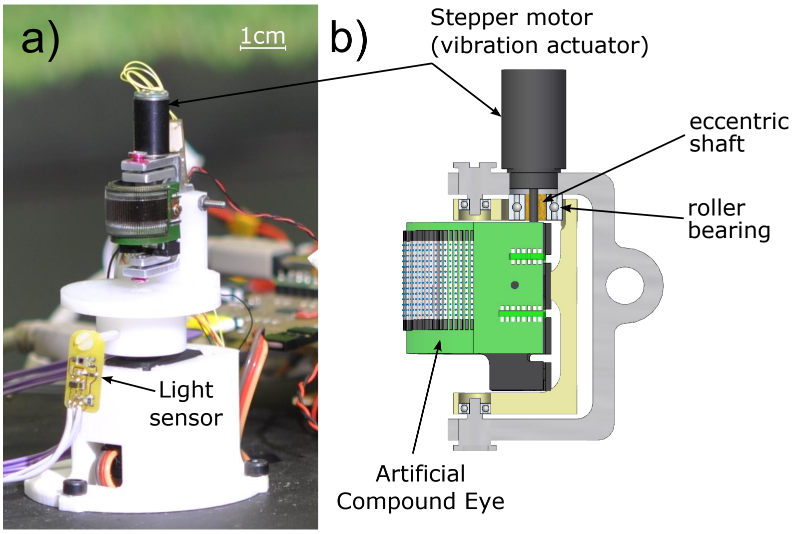


Figure 4.5: a) Picture of the ACEbot's visual sensor called Active CurvACE and the photodiode used for the illuminance measurements. b) Diagram of the mechanical system, the eccentric shaft is used with a roller bearing to limit the wear in the slotted hole. This active CurvACE assembling is very compact ($40 \times 27 \times 15mm$) and light (12.5 grams).

The pursuer equipped with a bio-inspired visual sensor consisting of only 23 artificial ommatidia (i.e., one pixel fitted with a tiny lens also called a photosensor in this paper) was able to perform smooth pursuit when following another mobile robot (the cylindrical target) at a constant distance. The FOV covered by the 23 photosensors was about 97°. As shown in figure 4.4a, the following two robots were used to test the pursuer scenario:

- The pursuer, named ACEbot, which stands for "Active Compound Eye on a robot", was a yellow rover equipped with an active CurvACE visual sensor [Floreano et al., 2013, Colonnier et al., 2015a] and Mecanum wheels (see figures 4.4 and 4.11).
- The target was a non-holonomic rover carrying a textured cylinder (see figure 4.4a). The two contrasting edges of the cylinder with respect to the background were the two visual cues used by the sighted pursuer to track the target's movement.

As depicted in figure 4.4b, the following notation is used throughout this paper:

- θ_{Left}, θ_{Right} are the angular position of each edge in the eye's FOV,
- ϵ_r is the measured retinal error, *i.e.* the position of the target in the eye's FOV,
- α is the measured subtended angle, giving the size of the target in the FOV,
- R_{cyl} is the radius of the cylinder.

4.5.4 From electric fishes' receptive fields to hyperacute visual target localization

Heiligenberg [Heiligenberg, 1987] has established that the Weighted Sum (WS) of an array of overlapped Gaussian receptive fields in an electric fish can result in a linear response. WS computations on an array of sensors with a Gaussian shaped response give an approximation of a straight line ($WS(x, \sigma) \approx \sqrt{\pi}\sigma x$, with WS defined in equation (4.34)). The validity of this linear approximation was proved in [Baldi and Heiligenberg, 1988].

The same (slightly modified) principle was used here for the first time in the context of bio-inspired artificial vision, where it was applied to a group of three in-line adjacent pixels (called a triplet) showing overlapped Gaussian-like responses of their "Edge receptive field" (see figure 4.6)c, and showed that once the WS has been normalized, this system produces a reliable output signal which can be used to locate an edge with great accuracy.

A - From modulated visual signals to Gaussian-like responses

Although the present visual sensor was based on the same mechanical principle as that described in [Colonnier et al., 2015a], only a few processing steps were used in the current visual algorithm: the principle of using a peak filter with an absolute value and a low-pass filter was still adopted as in [Kerhuel et al., 2012], but the filters were improved to increase their dynamic responses (see figure 4.6a). In addition, an envelope detector was added in order to filter the signal without increasing the delay. The main advantage of using this envelope detector rather than a classical low-pass filter [Colonnier et al., 2015a] is that it increases the cutoff frequency of the last low-pass filter (from 5 to $20Hz$), thus increasing the bandwidth of the visual processing.

To test the responses of the visual processing algorithm, the artificial eye was rotated back and forth in front of the target by activating the visual micro-scanning movements. As shown in figure 4.6c, the "Edge receptive field" of each pixel's output signal, denoted Ph_D, can be approximated with a Gaussian-like function centered at the angular position of the contrasting edge with respect to the eye (the gaze).

B - From Gaussian-like responses to angular position measurement within the triplet's FOV

Normalized Weighted Sum Computation In the case of the present visual system, the amplitude of the Ph_D signals depends on three main parameters: the ambient illuminance, the contrast and the scanning frequency. The novel feature of our approach is that it makes the visual sensor's output signal independent of these parameters by normalizing the WS by the sum of the Ph_D signals. As shown in the Appendix 4.5.8A, the output of the normalized WS ($NWS = WS/S$) is still linear and bounded. The hyperacuity of the NWS was also established because the maximum error amounts theoretically to less than 7% of the spacing between the receptive fields. This makes it possible to measure the position of a contrasting edge. However, only one edge at a time must be present in the FOV of the photosensors used for the computations at a time. It is therefore best to use as few photosensors as possible. It was decided to use only three neighboring photosensors, *i.e.* one triplet, in the NWS, because when an edge is in the neighborhood of the center of the triplet's FOV ($\pm\frac{\Delta\varphi}{2}$), it contributes more than 99% to the total Sum. The linearity error is less than 6.5% (see Appendix 4.5.8A).

To provide the pursuer with the angular position of one given edge, it is necessary to scale the NWS value, depending on the spacing between each pixel. The mathematical expression for our new algorithm applied to an array of pixels featuring Gaussian angular sensitivities separated by an angle $\Delta\varphi$ can therefore be written as follows:

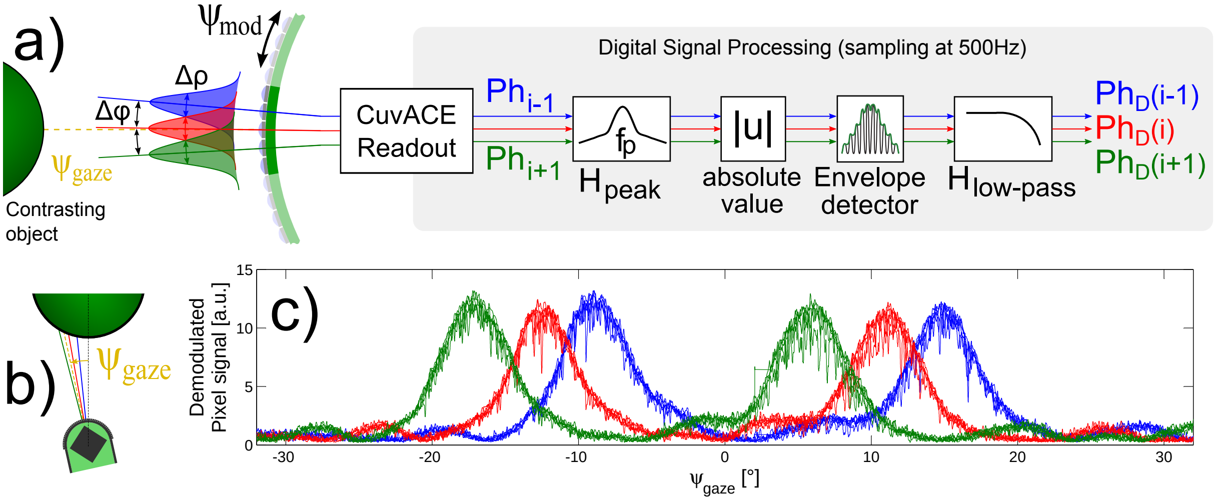


Figure 4.6: a) The demodulation signal processing algorithm applied to the visual signals delivered by the vibrating artificial compound eye ($\Psi_{mod} = A\sin(2\pi 50t)$ where $A \approx \frac{\Delta\varphi}{2}$). Three Gaussian shaped optical angular sensitivities of the Active CurvACE (defined by its acceptance angle $\Delta\rho$) are shown. The CurvACE readout circuit is in charge of digitizing each pixel's output signal here at a sampling frequency of $500Hz$. The subsequent digital demodulation steps are performed using a peak filter center at the scanning frequency cascaded with an absolute value function connected to a digital envelope detector and a low-pass filter (6 order, cut-off frequency of 20 Hz). b) Scheme of the active CurvACE sensor in front of the target. c) Normalized Gaussian-like "Edge receptive fields" (defined by their standard deviation σ) of 3 vibrating photosensors after the demodulation process during two back and forth rotations of Active CurvACE (vibration turned ON) in front of left and right edges of cylindrical target.

$$NWS(n) = \Delta\varphi \cdot \frac{\sum_{k=n-1}^{n+1} k \cdot Ph_D(k)}{\sum_{k=n-1}^{n+1} Ph_D(k)} + bias \tag{4.24}$$

where Ph_D denotes the photosensors' demodulated signals and $bias$ is a calibrated value used to set the offset.

Characterization of the visual sensor To test the responses of the hyperacute visual sensor (i.e., the active CurvACE) depending on its orientation (denoted ψ_{gaze}) with respect to a textured cylindrical target, the visual sensor was turned towards a fixed target (a textured cylinder 30 cm in diameter, placed $91cm$ ahead) and rotated stepwise about the vertical axis via an accessory position servo (not shown here) driven by a 0.1° staircase signal. The responses of each of the 21 NWS were monitored at each azimuthal orientation of the gaze ψ_{gaze} with respect to the target. As shown in figure 4.6, the various NWS were calculated by applying equation (4.24). Figure 4.7a gives the results of all the NWS calculations and shows the need for a selection process to obtain the NWS of greatest interest, which will be used to locate the target.

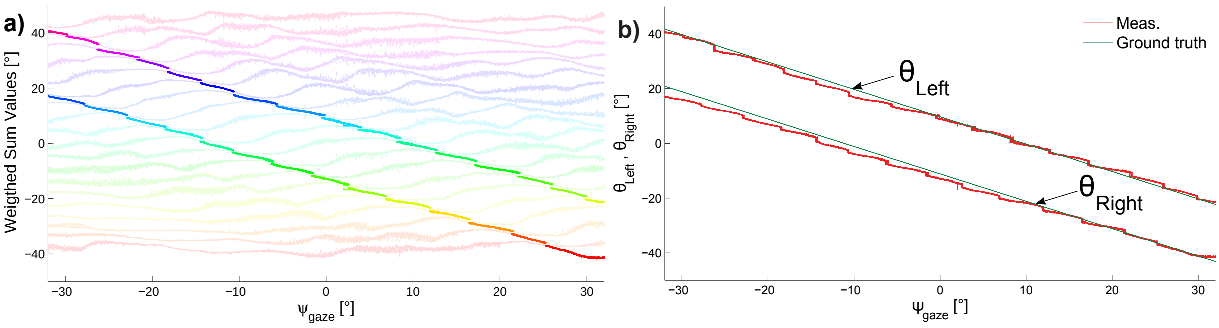


Figure 4.7: a) Normalized and scaled WS values of the 21 triplets as a function of the visual sensor's gaze (ψ_{gaze}) with respect to the fixed cylindrical target. The parts resulting from the selection process (see section C) are highlighted. b) Measured angular position of each target's edge (left and right) versus the visual sensor's gaze. Comparisons between the measured angular positions (red) and the ground truth values (green) based on a motion capture system showed that the linearity error of the sensor amounted to only 5 percent. The resolution was as small as 0.78° with a signal-to-noise ratio of $20.7dB$.

C - Weighted Sum Selection : edge localization over a large angular position range

As shown in figure 4.7a, a selection process was required to obtain the NWS value nearest to the real value. This process was performed using the following criteria:

$$S_{Criteria}(i) = Ph_D(i-1) + 2 \cdot Ph_D(i) + Ph_D(i+1) \tag{4.25}$$

The index value n corresponding to a given contrast is then updated as follows:

$$n = \underset{i \in [n-1, n+1]}{\arg\max}\ S_{Criteria}(i) \tag{4.26}$$

The new triplet index n indicates the current NWS signals, defined for the cylinder edge measurements as follows:

$$\begin{cases} \theta_{Left} = & NWS(n_{Left}) \\ \theta_{Right} = & NWS(n_{Right}) \end{cases} \tag{4.27}$$

This selection process was based on the assumption that the target is not moving faster than $\Delta\varphi$ between two successive sampling times. This means that an edge can only be located either via the same selected NWS or via its nearest neighbor ($n-1$ or $n+1$). In this study, the sampling frequency of the measurements was $500Hz$. The selection process can be applied to any number of edges to be followed (in the present case, two edges).

D - Application to Target localization

The two measured angular positions resulting from the selection process are denoted θ_{Left} and θ_{Right}.

This algorithm makes it possible to locate 2 contrasting edges at the same time if they are separated by a sufficiently large angle, *i.e.* $2\Delta\varphi$, in line with the hypothesis that only one contrast can be located at a time in the FOV of 3 photosensors. Assuming that these two edges belong to the same target, it is therefore possible to calculate its orientation $\bar{\epsilon}_{er}$ and its subtended angle $\bar{\alpha}$ defined as follows:

$$\begin{cases} \bar{\epsilon}_r = & (\theta_{Left} + \theta_{Right})/2 \\ \bar{\alpha} = & \theta_{Left} - \theta_{Right} \end{cases} \tag{4.28}$$

Table 4.2 sums up the Active CurvACE performances.

4.5.5 Nonlinear Control and Kinematic of the pursuer robot

The artificial eye was placed on a robot equipped with Mecanum wheels (see Appendix 4.5.8 for details), which allows omnidirectional movements. The maximum speed that the robot was able to reach was equal to $0.5 m.s^{-1}$.

The control laws implemented onboard the pursuer were drawn up using the kinematic model [Tsai et al., 2011, Viboonchaicheep et al., 2003]. The control strategy implemented was based on a non-linear approach to control each degree of freedom and a bounded control strategy adapted from [Guerrero-Castellanos et al., 2014] to control the 4-wheeled robot.

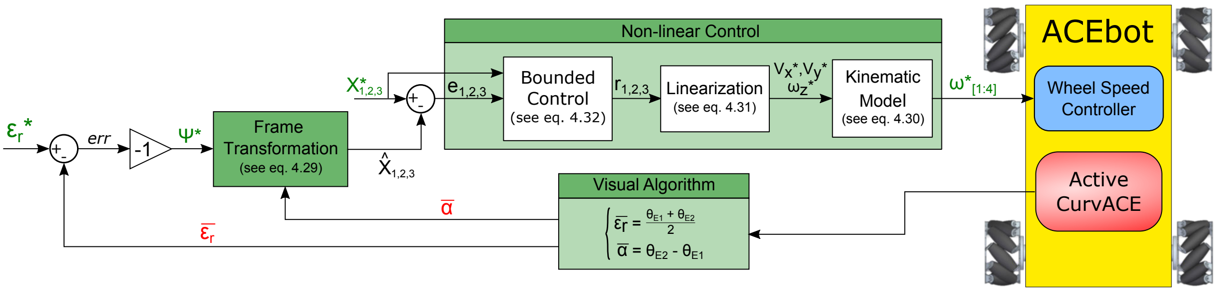


Figure 4.8: ACEbot controls its distance to the target using visual cues ($\bar{\epsilon}_r$, $\bar{\alpha}$) and the 4 wheel speeds.

Figure 4.8 shows the ACEbot's control strategy. The setpoints X^* are the desired position of the robot in the target frame. As the target is cylindrical, it has the same shape when viewed from all directions in the azimuthal plane. The robot's x direction in the local target reference frame was therefore taken to be always co-linear with the pursuer-to-Target's direction, as shown in figure 4.4b. Therefore, the x reference is the distance requirement, which has to be negative. The y reference in the target frame is null to ensure that ACEbot is aligned with the Target. The heading angle ψ was controlled in the closed-loop mode so as to keep the retinal error ϵ_r equal to zero.

The estimated state vector can be expressed as follows:

$$\hat{X}_{1,2,3} = \begin{bmatrix} x_{rob} \\ y_{rob} \\ \psi_{rob} \end{bmatrix} = \begin{bmatrix} -\cos(\psi^*) \cdot \left(\frac{R_{target}}{\sin(\bar{\alpha}/2)}\right) - D_{eye} \\ 0 \\ -\psi^* \end{bmatrix} \tag{4.29}$$

where D_{eye} is the distance from the eye to the ACEbot center. It can be noticed that $-x_{rob}$ corresponds to the distance measurement (noted D_{meas} in Table 4.2).

The kinematic model is the transfer matrix from the velocity of the robot's centre to the wheel speed:

$$\begin{bmatrix} \omega_1^* \\ \omega_2^* \\ \omega_3^* \\ \omega_4^* \end{bmatrix} = \frac{1}{R_w} \begin{bmatrix} 1 & -1 & -(l+L) \\ 1 & 1 & (l+L) \\ 1 & 1 & -(l+L) \\ 1 & -1 & (l+L) \end{bmatrix} \begin{bmatrix} v_x^* \\ v_y^* \\ \omega_z^* \end{bmatrix} \tag{4.30}$$

where v_x^*, v_y^* and ω_z^* are the robot speed control inputs. R_w, L, l are the wheel radius, the wheelbase and the track width, respectively. The linearization is the transfer matrix from the global frame, or in this case, the Target, to the ACEbot frame, which is expressed as follows:

$$\begin{aligned} v_x^* &= \cos(\psi_{rob}) \cdot r_1 + \sin(\psi_{rob}) \cdot r_2 \\ v_y^* &= -\sin(\psi_{rob}) \cdot r_1 + \cos(\psi_{rob}) \cdot r_2 \\ \omega_z^* &= r_3 \end{aligned} \tag{4.31}$$

where r_i denotes the bounded control output signals. This transformation was also useful for decoupling the control of each degree of freedom in the global frame.

Considering the notation $e_i = X_i^* - \hat{X}_i$, the r_i are the results of the following calculation:

$$r_i = \sigma_{M_{i3}} \left(\dot{X}_i^* + \sigma_{M_{i2}} \left(a_{i1} e_i + \sigma_{M_{i1}} (a_{i2} e_i + a_{i1} a_{i2} \textstyle\int e_i) \right) \right) \tag{4.32}$$

where the saturation function is defined as:

$$\sigma_M(s) = \begin{cases} s, & \text{if } \mid s \mid < M \\ sign(s) \cdot M, & \text{otherwise} \end{cases} \tag{4.33}$$

The values a_{ij} and M_{ij} are given in the Appendix 4.5.8B.

4.5.6 Experimental pursuit performances

A - Experimental setup

The Target robot was controlled either by a manual remote control system or via a feedback loop, using a VICON Motion Capture System which can give the position of the robots at a refresh rate of 500Hz with a precision of less than $1.5mm$ (see the Appendix B in [Manecy et al., 2015]). ACEbot and the robotic Target were equipped with infrared (IR) LEDs used as active markers for the VICON system to spot. The IR strobes of each VICON camera were disabled to prevent any visual perturbations from occurring due to the strongly flickering infrared lighting.

B - Ambient Light Variations

Figure 4.9a shows the trajectory of the Target in green and that of the ACEbot in yellow. The positions of both robots are shown every $8s$, and it can be seen that the ACEbot always kept close to the Target, while keeping it in its line of sight. The dark blue triangle shows the measured subtended angle.

In this experiment, the Target followed an imposed smooth trajectory with some small curves and ACEbot, after detecting the Target, was able to follow it unfailingly.

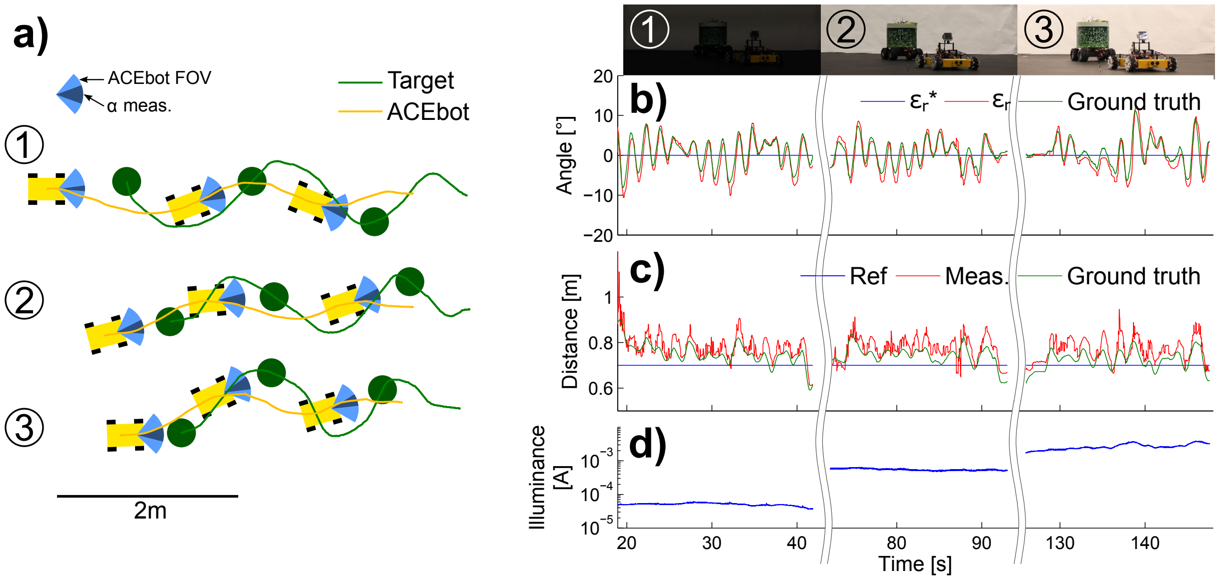


Figure 4.9: Three different following trajectories under different ambient light conditions. a) The trajectories of the center of inertia of the target and the pursuer ACEbot are presented in green and orange, respectively. The robots are drawn every 8*s*, with the full FOV of ACEbot in light blue, and the measured subtended angle of the target in dark blue. b) The orientation of the target in the FOV of ACEbot with the setpoint ϵ_r^* in blue, the measured angle ϵ_r in red and the ground truth in green. c) The distance from the target, the measured distance and the ground truth value in blue, red and green, respectively. d) Dynamic responses of the light sensor shown in fig. 4.5a, reflecting the changes in the ambient lighting, which measured 100 Lux in ①, 780 Lux in ② and 1500 Lux in ③. These responses show the robustness of the visual processing system with respect to several light levels.

As the photosensors in the artificial eye adapted fast to changes in the light level [Floreano et al., 2013], this pursuit behavior was consistently repeated under the various ambient lighting conditions tested. The tests of figure 4.9 (and also in the video in the supplementary data), were performed in a single experimental run, starting in the dark, where the target was detected and followed up to the end of the arena before returning close to the starting-point. The light was turned on when the target came to a stop. The pursuit was repeated and the blinds of the robotic arena were finally opened for the last pursuit. During this experiment, the target was remotely controlled by hand, giving similar trajectories.

It can be seen from figure 4.9 that the distance was accurately estimated and remained constant, although slightly above the reference value. This error was due to the fact that the ACEbot's speed is bounded and not large enough. The ACEbot's pursuit performances of the whole run are summarized in the table 4.2.

C - Repeatability

To test the robustness of both the visual algorithm and the robot's control strategy, the target's and pursuer's trajectories were repeated 20 times under the same lighting conditions (fig. 4.10).

It is worth noting that the pursuer took the same trajectory and never lost visual contact with the target. The standard deviation of both the target's and pursuer's trajectories was only $3cm$.

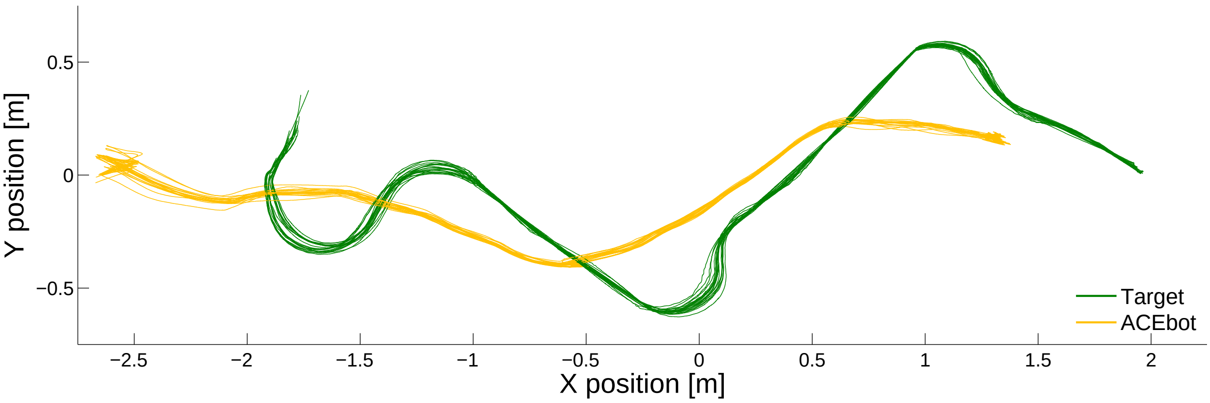


Figure 4.10: 20 recordings of target following tasks. The target's and pursuer's trajectories are plotted in green and yellow, respectively. This pursuing episode shows the excellent repeatability of the robot's ability to follow a target moving along several similar paths. The ACEbot consistently produced the same behavior, thus showing the high robustness of the visual processing algorithm.

4.5.7 Conclusion

In this paper, it was established that only 23 pixels part of a vibrating artificial compound eye was able to locate a robotic target with hyperacuity and to pursue it with great accuracy. Compared to a previous study [Manecy et al., 2016], we obtained a similar linearity in the target localization within a much larger visual range and without the need of a complex calibration process. The robot's orientation and its distance from the mobile target were precisely estimated under various lighting conditions. Table 4.2 gives the detailed results for the entire duration of the tests provided in the supplemental video. The novel computationally lean, small, lightweight bio-inspired visual sensor presented here is suitable for use in target following tasks. The limitations of the present strategy concerned the existence of a sufficiently subtended angle of the Target in the FOV (theoretically $\geqslant 2\Delta\varphi$) and the presence of a nearly uniform background or one without any sharp contrasts. We plan to test ACEbot with several target's size and target's visual characteristics as well as without any knowledge about its size. An improvement of the pursuer dynamic responses could also be explored by providing the robot with information about the target speed.

In the future, this active device could be embedded for example onboard a Micro-Aerial Vehicle (MAV). However, in the case of free flight, a second active Curvace or a 2D scanning would be required to detect the elevation of the target. In the automotive framework, a non-emissive optical sensor, like Active CurvACE, sensitive over a large angular position range and over a high range of light level could also provide a suitable alternative for the classical ultrasonic sensors and LIDARs which are being widely used these days in adaptive cruise control applications.

Table 4.2: Active CurvACE and ACEbot performance summary

Active CurvACE characteristics		
Static Optical parameters		$\Delta\varphi = \Delta\rho = 4.2°$
FOV		$96.6°$
Scanning frequency		$50Hz$
Scanning angular amplitude A		$2.1°(\approx \frac{\Delta\varphi}{2})$
Sampling frequency		$500Hz$
Visual processing bandwidth		$20Hz$
Hyperacuity resolution		$0.78° = 19\%$ of $\Delta\varphi$
Linearity		95% ($R^2 = 0.99$)
Signal to Noise Ratio		$20.7dB$
Estimated Target localization using Active CurvACE with respect to the Ground Truth (GT) (mean ± std)		
Target ang. position	$\bar{\epsilon}_r - \epsilon_{GT}$	$-0.62 \pm 1.28°$
Subtended angle	$\bar{\alpha} - \alpha_{GT}$	$-1.78 \pm 1.49°$
Target distance	$D_{meas} - D_{GT}$	$3.1 \pm 2.7cm$
Closed-loop ACEbot pursuit accuracy with respect to Ground Truth (GT) (mean ± std)		
Target ang. position	$\epsilon_r^* - \epsilon_{GT}$	$-0.67 \pm 2.63°$
Subtended angle	$\alpha_{Ref} - \alpha_{GT}$	$-2.40 \pm 5.09°$
Target distance	$D_{Ref} - D_{GT}$	$2.8 \pm 7.9cm$

4.5.8 Appendixes

A - Linear approximation and error estimation for the WS computation

The equation proposed by Heiligenberg [Heiligenberg, 1987] reads as follows:

$$WS(x,\sigma) = \sum_{k=-\infty}^{\infty} k \cdot e^{-(\frac{x-k}{\sigma})^2} \tag{4.34}$$

as $e^{-(\frac{x-k}{\sigma})^2}$ describes a Gaussian function, where σ is the standard deviation.

Let $S(x,\sigma)$ be defined as in this equation as follows:

$$S(x,\sigma) = \sum_{k=-\infty}^{\infty} e^{-(\frac{x-k}{\sigma})^2} \tag{4.35}$$

Let $x = n + y$ as in [Baldi and Heiligenberg, 1988], with $n \in \mathbb{Z}$ and $y \in [0,1]$, which means that $WS(x,\sigma) = n \cdot S(y,\sigma) + WS(y,\sigma)$

It was established in [Baldi and Heiligenberg, 1988], that:

$$\begin{cases} S(y,\sigma) = \sqrt{\pi}\sigma \left(1 + 2\sum_{k=1}^{\infty} e^{-(\sigma k\pi)^2} \cos(2\pi ky)\right) \\ WS(y,\sigma) = yS(y,\sigma) - 2\pi^{\frac{3}{2}}\sigma^3 \sum_{k=1}^{\infty} ke^{-(\sigma k\pi)^2} \sin(2\pi ky) \end{cases} \tag{4.36}$$

From the definition, $y = x - n$, which means that $S(x, \sigma) = S(y, \sigma)$ because S is periodic with a period of 1. The ratio of the weighted sum over the sum of the Gaussian functions is equal to $NWS = \frac{WS(x,\sigma)}{S(x,\sigma)}$. Given that:

$$\begin{aligned} WS(x,\sigma) = &\sqrt{\pi}\sigma \cdot x \left(1 + 2\sum_{k=1}^{\infty} e^{-\sigma^2 k^2 \pi^2} \cos(2\pi k y)\right) \\ &- 2\pi\sigma^2\sqrt{\pi}\sigma \sum_{k=1}^{\infty} k \cdot e^{-\sigma^2 k^2 \pi^2} \sin(2\pi k y) \end{aligned} \tag{4.37}$$

it can be deduced that,

$$NWS = x - 2\pi\sigma^2 \frac{\sum\limits_{k=1}^{\infty} k \cdot e^{-\sigma^2 k^2 \pi^2} \sin(2\pi k y)}{1 + 2\sum\limits_{k=1}^{\infty} e^{-\sigma^2 k^2 \pi^2} \cos(2\pi k y)} \tag{4.38}$$

The error ϵ (using $NWS = x + \epsilon$) can therefore be deduced as follows:

$$\epsilon = -2\pi\sigma^2 \frac{\sum\limits_{k=1}^{\infty} k \cdot e^{-\sigma^2 k^2 \pi^2} \sin(2\pi k y)}{1 + 2\sum\limits_{k=1}^{\infty} e^{-\sigma^2 k^2 \pi^2} \cos(2\pi k y)} \tag{4.39}$$

If ϵ is bounded, we establish here that NWS is similar to a linear function with a slope of 1.

From [Baldi and Heiligenberg, 1988], crude bounds can be computed for the series, if the inequality $\frac{\sqrt{2\ln 2}}{2\pi} \leq \sigma$ is true:

$$\begin{cases} \left|\sum\limits_{k=1}^{\infty} e^{-\sigma^2 k^2 \pi^2} \cos(2\pi k y)\right| \leq \dfrac{e^{-\sigma^2\pi^2}}{1 - e^{-\sigma^2\pi^2}} \\ \left|\sum\limits_{k=1}^{\infty} k \cdot e^{-\sigma^2 k^2 \pi^2} \sin(2\pi k y)\right| < \dfrac{e^{-\sigma^2\pi^2}}{1 - e^{-\sigma^2\pi^2}} \end{cases} \tag{4.40}$$

Thus, the absolute value of ϵ can be bounded as follows:

$$|\epsilon| \leq 2\pi\sigma^2 \frac{e^{-\sigma^2\pi^2}}{1 - 3e^{-\sigma^2\pi^2}} \tag{4.41}$$

In the present case, if we adopt the assumption that $\Delta\rho = \Delta\varphi$, $\sigma = \frac{1}{2\sqrt{\ln(2)}}$, which satisfies the inequation $\frac{\sqrt{2\ln 2}}{2\pi} \leq \sigma$, the error ϵ will be equal to 7% of the spacing between each pair of photosensors. This value results from a crude mathematical error calculation, because one can obtain an error of 6.5% numerically with only 3 Gaussian functions.

B - ACEbot control parameters

The values for each control parameter are equal to: $a_{11} = 22.5, a_{12} = 0.0225, a_{21} = 22.5, a_{22} = 0.0225, a_{32} = 45.0, a_{33} = 0$. These values make the trajectories very close to the linear region

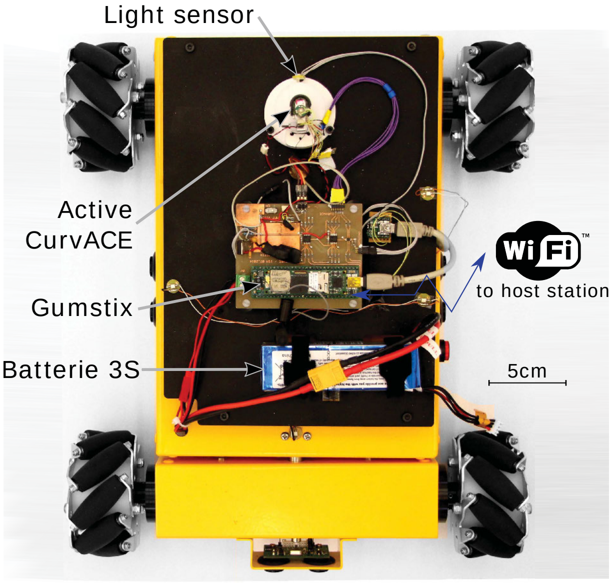


Figure 4.11: Picture from above of the robot, showing the electronic hardware. The Gumstix communicates with the ground station through Wifi as well as with the Motor control board and with the Teensy through an UART bus. Neither boards are visible on the picture as they are under the robot.

and guarantee the stability of the system. The limits M_{ij} for $i, j = 1, 2, 3$ are described in (4.42).

$$\begin{array}{lll} M_{11} = 0.54 & M_{21} = 0.675 & M_{31} = 0.9360 \\ M_{12} = 2.5515 & M_{22} = 2.835 & M_{32} = 5.256 \\ M_{13} = 7.0425 & M_{23} = 7.0425 & M_{33} = 1.5 \end{array} \tag{4.42}$$

C - Information about the Robotic platform ACEbot

The product reference of the platform is "Nexus 4WD Mecanum wheel mobile robot kit 10015" but the utlrasonic sensors are not used in the present application.

As regards the electronics, the main program was running on a *Gumstix Overo board*, which, thanks to the RT-MaG toolbox, can be programmed directly from Matlab/Simulink [Manecy, 2014]. This Linux based Computer-On-Module (COM) was running the signal processing algorithm for the 23 photosensors received from CurvACE through a synchronous serial bus (SPI). It was also connected through UART to a Teensy board. The latter was in charge of the acquisition of the photo-current sensor to record the ambient light level and to switch the stepper motor in charge of the eye's periodic tremor on and off. The wheel speed setpoints computed on the Gumstix were transmitted through a serial bus (UART) to the Arduino-compatible motor control board, which then controlled each wheel's speed with a PI controller. A WiFi connection between the robot and the ground station was used to monitor the variables of interest during the experiments and provided the rover with setpoints. The ACEbot robot was powered by a 3-cell lithium polymer battery.

Acknowledgments

We thank M. Boyron and J. Diperi for assembling ACEbot, A. Manecy for designing the target robot's control, J.F. Guerrero-Castellanos for his advice on the robot's control and E. Loret for her help with the wheel speed control.

4.6 Adaptation of the Normalized Weighted Sum with the photosensor equation

If an array of photosensors is used as input in the Weighted Sum WS and the Sum S, thanks to the equation (4.23), it can be written:

$$\begin{cases} WS = & \displaystyle\sum_{k=-N_b}^{+N_b} k \cdot Ph_{D_k}(\psi(t)) = K_3 \cdot (I_1 - I_2) \cdot A \cdot 2\pi f \sum_{k=-N_b}^{+N_b} k \cdot \exp\left(-\left(\frac{Z + k.\Delta\varphi}{d}\right)^2\right) \\ S = & \displaystyle\sum_{k=-N_b}^{+N_b} Ph_{D_k}(\psi(t)) = K_3 \cdot (I_1 - I_2) \cdot A \cdot 2\pi f \sum_{k=-N_b}^{+N_b} \exp\left(-\left(\frac{Z + k.\Delta\varphi}{d}\right)^2\right) \\ K_3 = & \dfrac{4 \ln 2}{\Delta\rho\sqrt{\pi}} \end{cases} \tag{4.43}$$

The division of WS/S allows to cancel the dependencies on the contrast, the ambient light and the active periodic vibration. However, it should be reminded that $K_3 \cdot (I_1 - I_2) \cdot A \cdot 2\pi f$ must be different from 0. $I_1 \neq I_2$ is true because a contrast should exist to be located. $A \cdot 2\pi f$ is different from 0 if the frequency and the amplitude are not null. The role of the envelop detector could be underlined here. Indeed without it, the condition $\frac{d\psi}{dt} \neq 0$ could not be met in the case of a sinusoidal scanning only. That is the reason why we are using a such demodulation processing, which integrates the high-pass filter for the derivative and the envelop detector which highlights the highest amplitude of the signal which is therefore not 0 in presence of a contrast.

It was proved in the section 4.5.8 that the NWS is equal to $x + \epsilon$ in the case of Gaussian functions described as follows: $e^{-(\frac{x-k}{\sigma})^2}$. As the demodulated photosensor signals are expressed as in (4.43), the mathematical equivalence are $X_{eq} = \frac{Z}{\Delta\varphi}$ where $Z = \psi - \psi_e$ and $\sigma = \frac{d}{\Delta\varphi}$.

Therefore, in the case of the visual sensor we are using, it can be established that:

$$\frac{WS}{S} \approx \frac{\psi - \psi_e}{\Delta\varphi} \tag{4.44}$$

Moreover, to calculate the error from the equation (4.41), the equivalence for σ is:

$$\sigma = \frac{d}{\Delta\varphi} = \frac{\Delta\rho}{2\sqrt{\ln(2)}\Delta\varphi} \tag{4.45}$$

In the present case, if we adopt the assumption that $\Delta\rho = \Delta\varphi$, $\sigma = \frac{1}{2\sqrt{\ln(2)}}$, which satisfies the inequation $\frac{\sqrt{2 \ln 2}}{2\pi} \leq \sigma$, the error ϵ will be equal to 7% of the spacing between each pair of photosensors.

Setting the scale and offset As one can see from the equation (4.44) is that there is no scaling of the output. It means that the output is proportional to the angle. Indeed, it is more a value fitting to the photosensor number of the array. Therefore, any scaling can be realized by multiplying the ratio WS/S by the scale required. Here, the output can be directly the angle by multiplying be the spacing between 2 photosensors $\Delta\varphi$.

An offset is also required to set the output value relative to its environment. This offset correspond in finding the ψ_e position relative to the photosensor array. It is the main advantage of this algorithm, it only requires 2 values as input; one from the layout of the photosensor array, usually known and another from a calibration.

But better results could be expected if a calibration is done for each $\Delta\varphi$. It would surely improve the linearity but also the complexity. It is a trade-off between complexity of the calibration process and the expected resolution. A such calibration would not really be a significant improvement if the ASF are not symmetrical or well shaped. But it would be particularly well suited to ASF fitting very well with the Gaussian-shaped functions.

4.7 Supplementary results and information about the robotic experiment

4.7.1 On the calibration process

It is not referenced, but an amplitude calibration is made in order to have a similar amplitude of each photosensor for a given contrast. This was done once and for all, before any experiments. The amplitude calibration is here to compensate for the difference between the photosensors. These difference are coming from the different Angular sensitivity and the electrical circuit responses.

4.7.2 About the control strategy

A - Distance control

This experiment consists in following the target at a distance of $70cm$. The Target generates short and fast movements, forward and backward. Figure 4.12a shows the distance measurement compared to the setpoint and the ground truth and Figure 4.12b shows the Target and ACEbot speed during the experiment. One can see that although the distance is well estimated, the maximal speed of ACEbot is not fast enough compared to the Target to decrease significantly the error. During this test, the mean error is only $3.86cm$ with a standard deviation of $2.1cm$.

Moreover the reaction time could be faster if the speed of the target was estimated and used in the control law (see appendix D.4). The inertia of the robot plays a role in this response too, but not as much as the controller, because the acceleration is quite close to the target one. Here, the lag between target and pursuer movement could be decreased.

The mistake comes from the controller in itself which is based on a position controller. It is similar to assumed that the target is not moving, leading to a steady state error. A controller with a target speed estimation would provide a better tracking, improving the pursuer response. For example, other studies [Mariottini et al., 2009, Das et al., 2001, Das et al., 2002, Cowan et al., 2003] used an EKF (Extended Kalman Filter) to improve the dynamic response in a

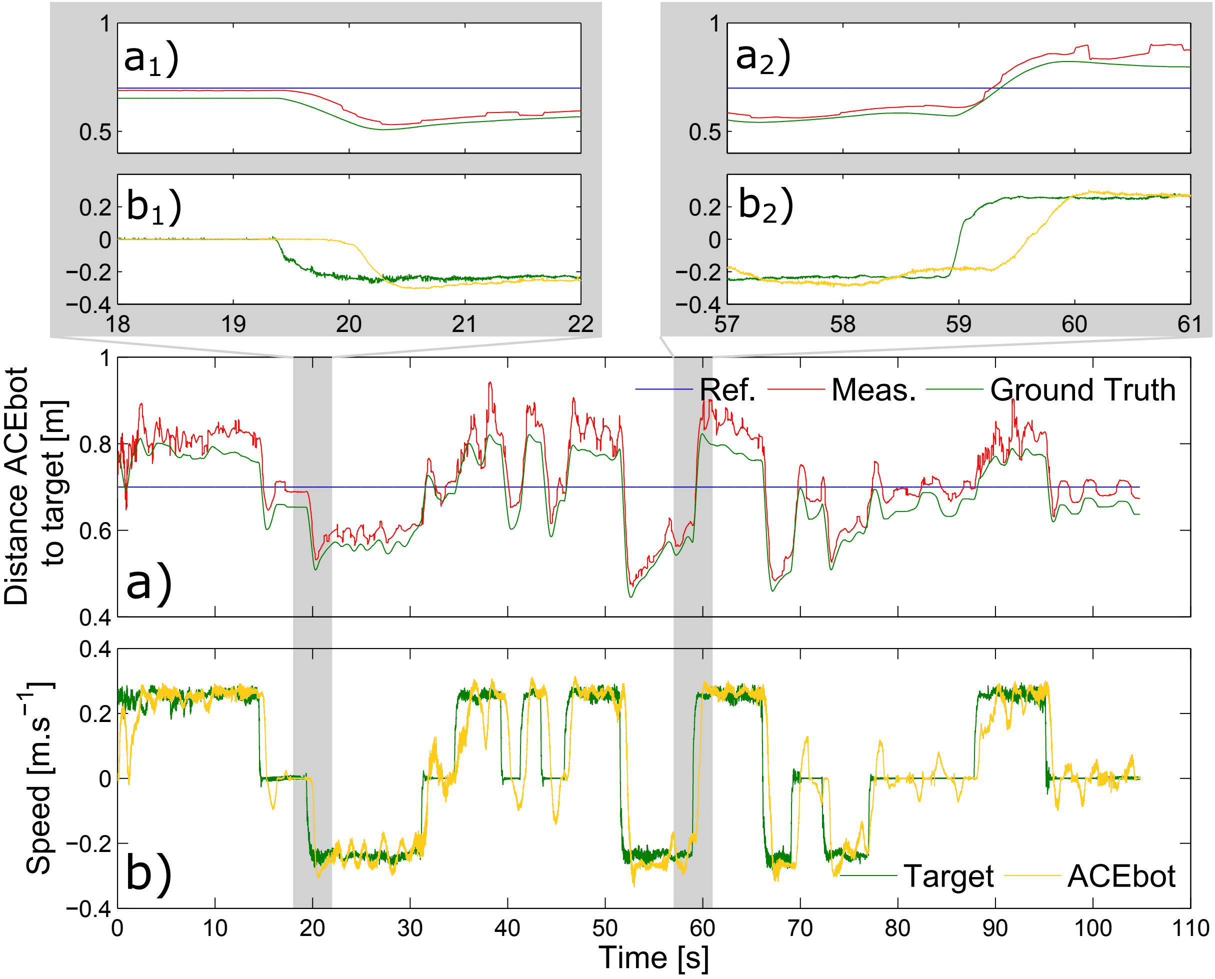


Figure 4.12: Simple target trajectory which is moving back and forth in front of the robot and marks some stops. a) The distance between the robot and the target, in blue the reference, in red the measurement and in green the groundtruth calculated from the motion capture system. b) The speed of the target and ACEbot in green and yellow respectively. a$_1$-b$_1$-a$_2$-b$_2$) are zoom of the grey part of subfigures a and b from 18 to 22s and 57 to 61s, for the first and second ones respectively.

leader-follower scenario without communication. This can be reproduce for pursuit behavior. See appendix D.4 to have a better understanding of the problematic and its possible solution.

B - Designed Limitations

It should be noticed that in the controller, there is a saturation that cancel any translation in the case that the orientation error is above ±15°. This saturation is useful to prevent from moving forward if the target is close to the limit of the FOV of the robot. Indeed, in this particular situation, the robot could go forward before compensating of the orientation which could lead the target to go outside the FOV. Moreover, in this situation the approximation of small angle can be assumed. If $\bar{\epsilon}_r$ is small, $\sin(\bar{\epsilon}_r) = \bar{\epsilon}_r$. The same equality can be done for the angle β, the angle between the direction of the ACEbot and the line between the center of inertia of both robots, ACEbot and the target (see figure 4.13)

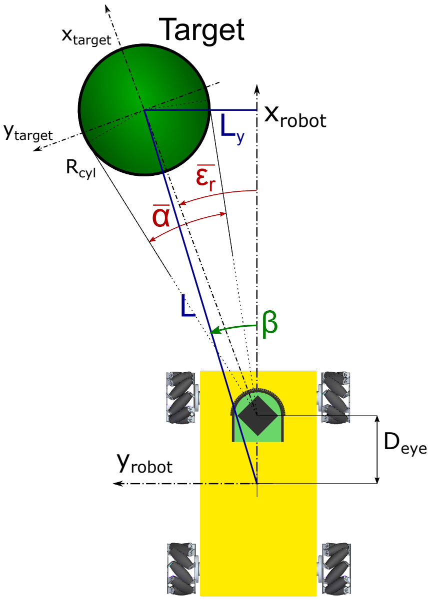


Figure 4.13: Modification of figure 4.4b. It shows that the angle β, the angle between the orientation of ACEbot and the line joining the center of the robot and the center of the target is not equal to $\bar{\epsilon}_r$, the retinal error.

Figure 4.13 shows indeed that $\bar{\epsilon}_r \neq \beta$ which led to a mistake in the expression of the state vector $\hat{X}_{1,2,3}$. It is not very relevant in our case because the approximation of small angle hold but if the FOV is increased, it could become a problem in the state estimation.

From the law of cosines (in France also called the theorem of Al-Kashi), the following expression can be written:

$$\left\{ \begin{array}{rcl} L^2 & = & {D_{eye}}^2 + {D_{meas}}^2 - 2D_{eye} \cdot D_{meas} \cdot \cos(\pi - \bar{\epsilon}_r) \\ {D_{meas}}^2 & = & L^2 + {D_{eye}}^2 - 2L \cdot D_{eye} \cdot \cos(\beta) \end{array} \right. \tag{4.46}$$

where D_{meas} is the distance estimated thanks to the subtended angle measurement, *i.e.* $D_{meas} = \frac{R_{target}}{\sin(\bar{\alpha}/2)}$. D_{eye} is the distance between the center of rotation of the robot and the position of the visual sensor.

The right values for L and β can be deduced:

$$\left\{ \begin{array}{rcl} L & = & \sqrt{{D_{eye}}^2 + {D_{meas}}^2 + 2D_{eye} \cdot D_{meas} \cdot \cos(\bar{\epsilon}_r)} \\ \beta & = & \arccos\left(\frac{L^2 + {D_{eye}}^2 - {D_{meas}}^2}{2L \cdot D_{eye}}\right) \end{array} \right. \tag{4.47}$$

Then, the correct state vector is expressed as follows:

$$\hat{X}_{1,2,3} = \begin{bmatrix} x_{rob} \\ y_{rob} \\ \psi_{rob} \end{bmatrix} = \begin{bmatrix} -\sqrt{{D_{eye}}^2 + {D_{meas}}^2 + 2D_{eye} \cdot D_{meas} \cdot \cos(\bar{\epsilon}_r)} \\ 0 \\ -\beta \end{bmatrix} \tag{4.48}$$

With this state estimation, the feedback loop on the retinal error used to compute ψ^* is redundant with the one on the robot control position. If the eye is rotated, the value of $\bar{\epsilon}_r$ should add this offset.

No experiments were made with this new controller, the impact on the robot dynamics is still unknown. I think it can improve the robot behavior a bit, but it is likely that it would not be significant.

C - Choice about the heading direction

The control could have been to use the frame transformation in the way the misalignment would have been compensated by a sway movement. It was not the first choice because it is not really effective when the target is far away and if the forward speed is higher than the lateral one, the target could be lost. However, sway movement could be interesting in the case of swarm. The robot could assess its relative position according to a leader without modifying its heading. In this case, the distance would not change very much or at least being bounded by the dynamics of the leader. The possibility of loosing it is very reduced. The estimated state vector, which is the Frame Transformation vector in figure 4.8 would be expressed as follows:

$$\hat{X}_{1,2,3} = \begin{bmatrix} x_{rob} \\ y_{rob} \\ \psi_{rob} \end{bmatrix} = \begin{bmatrix} -\cos(\psi^*) \cdot \left(\frac{R_{target}}{\sin(\bar{\alpha}/2)}\right) - D_{eye} \\ -\sin(\psi^*) \cdot \left(\frac{R_{target}}{\sin(\bar{\alpha}/2)}\right) \\ 0 \end{bmatrix} \tag{4.49}$$

where the notation are the same as in (4.29).

This kind of control would assume to have a constant heading, but a slow drift could happen if no other sensor is used. A compass could be used to compensate for this drift onboard. A more complex compensation strategy with visual feedback could be at work as well. Another easier solution could have been to used external sensor as the motion capture system to fix the heading with the appropriate feedback loop.

4.8 Discussion on the localization measurements and their use in a control law

4.8.1 About the known target hypothesis

The control law was designed around the hypothesis that the object to track was known. Indeed, it is assumed that the object is a cylinder with a known diameter. The assumption about the cylinder diameter is needed to convert directly the measurement of the subtended angle into the distance to the object. This assumption can be valid in a biological context, as suggested by Collett and Land [Collett and Land, 1978] that the size of the female could be known by the male in the female interception task. Although this hypothesis has not really be confirmed yet.

A such control is interesting because it helps to have a distance measurement at each time without requiring the computation of its egomotion [Jung and Sukhatme, 2004] nor binocular vision [Kwon et al., 2005]. It requires to have the target in its FOV and not losing it.

Another point to underlined is that, with the transformation, the distance estimation error is decreasing as the target get closer with the same resolution of the angular position of the edge 4.14. This feature is interesting as it ensures to converge toward the target and to catch it if its a prey or a potential mate.

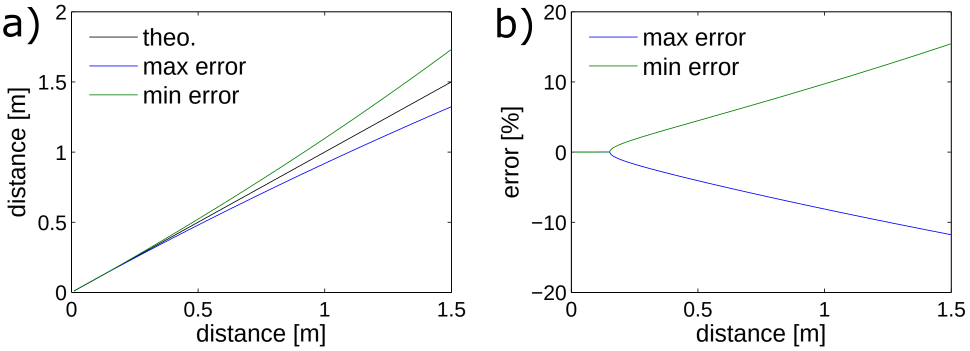


Figure 4.14: Comparison of the estimation error according to the real value. a) The maximal error is calculated with an error of +0.78° and −0.78° for the left and right edges respectively relative to the theoretical positions. +0.78° and −0.78° for the left and right edges respectively in were used to calculate the minimal error. The minimal error leads to overestimate the distance and the maximal error to underestimate the distance. It can be seen that closer the target, reduced is the error estimation. b) The error of the estimation relative to the distance.

4.8.2 A Controller with a linearization of the subtended angle measurement

But another way of doing the control of the robot is to assessed the subtended angle directly. Therefore, keeping a distance constant would also be possible but with a distance depending on the target size. A simple linearization at a certain point can be imagined as in figure 4.15 to transform the subtended angle measurement into a distance value.

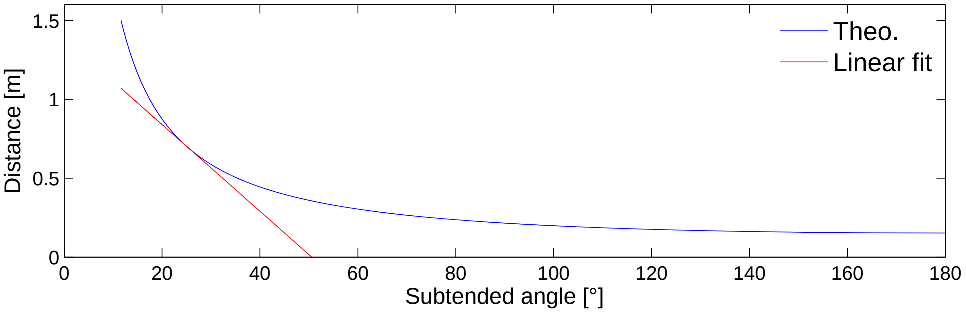


Figure 4.15: Linearization of the distance function according to the subtended angle.The blue curve is the theoretical function for a given target size of a radius equal to $0.152m$. The red one is the linearization at a distance of $0.7m$. The blue curve is the theoretical function for a given target size of a radius equal to $0.152m$. The green and the red ones are respectively the distance outputs if there is an error on the subtended angle of +10% and −10% respectively.

This solution is very simple and can work only if the target is close to the setpoint. However, far from this position, it would not provide an accurate response especially for large subtended angles. This simple solution could only be used in a leader-follower scenario or a swarm, as it is needed to start close to the reference point. In such scenario, the robot to follow is completely known, therefore the solution using the size of the target would be preferred.

4.8.3 A Controller with the subtended angle measurement directly

One can choose to do a control directly on the subtended angle. It present the advantage to not require the size of the target, at least in the control loop itself. But maybe, the knowledge of such data or at least a bounded value could be helpful in designing a controller.

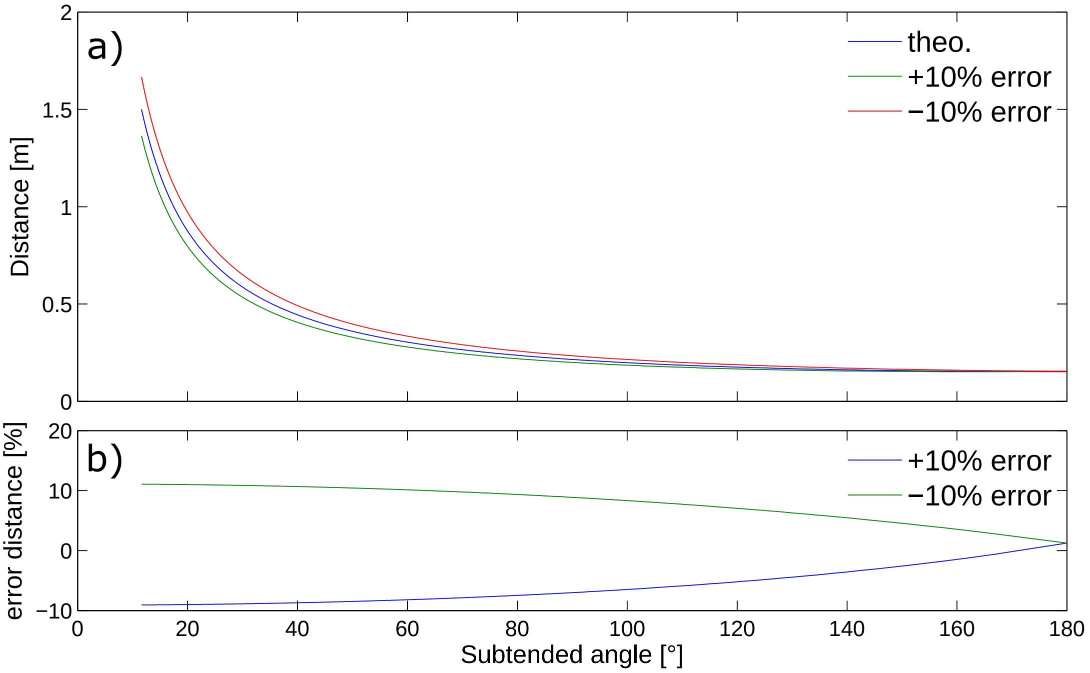


Figure 4.16: a) Distance according to the subtended angle. The blue curve is the theoretical function for a given target size of a radius equal to $0.152m$. The green and the red ones are respectively the distance outputs if there is an error on the subtended angle of $+10\%$ and -10% respectively. b) Percentage of the distance error for both conditions

It is shown in figure 4.16 that an error of $\pm 10\%$ in the subtended angle value will lead to a variation of distance higher for smaller subtended angle than bigger one. It will lead to a larger position error according to the target at larger distance compared to smaller one. It is a similar problem to the localization one expressed in figure 4.14. For target following, it could be annoying because the position error would depend on the setpoint chosen.

4.8.4 A Controller with the inverse of the subtended angle

Figure 4.17a shows the Inverse of the Subtended Angle (ISA) according to the subtended angle itself, which is obviously an hyperbolic function. The interesting property of the ISA is that the distance can be expressed as a linear function of it. The coefficient of determination R^2 is equal to 0.999 for this linear regression estimation in the example of figure 4.17b. Here, the example is made with a target with a radius of $0.152m$ as used in the experiment. It can be seen that for small ISA value, under $0.01deg^{-1}$ the error increase. It means that the error is always under 5% for $ISA = 0.0097deg^{-1}$, which correspond at a subtended angle larger than 103°, or a target of radius $0.152m$ seen at a distance closer to $0.194m$.

As depicted in figure 4.18, the size of the target has an impact on the slope of the ISA

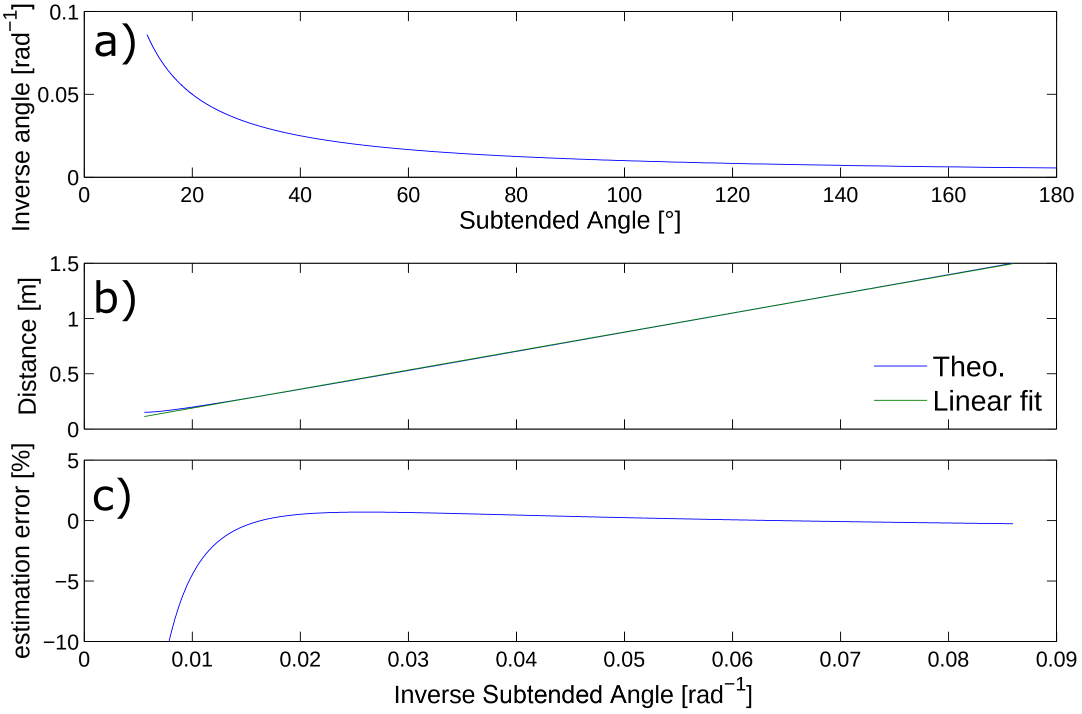


Figure 4.17: a) Inverse of the Subtended Angle (ISA) according to the subtended angle. b) Distance of a cylindrical target of a radius equal to $0.152m$ according to the ISA in blue compared to a linear fit. c) Error of the linear fit relative to the exact inverse function.

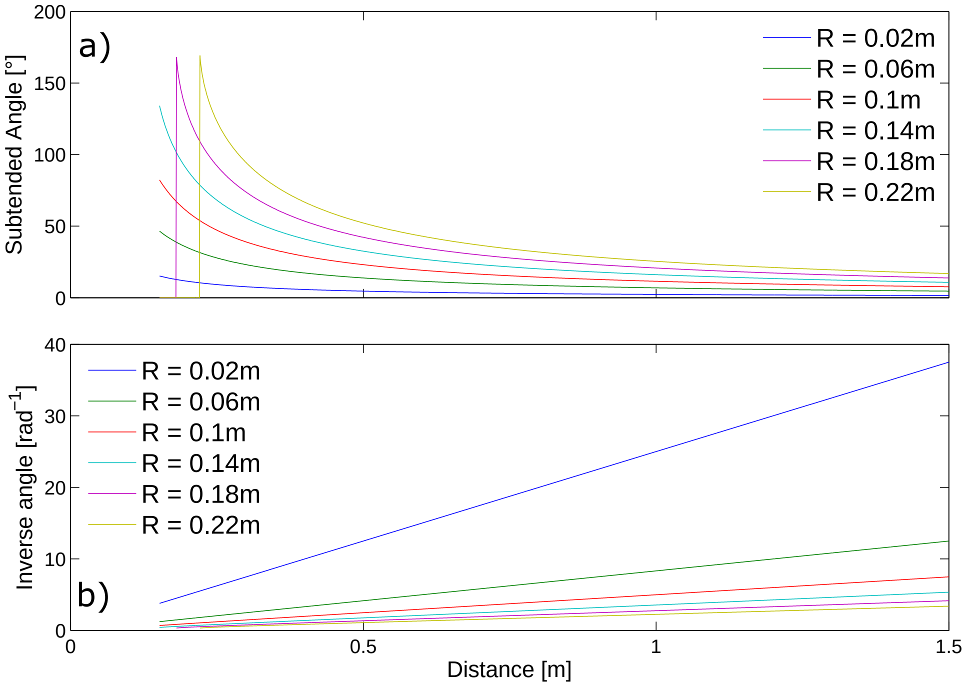


Figure 4.18: a) Variation of the subtended angle according to the distance for six different target sizes with a radius of $0.02m$, $0.06m$, $0.1m$, $0.14m$, $0.18m$ and $0.22m$ respectively. b) The Inverse Subtended Angle versus the distance for the same targets. These 2 graphs highlight the difference of the response of the ISA for different target size.

according to the distance. Therefore, in the context of an unknown target, it would be difficult to have a controller that adapt to all of these variations.

However, if the gain could be estimated, the size of the target would be known. In a situation where the target is immobile or considered slow compared to the dynamics of the pursuer, and the egomotion well estimated to have the variation of the relative position (*i.e.* the distance), the size of the target would be identified.

Chapter 5

Bio-inspired behaviors: Target following, Target detection and interception

Contents

5.1 Summary

This chapter depicts the similarity of the ACEbot pursuit behavior with the hoverfly trajectories when the male follows the female. The control strategy is discussed with the cues used to perform such behavior. It seems that the male hoverfly does not estimate the target speed of the female during pursuit.

An experiment with a decoupled eye from the ACEbot frame was made with a saccadic tracking. It shows good results but no real difference was observed with a fixed eye. We argue that the uniform resolution over the entire FOV balanced the need of an additional degree of freedom for the eye.

Target detection is also investigated. The need to initialize the visual processing for the pursuit task makes us think about bio-inspired principle of target detection. It is still unclear what the information used by the insect is to ensure to pursue a potential prey or mate.

Finally, it is established that few modifications in the pursuit controller of ACEbot are needed to perform interception. It is even shown that more elaborate trajectories are possible only by orienting the eye at a different location. It is known that dragonflies use even more complex strategies in order to anticipate the displacement of their prey.

5.2 Bio-inspired following behavior

5.2.1 Comparison between robotic and hoverflies experiments

The strategy developed in this thesis is based on the measurements of the angular position of 2 edges of the target, in order to elicit the subtended angle and the orientation of the target in the pursuer FOV. The measured subtended angle can be converted into a distance measurement on the assumption of a known target size. These estimated parameters, ψ and d (the orientation and the distance respectively), are controlled so as to keep ψ and d equal to $0°$ and $0.9m$, respectively.

[Collett and Land, 1975] observed some trajectories of the hoverfly *Syritta Pipens L.* that achieved a following behavior with a precision of $1cm$ at a distance of $10cm$. As can be seen in figure 5.1, similar trajectories between robotic and insect observations are achieved. Therefore, the strategy that consists of keeping the target in the middle of the FOV seems to be compatible with the biological observations. It should be highlighted that the insects are using visual cues to maintain the distance from the mate. The control is not identified as the difference between distance and subtended angle is hard to distinguish (see chapter 4).

It is also interesting to note that in figure 5.1e the distance profiles are very similar between the insect and the robotic experiments. These variations in the distance to target are likely to be caused by the position controller, which is slow when the direction of the target changes, as seen in chapter 4 and could be fully compensate for target movement with a target speed knowledge (see appendix D.4). For the insects, the reason could be identical. It would be a clue for no target speed estimation, but further investigation should be made in order to clearly conclude on this theory.

One could say these variations could also be due to the variation of the subtended angle of the female in the male FOV. Indeed, in the beginning of the pursuit, the male sees the face of the female, in the middle the female presents its side, then its back, to finally finish with its side. Figure 5.2 shows no direct correlation in the evolution of the 2 measurements. Even by comparing the invert of the subtended angle which has been proven to be quasi proportional to the distance in figure 4.17. It is very difficult here to draw a conclusion as the measurements are noisy. Another similar experiment should be realized with more precision to really form a

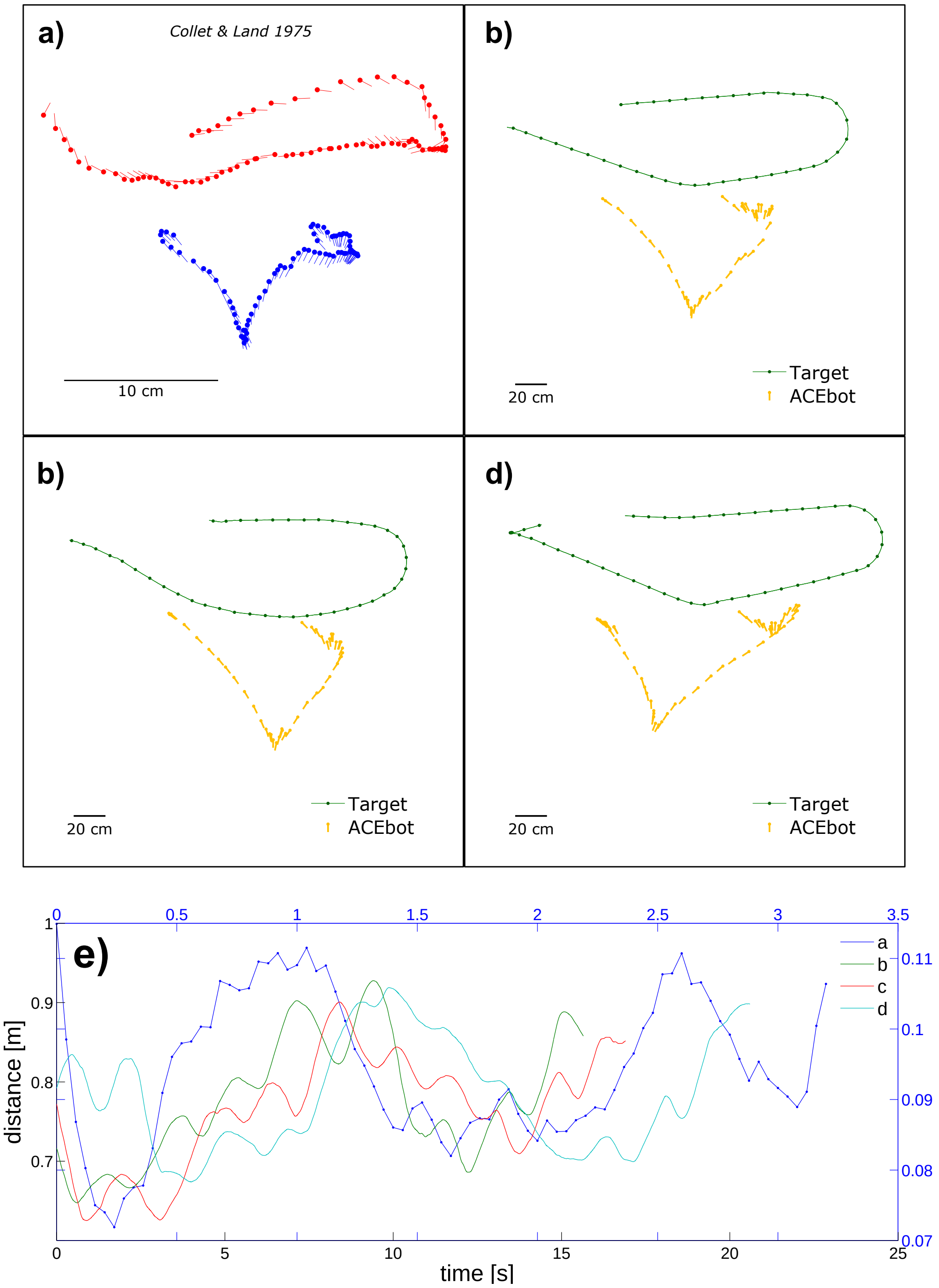


Figure 5.1: Comparison between the following trajectory observed during *Syritta pipens L.* mating process by [Collett and Land, 1975] (a) and the target following trajectories of ACEbot (b, c and d). In a), the male and the female are represented in blue and red respectively. Dots are representing the head position and the lines are the orientation of the body. In b-d), the target is represented by a green dot and a continuous line for the whole trajectory. ACEbot, the pursuer, is represented in yellow; the dot is the eye position and the line represents the robot orientation. e) The distance between the 2 subjects in every experiment is plotted versus time. It turns out that the distance to target follows a similar pattern for both the hoverflies and the robots, at a different scale of course. The markers are separated by $40ms$ in time in subfigure a and $400ms$ in the others.

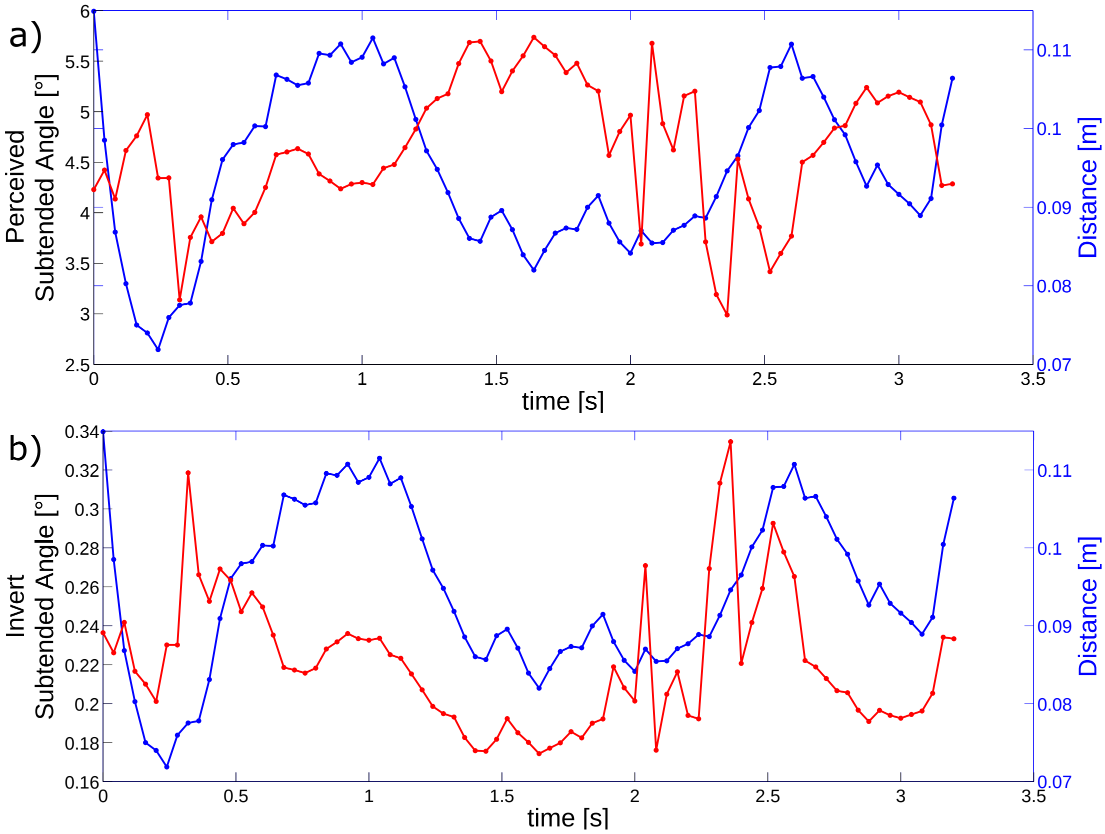


Figure 5.2: Evolution of the distance, in blue, compared in a), to the subtended angle of the female perceived by the male Hoverfly (in red) and in b), to the invert of the subtended angle of the female perceived by the male (also in red). To calculate the perceived subtended angle, the female is considered as a rectangle of 9 by 3mm and the largest subtended angle is taken as measurement.

conclusion.

Moreover, the pursuit strategy of the *Syritta pipiens L.* is not completely understood, as it can follow and maintain distance to its potential mate before trying to catch it in two manners depending on the position of the mate in its field of view. It uses mostly saccades and sway movements when the target is on the side but does continuous tracking when it is upfront. Sometimes, it was also observed that hoverfly does sideways movement even if the mate is in front of it, but these movements were unpredictable [Collett and Land, 1975].

5.2.2 Discussion on optics and visual cues with respect to biological counterparts

The good precision achieved by the male in the mate following is quite impressive regarding the nature of its eye. [Collett and Land, 1975] explored different hypothesis for the cues used by the hoverfly *Syritta Pipens L.* to display a following behavior with such precision. Focus, disparity, size determination or motion parallax are the four possible cues studied. The focus was excluded immediately because in the compound eye, it is a fixed value and a variation between 5mm to infinite would be difficult to sense. Disparity is the difference between left and right eyes. It is similar to the binocular vision which greatly depends on the baseline of the 2 eyes ($250\mu m$ for the *Syritta Pipens L.*). As a consequence, this seems unlikely because of the precision of the eye that it would require.

The robotic experiment here underlined the fact that the distance from the target should be measured during the following task. However, the principle used by the hoverfly might be different. The subtended angles they are dealing with are smaller compared to their optics. As far as I know, no study has tackled the problem of distance estimation with low spatial resolution ($\Delta\varphi > 2°$) and monocular vision in the case of target tracking. The ESTMD ('Elementary' Small Target Motion Detector) algorithm uses an optical blur with a Gaussian function of full-width at half maximum of 1.4°, but does not reduce the spatial resolution of the camera [Bagheri et al., 2015]. In robotics, the egomotion is used to create a depth map in SLAM algorithm with monocular camera. But, the robot moves in a steady environment and this could not be transferred to distance estimation in target pursuit context. However, with cameras, it is possible to identify a target and use known metrics of it to estimate distance [Dagan et al., 2004]. With the very coarse resolution of the insect eye, it is difficult to imagine such identification. But maybe it is what flies do with some color or shape cues.

5.3 Steering by Gazing: a Saccadic Eye Controller to follow the target

The same experiments of target tracking were reproduced with a rotation of the eye. This control architecture was previously suggested in [Kerhuel et al., 2010] and was inspired by the fly head and body decoupling. The main idea was to have an eye decoupled from the body with a rotational degree of freedom. In their experiments, it was shown that with this added rotation, the tracking response was quicker because the eye movement dynamics were faster than the robot's body dynamic.

In the present PhD project, a closed-loop control at high speed was not achievable. The maximal speed of the eye (induced by the servomotor MKS DS92a+) was about $860°/s$, which was a very interesting feature. But due to the adaptive pixel of the CurvACE sensor, the dynamics of the measured position were too slow after signal processing. As it will be seen in figure 5.4, the photosensor's response need time to adapt. The solution proposed here to overcome this drawback was to realize a saccadic control as done the praying mantis on a textured background [Rossel, 1980,Kirschfeld, 1994]. A fine tuning of a smooth controller could have been done but the dynamics would have been close to the robot's own.

5.3.1 Description of the Saccadic Eye Controller (SEC)

In this section, the eye's controller uses saccades to make it able to follow the target with discontinuous movements. The control scheme is presented in figure 5.3.

First, the comparison between the retinal error setpoint ϵ_r^* and the measurement $\bar{\epsilon}_r$ gives an error angle ϵ in the eye frame. Then, if the absolute value of ϵ is above $\Delta\varphi$, a saccade is triggered with an amplitude equal to the visual error ϵ. However, the saccade trigger can be inhibited if the duration since the last saccade $\Delta t_{saccade}$ is not long enough, here the minimum intersaccadic duration is $T_{min} = 0.5s$. Finally, the saccade generator gives a difference of position between 2 sample time equal to $-\epsilon$ in the case of saccade or 0 otherwise. As ϵ is expressed in the eye's frame, the position at $(t-1)$ should be added to the new correction in order to give the new ψ_{er} value. This computation acts as an integrator.

In the eye controller, a Vestibulo Occular Reflex (VOR) was also added to compensate for the rotation of the robot in order to keep the target at the same position in the FOV during the robot rotation. As the eye is moving faster than the robot thanks to the fast servomotor, it reaches the target line of sight at each saccade before the end of the change in the robot's heading. Then, the robot movement rotates at a slower speed to get in the same direction. During this movement, the gaze direction stays constant with respect to the environment thanks to the VOR as shown in section 5.3.3. The input of the VOR comes from a rate gyro (an Analog Devices ADXRS300), which is filtered, and provides a measurement of the rotational speed of the robot around its vertical axis. The opposite of the rotational speed is added to the saccade signal before being integrated to provide the angular position of the eye. The choice to use a rate gyro in the loop compared to the wheel speed measurement which could then give a rotational speed, was done to avoid false measurements due to wheel sliding and to demonstrate that such an algorithm can also be used on a flying robot as demonstrated in [Kerhuel et al., 2010].

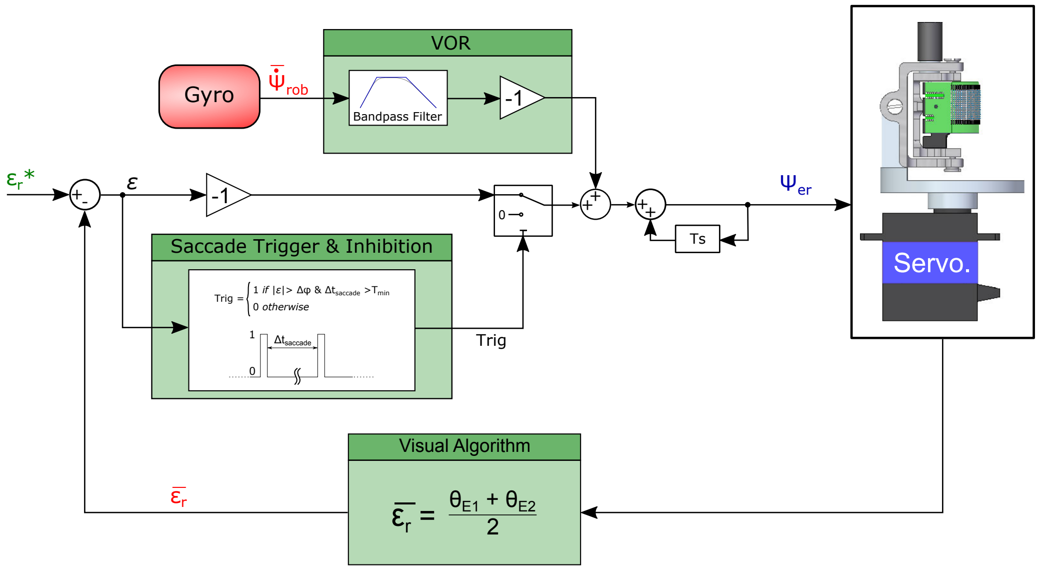


Figure 5.3: Description of the saccadic eye control system. The Bandpass filter of the rotational speed is composed of a high pass filter and a low pass filter, both order 1, with cutting frequencies of $0.05Hz$ and $6Hz$, respectively. This control loop is made at a sampling time Ts equal to $2ms$. It is highlighted that between two saccades, the retinal error measured has no influence on the position of the eye.

5.3.2 Vestibulo Fixational Reflex: Eye Tracking experiment

Firstly, to demonstrate the performance of such saccadic control, a target was placed in front of the Active CurvACE sensor, describing an eight-shaped trajectory as presented in figure 5.4a. Figures 5.4b and 5.4c show the angle measurements of the position of the target and its size in the eye FOV.

As can be seen in figure 5.4, the estimated target orientation has some overshoots at each

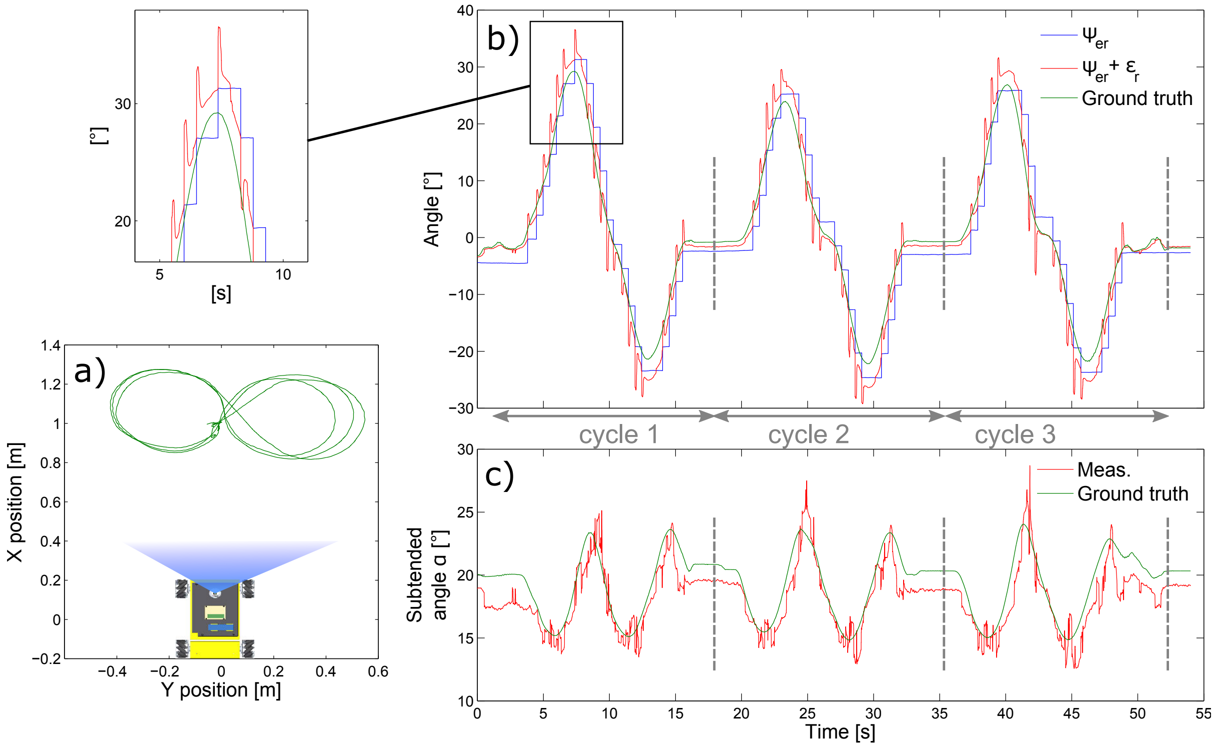


Figure 5.4: a) Trajectory of the target in front of the sensor in the XY plan with the ACEbot position and in blue the field of view of the visual sensor
b) Tracking response of the sensor with the eye position in blue, the estimated value which is the retinal error added to the eye position and in green the theoretical value.
c) Comparison between the measured subtended angle of the target and the ground truth calculated from the Motion capture system during the same experiment.

saccade but converges to the theoretical value. During the saccade, the target is considered not visible. Thus, the inhibition is required, similar to what may occur in animal saccadic tracking. For example in flies, they are using a mechanism to suppress the eye input during a body saccade [Heisenberg and Wolf, 1979]. As a result, the eye orientation tracks the target accurately despite the discontinuities.

The minimal time between two saccades (or the time of inhibition) T_{min} is equal to $0.5s$. This value is chosen according to the convergence dynamics of the visual processing. It could also be noticed that different strategies could have been chosen to saccades. For example, a quantization could have been chosen. But based on the observation on the praying mantis tracking a prey on a textured background [Kirschfeld, 1994], the saccades' amplitudes are not constant. Therefore, the solution of canceling the error at each saccade was preferred. It seems to be the best solution in order to keep the line of sight the closest to the target position during the inter-saccadic period, without making any assumptions on the future target position.

5.3.3 Performance of the Vestibulo Occular Reflex on ACEbot

A Vestibulo Occular Reflex was incorporated into the control law to make the line of sight lock onto the target regardless of the robot rotation. In the following experiment, no target was in front of ACEbot and the Fixational reflex was deactivated. The robot started with a simple rotation due to a step input reference. Figure 5.5a shows that the robot estimated the rotation and compensated for it. It can be seen that the gaze *i.e.* is converging to the robot orientation very slowly due to the effect of the High-Pass Filter, as the VFR was deactivated. Figure 5.5b shows the VOR response when the robot is making sinusoidal rotation. It can be seen that the gaze is always heading in the same direction with less than 5° of variation, despite the 15° of amplitude of the robot rotation. The performance could have been improved if a feedforward was included in the eye controller, using that the rotational setpoint as an input. But a good identification process would have been necessary in order to predict the robot dynamics. However, in the case of a flying robot, a feedforward would help in case of volontary movement but not in disturbance rejections. Therefore, a rate gyro would still be necessary.

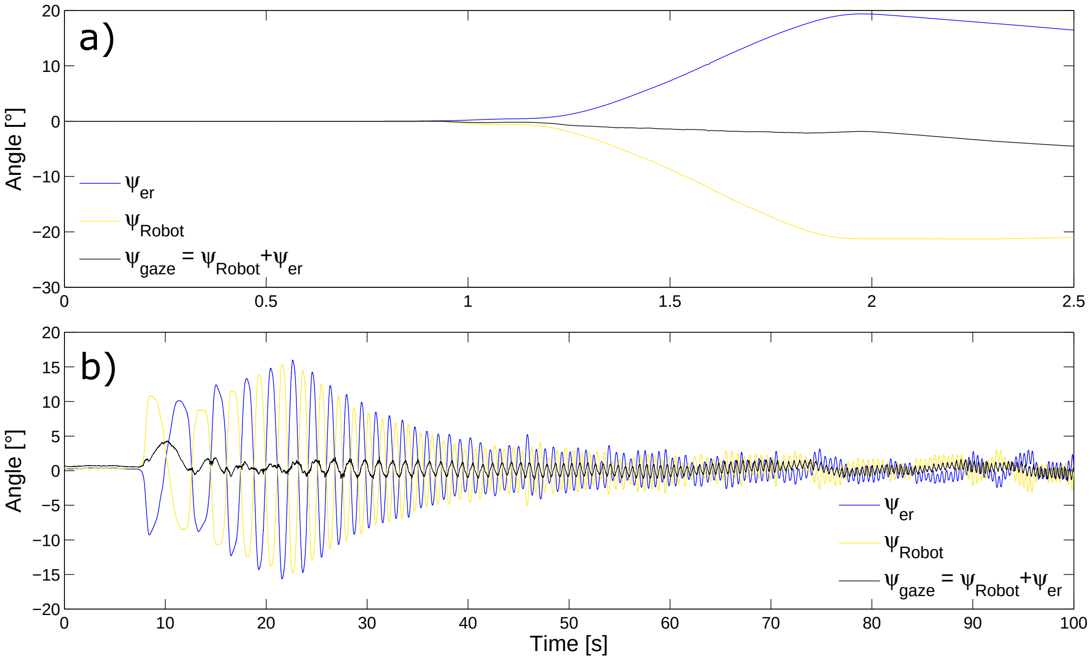


Figure 5.5: a) Measured response of the Eye angle to a robot rotation for a step input. b) Measured response of the Eye angle to a robot rotation for a chirp reference, from $0.1Hz$ at $7.5s$ to $1.2Hz$ at $100s$.
In both cases, the blue, the yellow and the black curves are the Eye-in-Robot angle, the robot angle in absolute frame and the gaze direction also in the absolute frame, respectively. In this situation, no target was in front of the eye and the VFR was deactivated. It shows that the rotation of the eye compensates for the rotation of the robot

Another experiment has been conducted to see the performance of the VOR in coordination with the VFR. The robot ACEbot was placed in front of the target and a step was applied on the retinal error setpoint, which went from 0 to 6. Figure 5.6 shows the results of the experiment. In the light grey part, it can be seen that the retinal error was constant and under the threshold

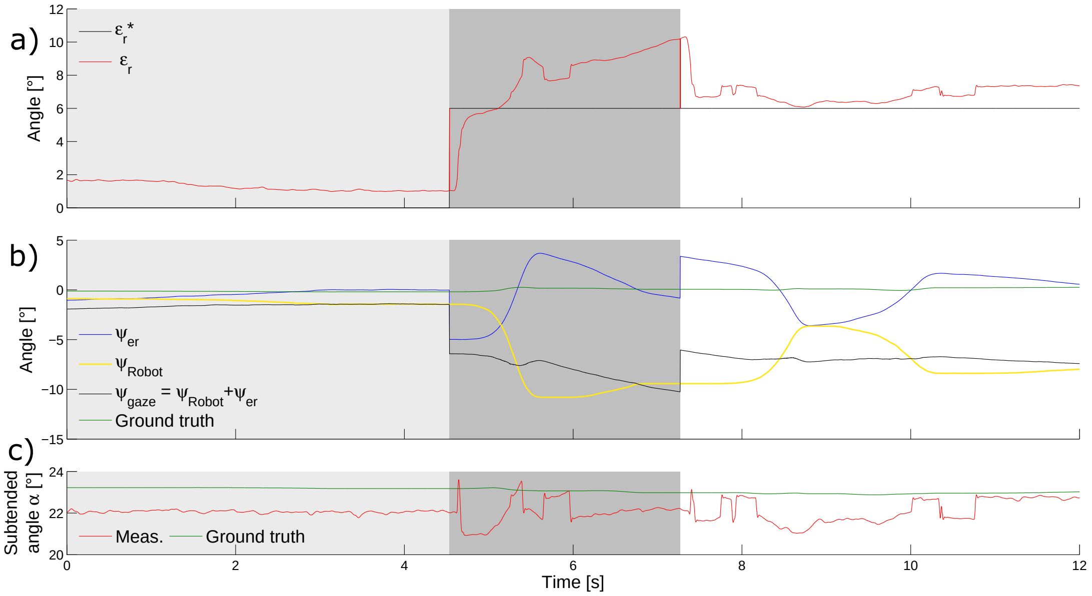


Figure 5.6: The target is placed in front of the robot and a step of 6° is made on the retinal error setpoint at $4.5s$, which lead to a saccade of the eye, hence a rotation of the robot. a) Measured retinal error of the target position in the eye FOV in red and the reference in plotted in black.
b) In blue, the Eye-in-Robot angle. In yellow, the orientation of the robot. In black, the gaze direction (*i.e.* the orientation of the eye) expressed in the absolute frame and in green, the orientation of the target according to the robot also expressed in the absolute frame.
c) Measured subtended angle in red compared to the ground truth.

$\Delta\varphi$. Then, the step is applied (see fig. 5.6a). A saccade follows because the error became too big. The convergence of the retinal error measurement and then a drift are also observed in the dark grey part. This drift is mainly due to the high pass filter used in the VOR processing. Indeed, the cutoff frequency is slightly too high and does not take into account the low speed movements of the robot. The eye (ψ_{er}) rotated and compensated well the rotation of the robot (ψ_{Robot}) at the beginning, but did not follow the robot movement after the overshoot (see fig. 5.6b). Therefore, the gaze orientation was not straight and showed a small drift, which lead to a second saccade and the convergence to a stable position (see the white part of the fig. 5.6). This position is very specific as the robot is at the end, heading in the same direction as the gaze, *i.e.* nearly at $-6°$ from the target direction, to have a retinal error of around 6°.

5.3.4 Robotic application of the Pursuit scenario with the Saccadic Eye Controller

A - Fitting the Saccadic Eye Controller into the Robot Controller

The structure of the robot controller with the Saccadic Eye Controller is similar to the one designed in [Kerhuel et al., 2010], as the Eye controller setpoint is also used as the yaw measurement in the robot controller. The robot controller in itself is very close to the one used in section 4.5.5 of chapter 4, because the control and the state estimation are using the same methods. Figure

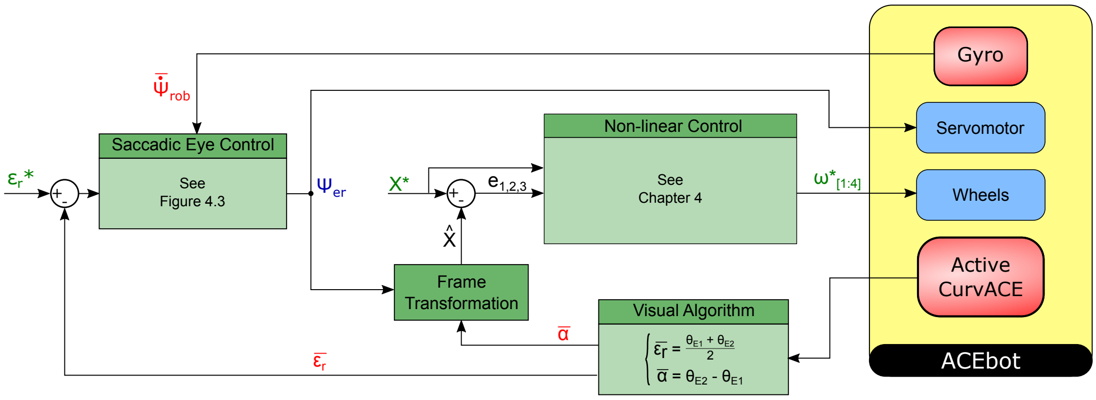


Figure 5.7: ACEbot controls its distance to the target using visual cues ($\bar{\epsilon}_r$, $\bar{\alpha}$) and the 4 wheel speeds. A rate gyro is added to provide a yaw speed measurement of the robot and a servomotor to rotate the visual sensor.

5.7 sums up the whole controller with the sensors and actuators embedded onboard ACEbot.

B - Experimental performance achieved

The same target trajectory as the one done in the section 4.5 to test the repeatability, was reproduce to test the response of this different control law. The results of several experiments can be seen in figure 5.8. The robot is able to follow a target accurately at a constant distance.

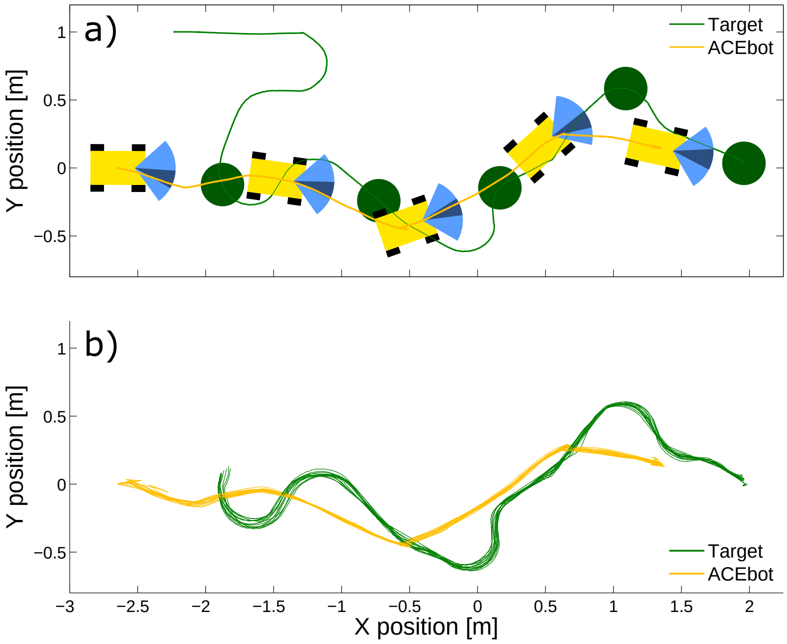


Figure 5.8: Pursuit trajectories with the Saccadic Eye Controller. a) One trajectory of the target and the pursuit behavior of the robot. ACEbot and the target are plotted every $6s$ and the first position plot is the moment where the pursuit starts. In dark blue, the measured position of the target in the active CurvACE sensor FOV and in light blue, the whole FOV used. b) All the trajectories made with the robot, with in green the target trajectories and in yellow the pursuer ones.

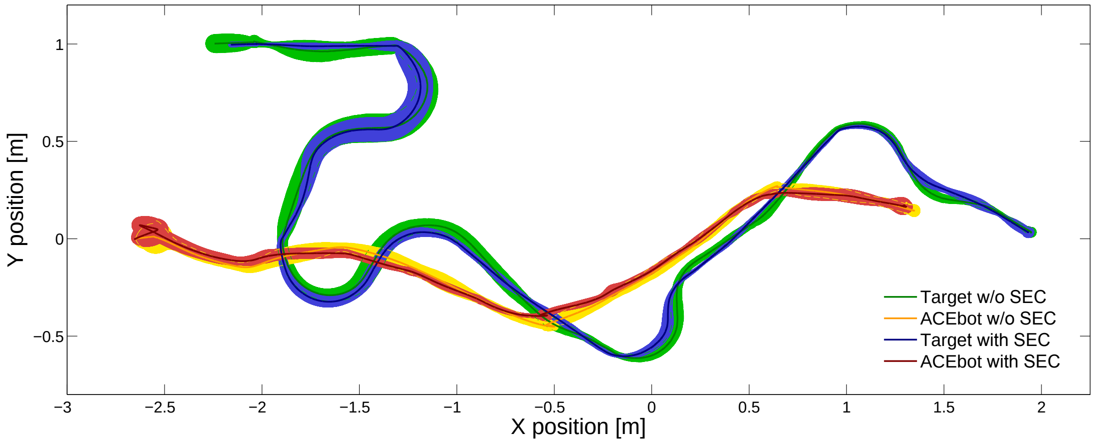


Figure 5.9: Pursuit trajectories with and without the Saccadic Eye Controller (SEC). The means of all trajectories without the SEC, presented in figure 4.10, are plotted in green and yellow for the target and the pursuer, respectively. The means trajectories with the SEC, presented in figure 5.8b, are plotted in blue and red for the target and the pursuer respectively. The colored areas are the standard deviation observed for the different trajectories realized.

5.3.5 Comparison between the fixed and decoupled eyes

It is shown in figure 5.9, that the results with a fixed or a decoupled eye in this experiment lead to similar results. These performances could be explained by the fact that the angular precision is the same over the entire field of view. Therefore, as long as the target is in the FOV, the localization is achievable with the same precision. The Saccadic Eye Controller displays no real advantage on this application with this target trajectory.

The other advantage could have been to reduce the number of the photosensors used and therefore the computational resource. Indeed, as the eye rotates toward the target, it should be in the center of the FOV more often. But as seen in figure 5.10, it is not very clear. Indeed, the maximal and minimal values of the edge positions measured are slightly equal during the whole pursuit. However, an interesting observation can be made at the beginning of the pursuit. In the case of the eye orienting itself toward the target, the FOV concerned at the starting point is always in the interval $[-20; 20]$. After, the FOV used is larger because the eye is moving too slow to keep the target in the central zone. If the time between the saccades T_{min} could be reduced, the number of photosensors used for the tracking could be also reduced.

In a situation where the target crosses the FOV very fast, an eye featuring saccadic movements could be an interesting feature. It could enable keeping the target in the FOV. This advantage has been established in [Kerhuel et al., 2010] which used a visual sensor with a FOV of $\approx 4°$. It could show also good performance with a sensor with a fovea (*i.e.* a higher resolution in the center of the FOV).

But, with our visual sensor, all of these possibilities are not shown. If the eye could move fast, the time convergence of the adaptation process of the CurvACE photosensors would make

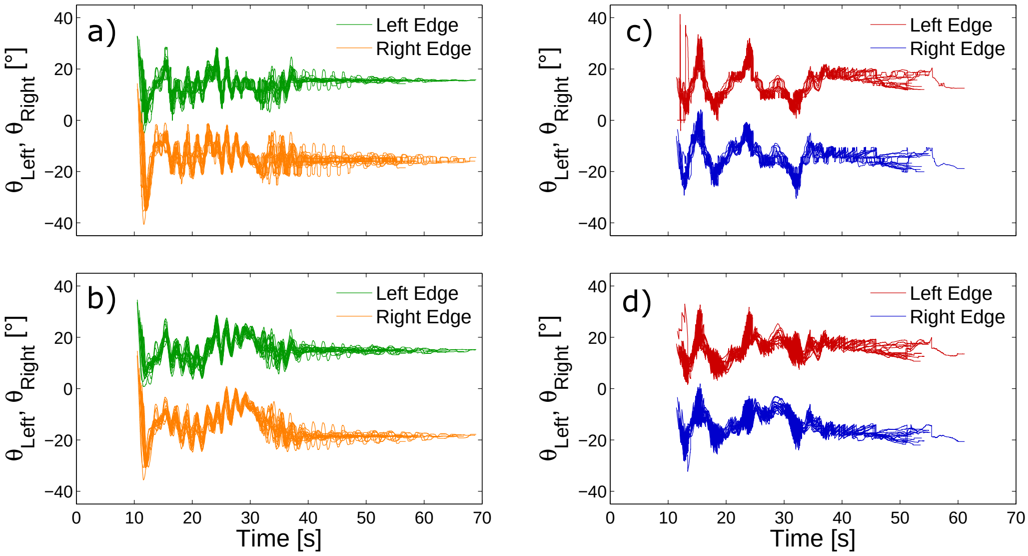


Figure 5.10: a) The edge position measurements in the FOV for all the pursuits with no eye rotation and b) the ground truth measurement in the same case. c) The edge position measurements in the FOV for all the pursuits with the SEC and d) the ground truth measurement in the same case.

it blind for a few moments, at least unable to perform tracking. To conclude, with this sensor and this robot, the advantage of using a rotational eye was not clearly highlighted.

5.3.6 Bio-inspired discussion

It was observed in nature that the insect uses such a strategy that the head is moving before the body with a faster dynamic response [Hateren and Schilstra, 1999].

It was shown thanks to the robotic experiment that the rotation of the eye is not needed. But in the case of the insect, the existence of a fovea (which is an acute zone, see section 1.2.2) could explain the need of such head re-orientation. The small resolution of some area of the FOV could be not sufficient to perform some tasks, such as target tracking. Therefore, the head movement orients the acute zone toward the target to fulfill the target localization task. This mechanism compensates for the non-uniform precision over the entire eye. It seems to be an interesting feature as motor-control is less computationally demanding than the visual processing of several dozen ommatidia.

5.4 Target detection approach

The visual algorithm that we developed to localize a target requires that the indexes of the 2 NWS to be initialized. To fulfill this purpose, a target detection algorithm must be developed. With the hypothesis that the target is the only moving object in the environment, or at least the environment moves slowly or with low amplitude, the objective is to detect the moving object with the target specific features (*i.e.* two edges sufficiently separated). As CurvACE responds to light variation, with an appropriate temporal filtering, it becomes possible to detect the target. Indeed, the response of a photosensor is dependent on the speed and the contrast. An appropriate selection can be applied to select the appropriate edges, then the NWS indexes. The algorithm developed will be compared to what is known in prey or mate detection in insects to see the similarities.

5.4.1 Constraints and hypothesis for target detection

The goal is to detect the cylindrical target used previously. As the tracking algorithm follows 2 edges, a criteria should be found to be able to discriminate the edges of the target from the ones of the experimental visual environment.

The visual sensor could be compared to a monocular camera in the fashion that it is impossible to detect the depth of the visual objects, without any motion. In this situation, the binocular vision of insect is not used as well to detect prey or mate because of the distance [Collett and Land, 1975]. The movement detection should be the best cue. Indeed, in our setup, only the target is moving and the world is considered still. The movement of the contrasts in the FOV cues is the optic flow. Therefore, it has the known property to be dependent of the speed and the distance of the target.

As the CurvACE sensor response varies with change of illuminance for each photosensor, because of the Delbrück circuit, the variation of illuminance is highlighted with strong output variation. With the hypothesis that the ambient illuminance is considered constant (or varies slowly), if the response of one photosensor changes, it is because an object has moved in the field of view of this photosensor. Therefore, it is an interesting feature to be able to detect the target.

Thus, the idea developed here, is to detect 2 moving contrasts with a sufficient angle between each other which are moving with enough speed. To do this, the vibration of the eye is stopped to avoid creating transient movements. The pixel response is filtered and triggered to detect this illuminance variation. As the pixel response depends on the speed of variation of the illuminance and the intensity change, to cross the threshold the target has to have enough speed and be salient enough compared to the background.

5.4.2 Target Detection Algorithm

The low-pass filter is an order 4 filter with a cut-off frequency of $30Hz$ and the high-pass filter is order 2 with a cut-off frequency of $0.5Hz$. The temporal filters used are inspired by [Expert et al., 2011], especially for the low-pass filter. The cut-off frequency of the high-pass filter is different, however, because in the optic flow algorithm its goal is to act as a forgetting factor to avoid using too old contrast detection that would have no relevance as compared to the movement of the robot. However in the target detection, its role is only to remove the slow variation of the ambient light, but should not be too high to detect target moving far away. Finally, these temporal filters have for objectives to reduce the noise and also to cancel the steady state response, in order to detect only sufficiently fast light change.

The absolute filtered photosensor signals are tested if their values are above a threshold (here, 4.5). This threshold is chosen accordingly for the target contrast and speed range. The difficulty is to have a low enough threshold value to detect low speed or far targets, to detect the target each time it is in the FOV, but also high enough to ensure the signal triggered is not noise, leading to a false detection.

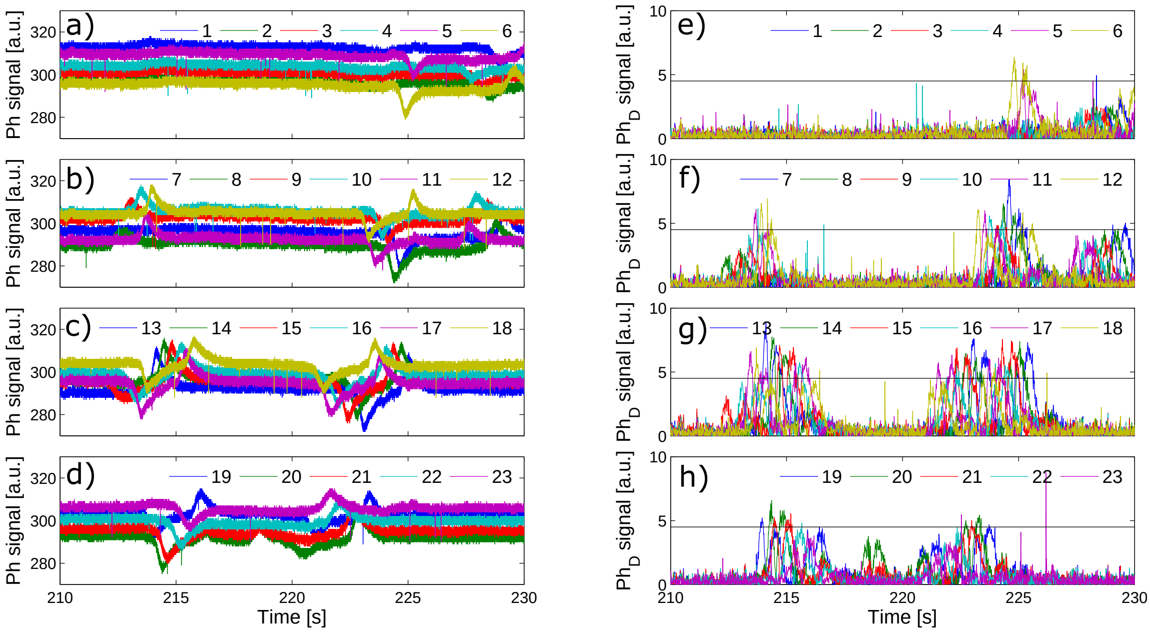


Figure 5.11: A target pass in front of the CurvACE sensor. a-d) Photosensor signals after the readout of CurvACE and e-h) Absolute filtered signal of the same photosensors. The black line shows the threshold used for the detection algorithm. The number of the photosensors goes from right to the left of the FOV, here centered in between the 11th and 12th photosensor.

After the trigger signal processing, two conditions should be met to start the pursuit. The triggered pixels should be:

- only 2 during $30ms$, or more than 2 during T_{detect} and the 2 last on are selected
- spatially separated by 2 others

These conditions ensure a factual and robust detection. The condition on the minimal time where 2 photosensors are triggered is to avoid false detection due to noise. The condition of having only 2 and to have them be separated by 2 others is to ensure that the localization processing will be able to work after the detection.

Algorithm 2 details how it is done. TMD refers to the triggered photosensor, the value it can contain are 0 or 1, for the non-triggered pixels and triggered ones, respectively.

The results of this detection can be seen in different situations: in the following behavior in figure 5.9 and in the interception behavior in figure 5.14.

A big assumption in the processing is to suppose that only the target is moving in the world or at least this environment is slowly changing. The possible limitation is that this selection process does not sense any difference between a big target moving fast far away and a smaller one moving slowly in front of the sensor. This behavior has not been highlighted during our experiments because the target was always the same.

Algorithm 2 Pseudocode for the detection process of the target

```
$t_0 \leftarrow 0$
acquisition $TMD$
if $sum(TMD) >= 2$ and $t0 = 0$ then
    $t0 \leftarrow time$
    while $(time - t0) <= T_{detect}$ and $sum(TMD) >= 2$ do
        $time \leftarrow time + Ts$
        acquisition $TMD$
    end while
    if $sum(TMD) >= 2$ then
        while $sum(TMD) \neq 2$ do
            acquisition $TMD$
        end while
        $ctrst \leftarrow 1$
        for $i = 1$ to $N$ do
            if $TMD(i) = 1$ then
                $idx\_old(ctrst) \leftarrow i$
                $ctrst \leftarrow 2$
            end if
        end for
        return $idx\_old$
    end if
end if
```

5.4.3 Target detection hypothesis in Biology

The target detection in the chasing insects is not very well known and different studies are suggesting different approaches.

A - Binocular vision

The Praying Mantis use binocular vision to estimate the distance of a prey in order to launch its attack [Nityananda et al., 2016].

A specific case of the dragonfly which has a large FOV covering by its 2 eyes varying from 22° above the head to 34 in the anteroventral part in *Aeshna interrupta lineata walker* [Pritchard, 1966] could be involved in target detection. But it is most likely to be used in the last phase of the interception process, because it starts its chase quite far from its prey.

B - Velocity Size Ratio

The Killerfly target detection is based on a proportionality between the speed and the subtended angle of the target in its field of view [Wardill et al., 2015]. The identified parameters of the target detection process seems to be the subtended angle and the angular speed of the target. But nothing ensure to detect a close small target at a low speed from a big one seen further at high speed.

C - Subtended angle profile over time

In [Olberg et al., 2000], the angular velocity evolution, (*i.e.*, angular acceleration) is thought to be used for the target detection and discriminate between nearby small objects (insects) and distant large ones (birds). As shown in figure5.12, for the same subtended angle seen at the shortest distance between the objects and the dragonfly, the angular velocity evolutions of the two objects are different and these variations can be used for the target discrimination. Furthermore, the evolution of the subtended angles is different which could be also used as a cue.

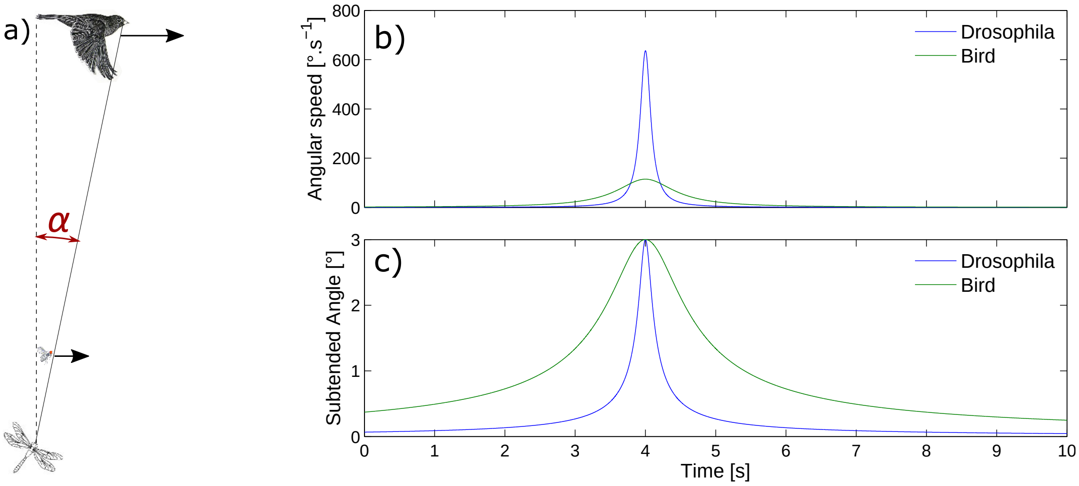


Figure 5.12: a) Schematic representation of a dragonfly seeing a drosophila and a bird in its FOV, at a distance of $5cm$ and $5m$ respectively, both seen with a subtended angle of 3° at those two distances. The drosophila and the bird are moving in straight line at $50cm.s^{-1}$ and $10m.s^{-1}$ respectively. b) Theoretical angular speed of both animals measured by the dragonfly. (a and b are modified from [Olberg et al., 2000]) c) Theoretical evolution of the subtended angle of drosophila and bird passing in the dragonfly FOV.

However, the case presented here is particular. Indeed, another particular case where the speed of the bird is now considered $50m.s^{-1}$ would show the exact evolution of both the angular speed and the subtended angle.

Cases study It can be shown in 2 cases presented in figure 5.13 that only the ratio between speed and the distance is important. The two cases are perhaps not biologically relevant but if a constant target speed is considered, scaling and combining these 2 trajectories can describe a random trajectory. Therefore, they are very simple trajectories that highlight the physics principle that the ratio between speed and the distance is not sufficient to discriminate the distance to the object.

First case: rectilinear trajectory Let be an object moving at a constant speed V_o on a straight line. Let be x the position of the object on the line with:

$$x(t) = V_o \cdot t \tag{5.1}$$

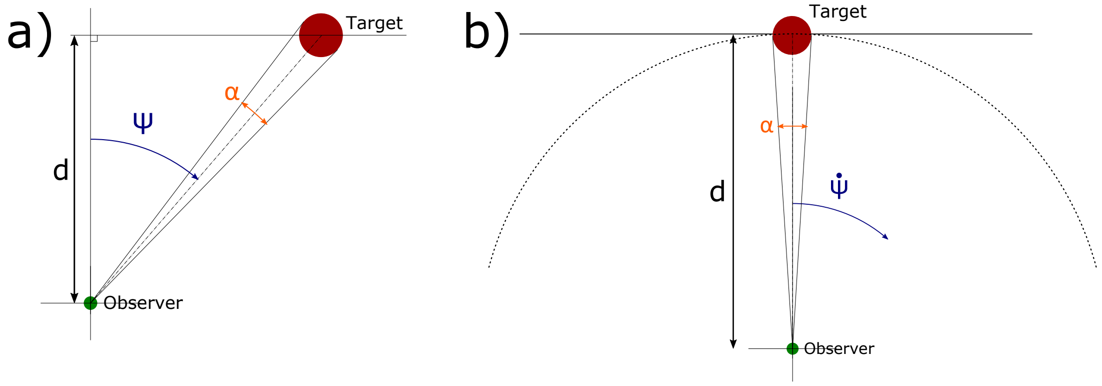


Figure 5.13: a) First case, the target is moving along a straight line at a constant speed. In this situation, the distance between the target and the observer is changing. It is decreasing to the point where the distance is minimum (*i.e.* the point where the target trajectory is perpendicular to the line of sight, the line between the target and the observer) and then the distance increases after this particular point. The subtended angle and the angular position of the target are therefore varying in the observer's FOV. b) Second case where the target is moving along a circular trajectory centered on the observer at a constant speed. The subtended angle is constant and the angular speed in the observer FOV is also constant in this situation.

where t is the time.

If d is the shortest distance between a given observer and this object (see figure 5.13a), the angular position ψ of the object in the observer's FOV is:

$$\psi(t) = \arctan\left(\frac{x(t)}{d}\right) \tag{5.2}$$

The angular speed is therefore the time derivative of ψ:

$$\dot{\psi}(t) = \frac{\dot{x}(t)}{d} \cdot \frac{1}{1+\left(\frac{x(t)}{d}\right)^2} = \frac{V_o}{d} \cdot \frac{1}{1+\left(\frac{V_o \cdot t}{d}\right)^2} \tag{5.3}$$

This equation highlights that the evolution of the angular speed in this situation is dependent on the ratio linear speed over the distance. Then, for a given distance, a speed can be found to match a given angular speed profile.

A similar observation can be done for the subtended angle $\alpha(t)$:

$$\alpha(t) = 2\arcsin\left(\frac{diameter}{2\sqrt{x(t)^2+d^2}}\right) = 2\arcsin\left(\frac{diameter}{2d\sqrt{\left(\frac{V_o \cdot t}{d}\right)^2+1}}\right) \tag{5.4}$$

where $d > 0$. If the subtended angle is the same, the ratio $diameter/d$ will be constant and if the ratio V_o/d is also constant, then the subtended angle evolution will also have a similar profile according to the time.

case: the circular profile The same observation can be done on a circular profile, *i.e.* a target that does circle around the observer (see figure 5.13b).

As it is known, in the case of a circle, that:

$$V_o = d \cdot \dot{\psi} \tag{5.5}$$

The relation between the angular speed and the ratio V_o/d is direct. The distance is also maintained as constant, so the subtended angle is also constant.

D - Is there another cue used to discriminate a suitable target?

From these two situations described, it could be extended that for a target moving at a constant speed, it is not possible to discriminate if it is a small target at a short distance or a big target at a large distance as a speed profile can be associated in both cases to match the same angular speed profile. But are all the ratios of speed over distance even possible? Maybe not, seeing a bird flying at $5m$ from the ground at $180km/h$ is not usual and even not achievable by many of them. So, this observation could be enough because nothing in the animal kingdom matches.

It was observed that a specific ratio between speed and size trigs the chase in the killerfly [Wardill et al., 2015]. Maybe the identified ratio explains the takeoff, but other information can be used during flight, like the motion parallax, to check if it is a suitable prey or not and therefore abandon the chase. It would have been interesting to see the catch rate or the duration of chase for the different target sizes and speeds. Indeed, in this study, the takeoff is observed but what happens after could also shed light on the process.

Indeed other cues can be used to discriminate targets before initiating pursuit. If we observe a typical insect flight, it can be seen that saccades, *i.e.* fast body yaw movements, appear during flight. These kinds of flight patterns could be used to discriminate a suitable target. To be effective, the chaser would need to analyze the flight along time. But the frequency of these saccades seems to depend on the nearby obstacles [Lindemann et al., 2008, Kern et al., 2012]. Moreover, the saccade are also accompanied by sway movements which smooth the trajectory and make them difficult to detect in case of small amplitude [Schilstra and Hateren, 1999]. Detecting the quick change of the orientation of the target could be possible, but in the case of interception and closing quickly to the target, it should be difficult to achieve with the expansion of the target that constantly vary. Nothing for now supports this theory and the interception is perhaps too fast for dragonfly to do such processing (duration of flight up to $400ms$ [Olberg et al., 2000]).

Another hypothesis would be that the insect is still able to recognize a shape between two possible targets. This is perhaps not true because flies reacts to a ball [Wardill et al., 2015, Boeddeker et al., 2003]. However, a bird could less be confused with a sphere than a drosophila for example. As birds do not have transparent wings, their flapping wings are modifying their shape in the visual field of view. This could be a solution to avoid takeoff for a bird. Therefore, it could be only rough shape recognition involved in the Target identification in addition to the angular speed of the target in the pursuer FOV.

5.5 Bio-inspired interception behavior

In section 5.2, it was shown that a target could be followed with a similar trajectory as compared with a hoverfly thanks to a basic position controller based on the distance and maintaining an angle between the target and the follower to 0°. Other experiments made on the mating of the blowfly *Lucilia* displayed a smooth pursuit of the chaser regulating its speed according to the distance and its orientation to keep it in the center of its FOV [Boeddeker et al., 2003]. Therefore, we tested the possibility to realize this kind of interception trajectory with our controller. As other insects exhibit more complex strategies (Hoverflies like *Eristalis* and *Volucella* [Collett and Land, 1978] or dragonflies [Olberg et al., 2000]), the effect of only changing the retinal error setpoint was investigated.

5.5.1 Robotic experiments

A - Interception with target detection

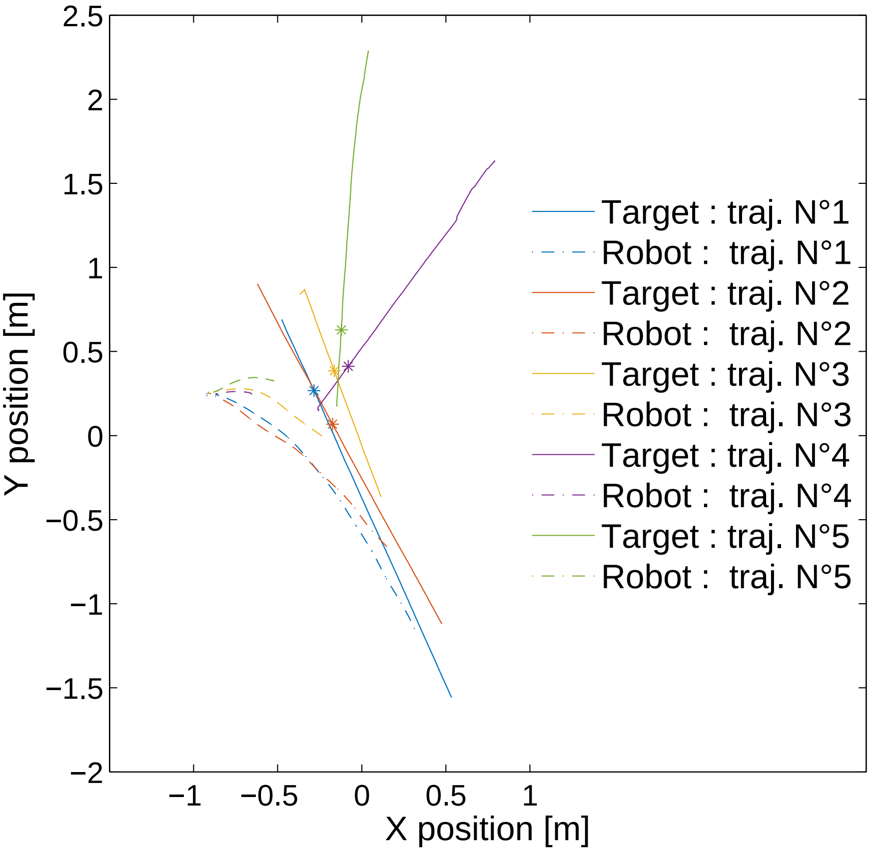


Figure 5.14: Interception trajectories with an eye-in-robot angle of 0°. The trajectories of both, the target and ACEbot, are the trajectories of the inertial center point. The continuous lines are the target trajectories and the dashed lines are the pursuer ones. The star displays the position of the target when the detection occurred and the interception begins.

The interception trajectories of the robot were realized only by modifying the distance reference of ACEbot in the control loop (see figure 5.14). The setpoint given for the X^* reference is

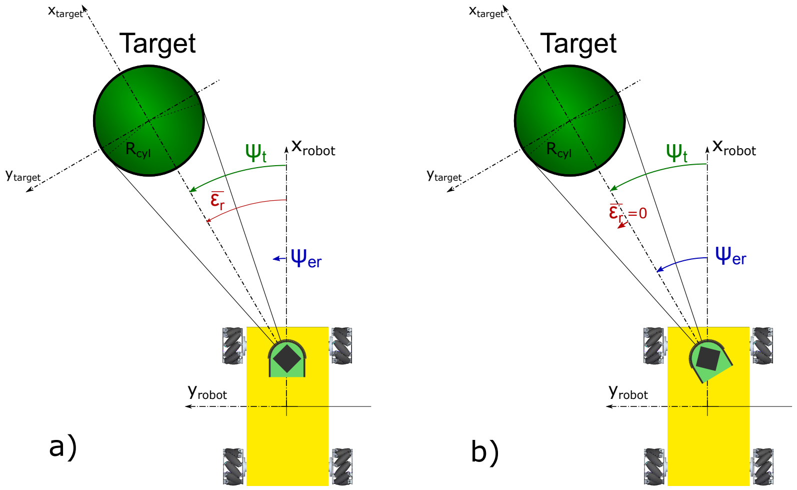


Figure 5.15: a) ACEbot and the target with an Eye-in-Robot angle (noted ψ_{er}) equal to 0. b) ACEbot and the target with the retinal error ϵ_r equal to 0. In all cases, the orientation of the target noted ψ_t satisfies the equation $\psi_t = \psi_{er} + \epsilon_r$.

$1m$. It means that the robot tries to go further than the target. A value of 0 could not have been used, because it would have meant that the speed of the robot should equal 0 at the moment it reaches the target. It would therefore have been impossible to reach the target because of the deceleration.

In the experiments, the interception is considered done if the distance between the inertial center of the target and the pursuer robots is under $0.46m$. This distance is chosen to avoid collision between the two robots.

Figure 5.14 shows different target trajectories and the interception realized by the pursuer. All the target trajectories are linear but with different orientations. This also shows that the detection process is able to detect the target in different situations.

B - Comparison for different retinal error setpoints

To test the effect of the retinal error setpoints on the pursuer trajectory, it was preferred to change the orientation of the eye to ensure having the larger FOV when the target is close to the reference, to avoid losing the target. The control of the eye is still to keep the retinal error to 0°. The effect is the same as maintaining the target at an angle equal to the eye-in-robot angle (see figure 5.15).

Figure 5.16 shows that in the case of quasi linear trajectories of the target. The interception trajectory is faster when the retinal error is larger. It can be seen that the pursuer is heading away from the target, but toward the target's future position. It leads to a straighter trajectory. The perfectly straight one is not achieved here. The retinal error setpoint should be chosen carefully knowing the target speed and its own acceleration and speed capacity. But the target trajectories are not perfectly linear either; a better position controller of the target should be developed in order to make more reproducible trajectories.

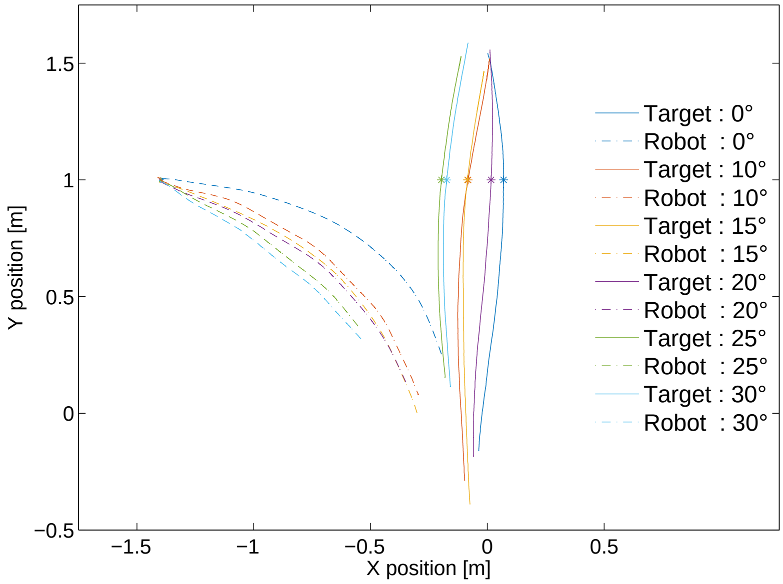


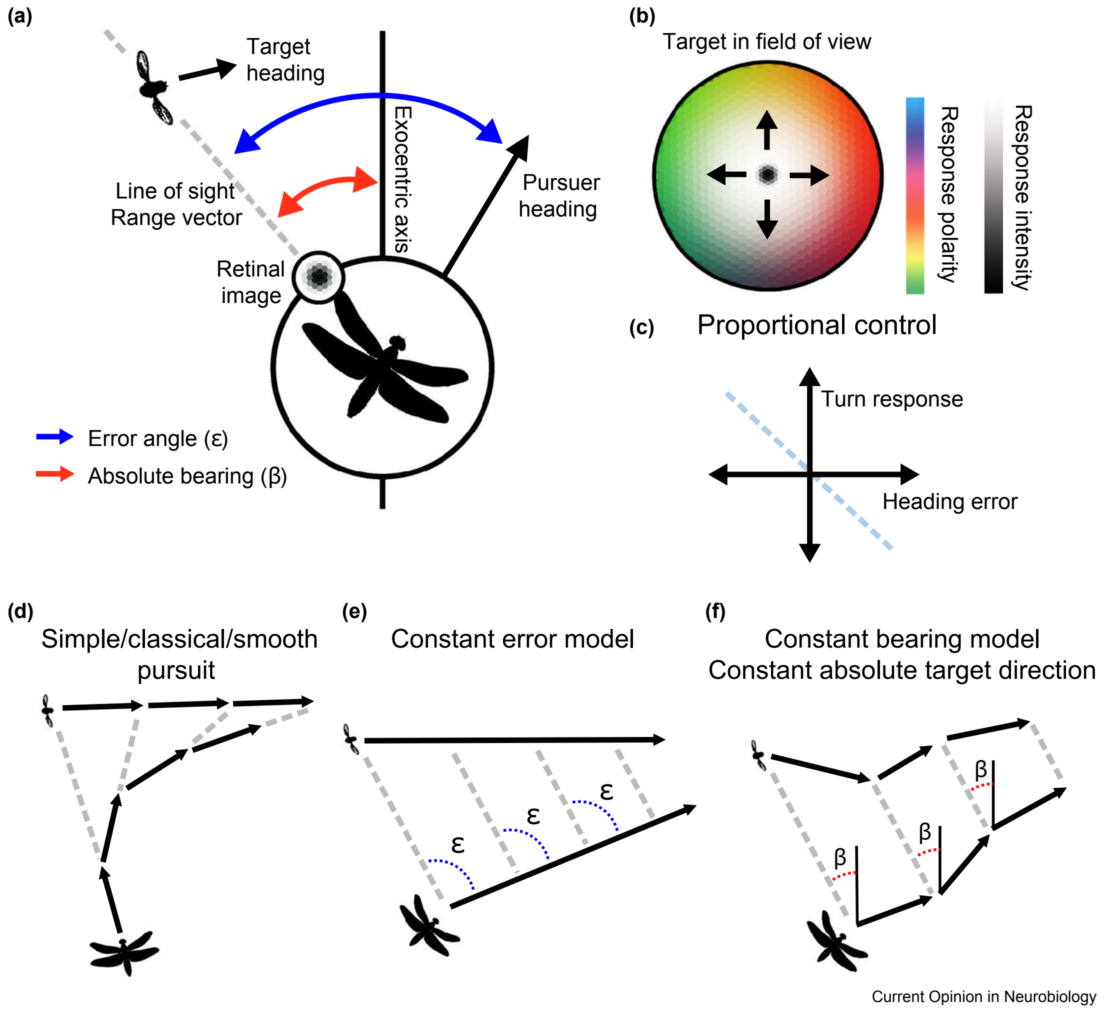
Figure 5.16: Interception trajectories with different eye-in-robot angles. The trajectories of both, the target and ACEbot, are the trajectories of the inertial center point. The continuous lines are the target trajectories and the dashed lines are the pursuer ones. The star displays the position of the target when the interception begins, the detection was made previously.

5.5.2 Discussion with respect to the biological counterpart

Small insects are, for example, relatively simple living things but have to be capable of catching prey to survive and mate to perpetuate the new generation. However, all species are well adapted to the prey they are chasing and have developed different interception strategies (see figure 5.17). The first and what seems the most simple interception strategy is to track the target by keeping it in the center of the FOV. The second one could be summed up by keeping the error angle to a constant value, which is what has been done in figure 5.16. It is also the same principle used in nautical navigation, that if a ship is getting closer at the same line-of-sight angle, it will end in a collision [Murtaugh and Criel, 1966]. The third one would be to keep an angle between the target position and an external reference constant, also called the bearing angle. The two last strategies lead to head toward a possible future position of the target.

Some insects perform the most simple interception trajectory, like the blowfly *Lucilia* [Boeddeker et al., 2003]. In [Boeddeker and Egelhaaf, 2005], the authors do the hypothesis that the heading direction is assessed in order to keep the target in the central part of the FOV and the forward speed is controlled non-linearly according to the subtended angle of the target. I think it is a more complex solution than having a position controller based on the subtended angle measurement, because the function that regulates the speed does not take into account the specific target speed, except in the case that all the females go at a similar speed. In the case of the housefly (*Fannia canicularis*), it tracks the female by always flying toward it and makes no assumption on the future position of its quarry. But it seems to use a feedback loop on its speed when it is close enough from the heading axis [Land and Collett, 1974].

Hoverflies like *Eristalis* and *Volucella* are also capable of computing an interception course to catch females based on optic flow measurement, position of the target in its field of view



Current Opinion in Neurobiology

Figure 5.17: Target pursuit modes. a) Summary of the terminology. b) The target's position in the pursuer's field of view affects its response, such as turning direction and response intensity. c) Proportional navigation is where changes in either the pursuer's forward or angular velocities are proportional to the size of the heading error. d) In simple pursuit the pursuer aligns its axis directly towards the target's current position. e and f) During interception, or parallel navigation, the pursuer heads towards the target's future position. e) One way is to keep the error angle ϵ constant. f) Alternatively, the absolute bearing angle (β) is held constant against an external reference point. (Reprinted from [Gonzalez-Bellido et al., 2016])

and known constant values: its own acceleration, the size of the target, and the distance of detection [Collett and Land, 1978]. However, these trajectories with anticipation are optimal if the pursuer is much faster than the target and this one do not do quick turn, it is highlighted by the small time of chase - under a second.

Dragonflies are performing smart interception trajectories; their strategy consists of keeping them in the accute zone of its eye, which leads to fly to the interception point [Olberg et al., 2000, Olberg, 2012]. They seem to be endowed with a prediction capability to anticipate target movements as their reaction time is very fast ($\sim 26ms$) [Olberg et al., 2007]; even under $4ms$ was recorded in the head movements when the target moves at a constant speed and there are no unexpected movements [Mischiati et al., 2015]. They are also able to predict the area of the future target position when it becomes invisible because of lack of contrast [Dunbier et al., 2012].

It will require a lot of work to completely understand all the parameters at work during a target interception, from the visual perception to motor control. We had learnt a lot on the subject of visual processing but we are still not able to reproduce completely all the insects' capabilities. In this short summary and during the robotic experiments, we did not even express the difficulty of target tracking in cluttered environment and the implication in the interaction between the tracking and optomotor systems, as described by [Collett, 1980].

Chapter 6

Conclusion

Contents

6.1 Accomplished work

During this thesis, the extension of the edge and bar localization processing unit (referred previously as LPU) was realized with an array of artificial ommatidia. It was established that such algorithm can provide visual feedback for stabilization over a textured ground. Some limits were also pointed when it comes to the linearity and scaling of the output when the FOV is too small.

A calibration process was proposed to linearize the output when a bar is identified in the FOV by using the neighboring photosensor pair. It could also provide the subtended angle as well as the angular position of the bar. The algorithm yielded good results, but it was very sensitive to the speed of the seen object and to the direction of light. Indeed, some shades could appear and modify slightly the contrast and create an asymmetry between left and right contrasts of the bar.

This solution was therefore abandoned for robotic application. Then, a work on the electric fish involving Gaussian sensitive receptive field neuron for a stimulus localization was thought to solve partially the problem. Indeed, with a Weighted Sum of Gaussian like sensitivity function, a linear output is provided. The same situation is encountered when an edge is seen in front of a vibrating photosensor. It was therefore established that the Normalized Weighted Sum leads to hyperacuity. Theoretically, the application is limited to edge localization, but can be applied to bars with limited error measurements (see section 6.2.4). The advantage is to avoid a complicated calibration process to transform the LPU output into angular measurement.

This novel algorithm depicted in this PhD thesis was used for target localization. A known cylinder was put on a rover and was pursued with a robot equipped with an active CurvACE sensor named ACEbot. The two edges of the cylinder were tracked. As the size of the cylinder is known, the subtended angle measurement, inferred from the edge positions, was enough to recover the target distance. Then, ACEbot was able to pursuit the target.

Some biological comparison were made and it appears that the ACEbot behavior was very close to the pursuit behavior of the hoverfly. We showed also that a decoupled eye is not of great interest in the case of a wide field of view with homogeneous accuracy. Indeed, having a rotation of the eye enables to keep the target in the center of the acute zone in insects and human. In our case, the resolution is homogenous, because the same algorithm is computed on all the photosensors involved in the visual task. Hyperacuity has been observed in insect vision, but not over a wide FOV. It seems that it is not needed and can be compensated with the head movements. Indeed having a small hyperacute zone, identified as a fovea can be sufficient to solve most of the problem meet in fly, without requiring high computational load. Discussion were made on the interception task and only two light modifications were made in the robot's control system to make it able to perform such behavior. It can point toward the future location of the target but another criteria depending on the target speed should be added to orients the robot in the right direction, which was assumed in our experiments.

6.2 Future possible improvements and perspectives

6.2.1 CurvACE sensor: previous engineering trade-off and possible improvement in a future design

The CurvACE sensor has been nicely inspired by insect vision. It reproduces bio-inspired features that are very interesting:

- A wide Field of View
- A Gaussian light sensitivity with a width at half maximum $\Delta\rho$ similar to the insects' compound eye
- A spatial resolution $\Delta\varphi$ equal to $\Delta\rho$, which enable an overlapping of the FOV of the artificial ommatidia.
- A light adaptation at the pixel level

However, it has also some drawbacks mainly due to engineering trade-off.

A - Engineering limitations

The Analog-to-Digital Converter (ADC) is not perfectly adapted to the tension measured by the sensor. The measured voltage can range from 500 to $1500mV$. But the lower bound of the total ADC range is from $0V$, reduces greatly the accuracy of the sensor due to the quantization. This issue came from the limited choice of ADC available in VLSI technology at the design time. But for a next sensor design, it is a point that needs to be carefully considered.

The Signal-to-Noise ratio should also be improved. The size of both the optics and the pixels could be a way toward this goal. If the optic diameter is increased, it means that more photons can hit the pixel (it means decreasing the F-Number, equals to the focal lens over the diameter, which in photography lead to brighter image). The other possibility is to increase the size of the pixel. But keeping the Gaussian angular sensitivity thanks to specific focal lens and aperture pattern, may make this option impossible in practice.

The other difficulty is the data rate possible through the SPI bus at the output of the sensor. In practice, it limits the possibilities of the sensor and a trade-off has to be made between the number of pixels read and the frame rate. Therefore, the use of Region of Interest (ROI) is required in most application, which is not user-friendly in addition to be binding. Maybe the use of the camera standards could be used, as the Camera Link which can provide a bandwith of $680 MB/s$, or the GigE Vision standard to replace the SPI bus. But the integration of a fast camera link could be difficult in terms of integration.

B - New groundbreaking techniques applicable to CurvACE

It was shown the possibility to go from a simple model of 1D world with simple hypothesis that it is composed of edges and bars. But it has been also seen that this hypothesis does not always hold and therefore a better algorithm should be found. I really think that going in the way of a 2D array of pixels with equals interommatidial angle could help to develop better algorithm to deal with more natural patterns. A hexagonal layout of the photosensor allows such property. But the assembly of columns mixed with a bent PCB would not be possible to create this new kind of sensor.

As presented in the section 1.3.5, event-based camera are already on the market. One can imagine to mix the technology and the knowledge of the compound eye with event-based technology. Indeed, the minimalist approach of using a compound eye to reduce the data with a low spatial resolution could be mixed with the event-based principle, which reacts to visual scene variation. It is also a way of reducing the data to be processed but without losing information, as it updates only pixels with a changed value. A such sensor would be of a great interest for small robotic application.

In the future, it might be possible to print a new kind of artificial compound eye thanks to 3D printers. Today, it is possible to mix different material with one 3D printer and produce electronic devices including simple display [Willis et al., 2012]. Moreover, it is already possible to print optics thanks to 3D printer (see Printoptical ©Technology for example). Maybe in the future a spherical artificial compound eye would be realizable including optics and electronics. Even if it is not ready to use and need some assembly process, this new way of manufacturing complex designs is maybe the future of these artificial compound eyes.

6.2.2 Discussion about demodulation processing

A - New solution inspired from neural networks

As shown by Carriot [Metzen et al., 2015], the primate vestibular system is able to discriminate different frequency of a signal. Therefore, it can code the amplitude of the carrier signal, which can be considered as an envelop detector.

[Roberts et al., 2015] shows also that training neuron to perform temporal correlation is possible, which could open the way to the reproduction of a neural network able to perform demodulation processing.

Moreover, the photoreceptor signals could be processed differently. Indeed, we are currently using a known carrier frequency and do a demodulation accordingly, but with a neural processing, it could be possible to have the frequency of the carrier and the amplitude of the signal. It could lead to a phase descriptor to have the position of the retina in phase with the visual signal. It could also be a possible solution to avoid the use of an absolute operator in the demodulation process to have the ON-OFF or OFF-ON information. It was highlighted in [Juston et al., 2014], that the phase signal contains information. Moreover, it could be possible to use a different speed profile which would lead to a variation of frequency according to the position of the retina in order to discriminate different objects' position. Or another thing would be to explore the possibility of a feedback loop on the scanning amplitude according to the visual signal.

B - Bio-inspiration relevance

It should be very interesting to just think about what are the parameters that can be controlled in the vibration actuation. Indeed, the parameters are the frequency, the speed profile, the amplitude and the direction of the scanning. For now, only 1D scanning was experimented on real sensor. Works were made with a triangular and exponential speed profiles to detect edge location [Viollet and Franceschini, 1999] and perform obstacle avoidance [Mura and Franceschini, 1996].

Then, different speed profiles were investigated in [Kerhuel, 2009]. It was highlighted that to work on the amplitude modulation, it was a pure sinusoidal signal that provide the best output, because only one frequency is stimulated. But Kerhuel showed that an adaptation of its algorithm could also work with the vibration generated by the robot itself.

R. Juston, in [Juston, 2013], used directly the improvement of using a larger scanning amplitude that could enhance the signal because the variation in one period is bigger and the signal-to-noise is reduced at a given noise level (see remark3). Moreover, with a higher amplitude at the same frequency, the rotational speed of the eye is increased, which leads to a signal of higher amplitude. In those applications, it is assumed that nothing moves during a period of scanning hold. It leads to the observation that to increase the dynamics of the robots, the frequency needs to be the highest possible (in the last experiments in the lab, around $50Hz$). It is not close to the original observations, which show frequencies between $5 - 7Hz$ [Franceschini, 2014].

The observation made by Northrop seems to shows a correlation between the speed of the target and the amplitude of the vibration [Northrop, 2000]. However, the feedback loop on the amplitude of the scanning regarding some parameter on the signal received was not investigated on artificial eye. But a research in this direction could open new hypothesis on the visual processing of the flies.

6.2.3 A challenge of visual perception: Vision in the dark

By using powerful Infrared (IR) leds, it becomes possible to see in the dark, when dark is defined from a human perspective. As the spectral sensitivity of the photo-diodes used in CurvACE is in the red and near IR light, the visual sensor can still see though the human eye can no longer do. Thus, two main situations of interest can be considered with IR leds.

A - IR leds as headlight

The first possibility is to use the IR leds on the robot and emit the light in the forward direction. The IR leds are used the same way as it is for headlight in a car or active IR cameras. The light is reflected by the object to track, which makes it visible. The active CurvACE sensor and the processing remain the same in this situation.

Another possibility is to modulate the infrared signal and adapt the processing after the photodiode transduction with a peak filter centered at the frequency of the modulation. This processing does not require any mechanical vibration which is replaced by the LED flickering.

B - IR leds as beacon

The second possibility is to use the IR led as beacon. This principle is already used by hypercube sensor [Raharijaona et al., 2015], presented in figure 6.1. The IR light beacon is modulated at a known frequency and the appropriate demodulation processing is done. With 3 photodiodes, the 3D orientation of the beacon relative to the sensor can be computed.

If the beacon is a fixed IR light source, the vibration is required with CurvACE to avoid fading due to the adaptation. The processing could however remain the same. The beacon would be seen as a point in the FOV. In one dimension, it is similar to a tiny bar with high very high contrast. A simulation was done with this hypothesis with the results presented in figure 6.2. The simulated bar is $1mm$ large seen at a distance of $1.5m$ with a contrast higher than 99.98%. The beacon is perfectly seen but the demodulated signal is not a Gaussian anymore, resulting in more linearity error after the NWS calculation.

6.2.4 Localization of a bar

A - With periodic scanning

Figure 6.3 shows the response of a 1D array of 6 photosensors rotating in front of a bar and submitted to a sinusoidal scanning of an amplitude of $\frac{\Delta\varphi}{2}$ at $50Hz$. Different subtended angle are tested. It shows that for small subtended angle, relatively to the $\Delta\varphi$ parameter, the output is quasi-linear. However, when the subtended angle gets closer to $2.\Delta\varphi$, the NWS output is becoming a staircase signal. It would not display any hyperacuity in such case.

It is interesting to notice that using only 2 photosensors to compute a Normalize Weighted Sum (noticed NWS_2) would provide a better linear approximation. The NWS_2 signal plotted in figure 6.3c) is defined as follows:

$$NWS_2 = \frac{(i+1) \cdot Ph_D(i) + i \cdot Ph_D(i+1)}{Ph_D(i) + Ph_D(i+1)} \tag{6.1}$$

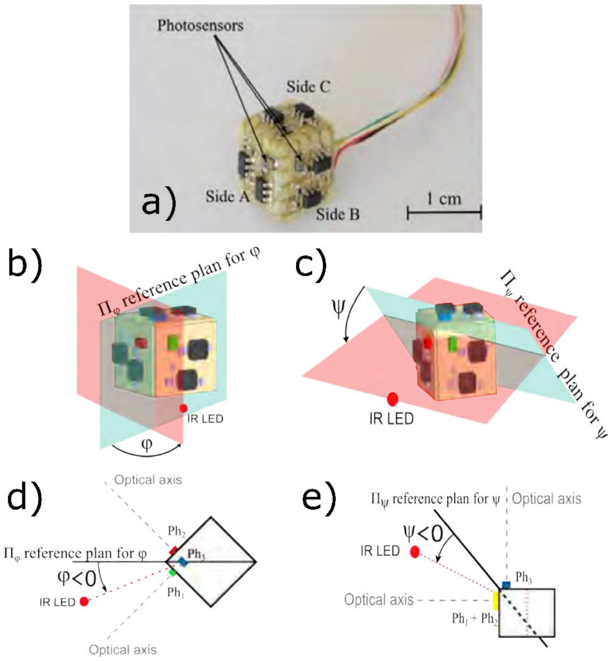


Figure 6.1: a) Each side of the HyperCube sensor integrates one photosensor and an analog amplifier for the conversion of the photodiode current into an output voltage. b) Front view, which gives an illustration of the azimuth φ and the reference plane Π_φ. c) Front view, which gives an illustration of the elevation angle Ψ and the reference plane Π_Ψ. d) Top view with the reference plane Π_φ, which is the plane through the IR LED and the optical axes of the photosensor Ph_1 and the photosensor Ph_2. e) side view with the reference plane Π_Ψ, which is the plane through the IR LED and the optical axes of the virtual photosensor $(Ph_1 + Ph_2)$ and the photoreceptor Ph_3. (Reprinted from [Raharijaona et al., 2015])

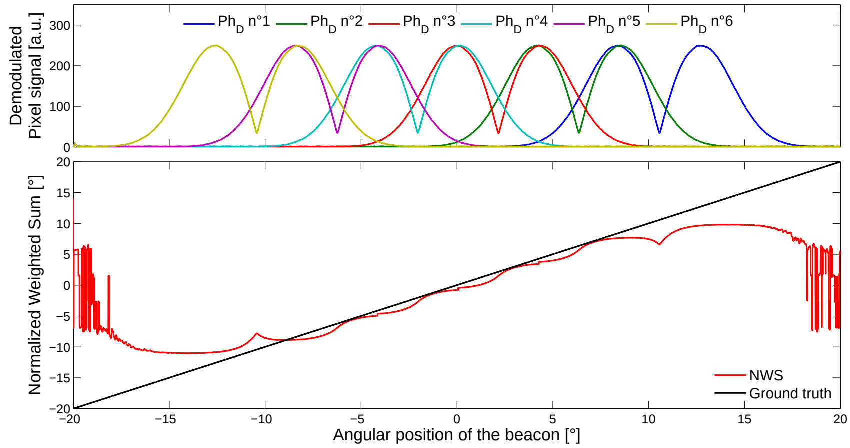


Figure 6.2: Simulated IR beacon seen at a distance of $1.5m$ by 6 photosensors separated by an angle of $\Delta\varphi = 4.2°$ and rotating at an angular speed of $1°.s^{-1}$ a) Demodulated signals of each photosensor according to the angular position of the eye. It could be seen that as the light is directly received from the source, the SNR is higher than normal. b) Normalized Weighted Sum calculated according to the position. It could be seen that the error is under 2°

.

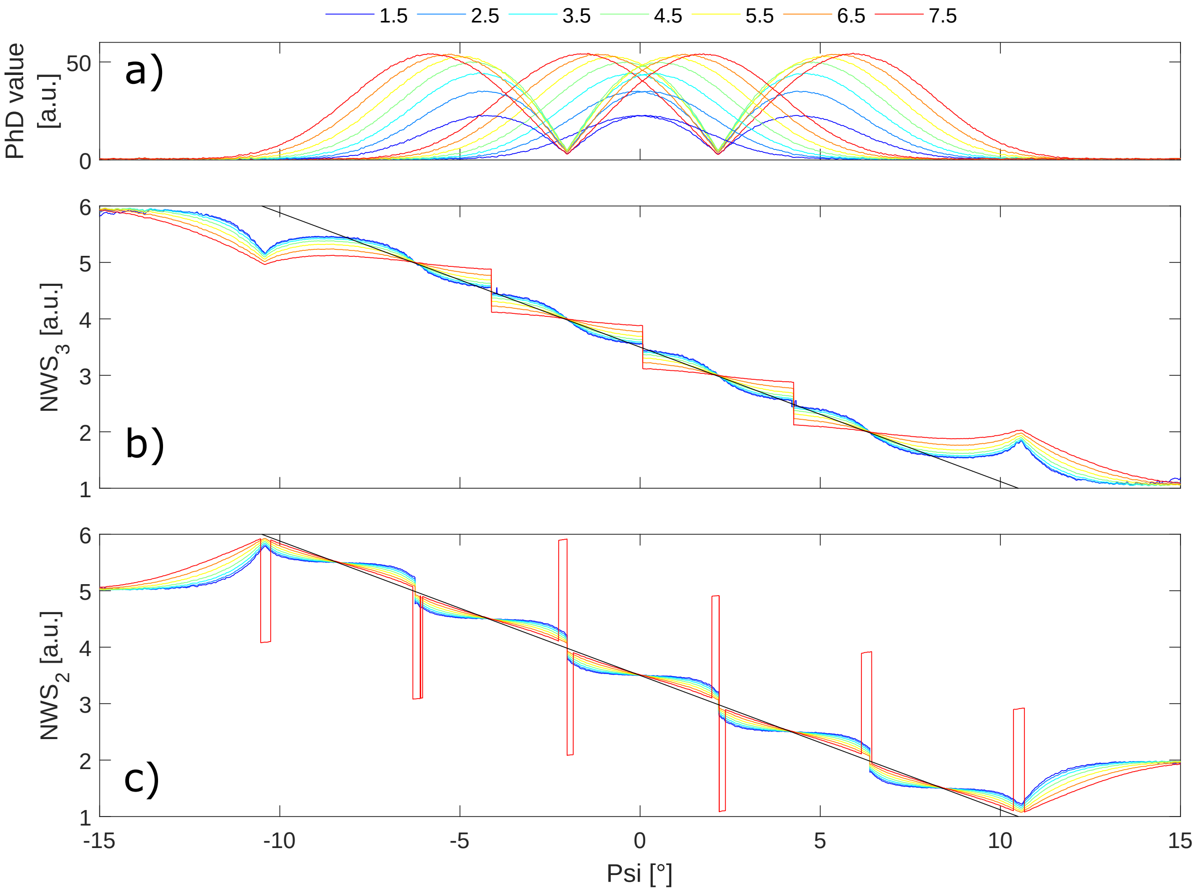


Figure 6.3: Simulation of a 6×1-array of artificial ommatidia centered in the middle of the 3^{rd} and 4^{th} one, rotating in front of a bar with a subtended angle from 1.5 to 7.5°. The visual parameter $\Delta\varphi$ and $\Delta\rho$ are both equal to 4.2°. a) Photosensor responses of the photosensors number 3 and 4. b) Normalized Weighted Sum of the 3 photosensors selected for the Fused NWS signal (here noticed NWS_3), according to the position of the bar for different subtended angle. c) Normalized Weighted Sum of the 2 photosensors selected (here noticed NWS_2 and see equation (6.1) for definition). The selection is based on the maximum of the sum of 2 neighboring photosensors.

It may need some improvements of the selection process, but it could lead to a better bar localization than with 3 photosensors for some subtended angles.

Remark 4 *It can be noticed that it is similar to the VODKA algorithm [Kerhuel et al., 2012]. Let* $S = \frac{a-b}{a+b}$. *If you set* $N = \frac{S+3}{2}$, *it would lead to* $N = \frac{2 \cdot a+b}{a+b}$, *which is the definition of* NWS_2 *in the case* $i = 1$.

B - Without scanning

With the method described in chapter 3 (*i.e.* using a Look-Up table), it seems difficult to get the value of the subtended angle of a target. But still, the hoverfly is capable of such behavior (for example see [Collett and Land, 1975]). A new idea that has not been evaluated so far, is to cancel the vibration of the eye in order to localize a small target. In the case of a sensor like the M^2APix [Mafrica et al., 2015], it is possible to see even in the case of no movement with a wide range of illuminance. It is possible because the adaptation process used the mean value temporally filtered as follows:

$$V_{out} = -K \frac{I_{ph}}{I_{ph} + I_{mean}} + V_{BG} \tag{6.2}$$

where K is a constant depending on conception parameters and V_{BG} is the band-gap voltage.

As the I_{mean} value is a low-pass filtered sum of all the pixels current, it means that if 2 pixels of the same chip see different pattern, they will respond differently, even in the steady state because the photodiode current is different for the two pixel.

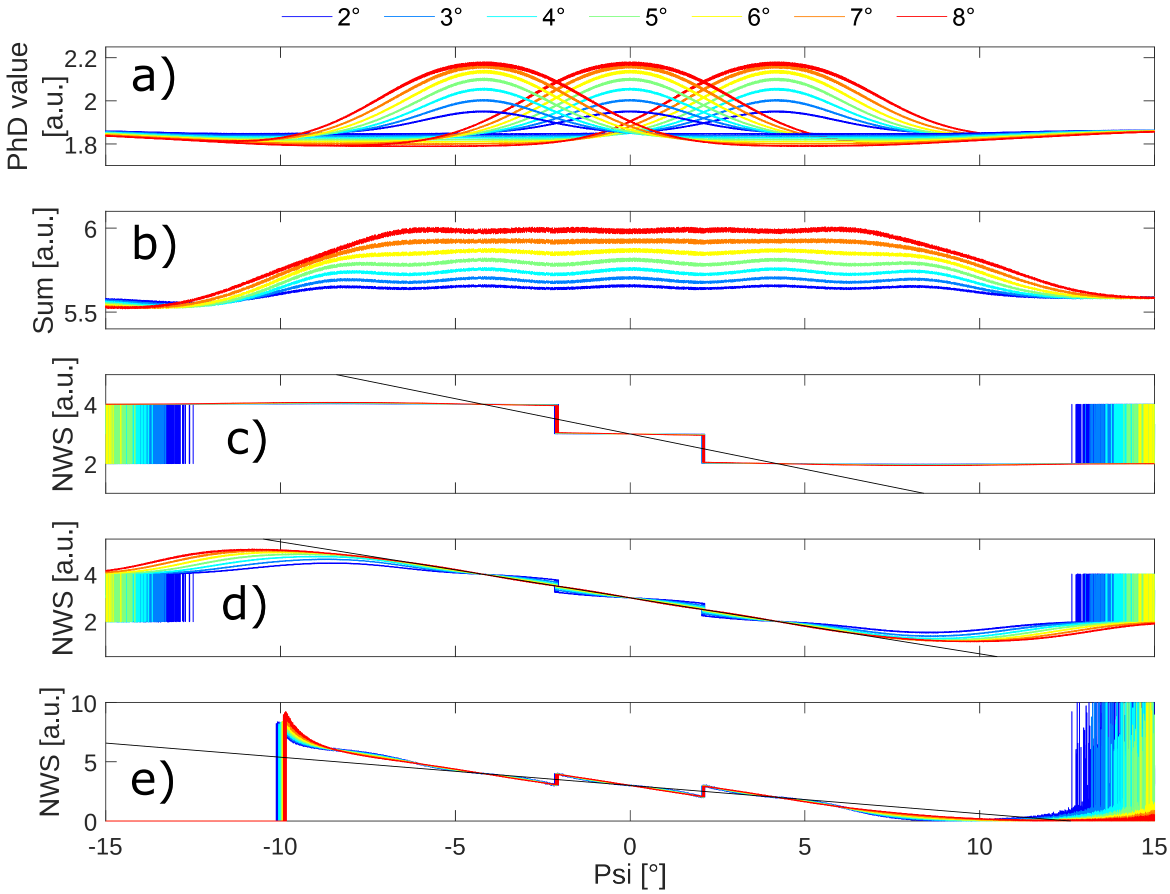


Figure 6.4: Simulation of a 5 × 1-array of artificial ommatidia centered on the 3^{rd} one, rotating in front of a bar with a subtended angle from 2 to 8°. The visual parameter $\Delta\varphi$ and $\Delta\rho$ are both equal to 4.2°. a) Photosensor responses of the photosensors number 2 to 4. b) Sum of the 3 photosensors selected for the Fused NWS signal, according to the position of the bar for different subtended angle c-e) Normalized Weighted Sum of the 3 photosensors selected for the Fused NWS signal, according to the position of the bar for different subtended angle. c) The offset of the photosensor response is not removed. d) The constant value 1.81 (offset in figure a) is removed to each photosensor signal before processing the NWS algorithm. e) The average value of all the 6 photosensors at each sample time is subtracted to the photosensor signals before processing the NWS algorithm.

Figure 6.4 shows that the simulated response of an artificial ommatidia to a bar placed in its FOV fits well with a Gaussian function when the subtended angle is under $2\Delta\varphi$. It highlights the fact that the NWS algorithm can be used in such situation. In this simulation, the photosensor response is the same as a linear one with the calculation presented in equation (6.2) at the output. The value I_{mean} is an averaging of all the pixel outputs, low-pass filtered with a cut-off frequency of $0.5Hz$.

The localization of the bar is quite well estimated thanks to the NWS method developed in this thesis, once the offset is removed. Without any offset removed from the photosensors signal, the output does not vary according to the bar's angular position. It indicates only which central photosensors is used to see the bar. Two possiblities are provided and only real experiments could help to choose one over the other. The first one is only subtracting a constant value known thanks to calibration, here (1.81). The second one is to removed the mean value of all the photosensor output after the ADC. The NWS Output shows a little discontinuity between each triplet, but it seems to not depend on the subtended angle unlike the NWS ouptut with a constant offset. One limit here to mention is that it might work only for a constant ambient lighting. During the variation of lighting, I do not know if the output would be reliable.

The sum of the 3 selected photosensors gives also an information about the size of the target, but as it is not normalized according to the contrast, it could only give a relative measurement and not absolute. An interesting feature of this parameter is also that the variation does not depend on the position for the subtended angle values.

6.2.5 Toward 2D target localization

A - Localization of a spherical target with a 1D Vibration

By having a 2D array of photosensors, it would be possible to detect different subtended angle for each line and then to be able to detect the sphere. Theoretically, it seems feasible, but there is a lower limit in the subtended angle that can be measured (should be greater than $2\Delta\varphi$).

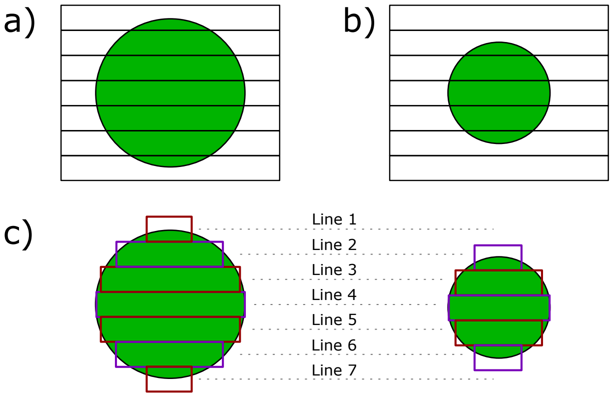


Figure 6.5: Two size of sphere are seen in the sensor FOV, a) the bigger and b) the smaller. The rectangles are representing the FOV of lines in a sensor like CurvACE. c) Representation of both sphere with a possible identification of bars for each lines

Depending if the sphere goes toward or away of the sensor, the number of line that sees the target will change. For the lines, that sees the top and the bottom of the sphere, the signal would be of less amplitude because the background would be also seen at every position. The difference between the top and the bottom would help also as a symmetry indicator in height, if the background is uniform or at a sufficient distance compared to the eye-to-target one. This possibility implies a lot of calculation for maybe a small measuring range. For example for CurvACE sensor, the size of the target should be big to be seen several lines of photosensors. At a distance of $1m$, the target should have a diameter larger than $30cm$ to be in the FOV of 4 lines.

B - Localization of different target with 2D Vibrations

A circular scanning with one actuator A first kind of a 2D vibration could be a rotation of the eye like described in figure 6.6. The eye is mounted on a gimball assembly and can rotate along 2 axes. It is a kind of miniature pantilt platform with a limited range in the rotation (It can do pitch and yaw but no roll movement). The actuator is a motor placed behind the eye. An eccentric shaft ended by a spherical joint implement the connection between the motor and the frame of the eye, where a spherical hole is made. The sphere described a circular trajectories and therefore, the eye is rotating and the visual axis is moving on a virtual conic surface. It means that the eye could never point straight ahead (see figure 6.7).

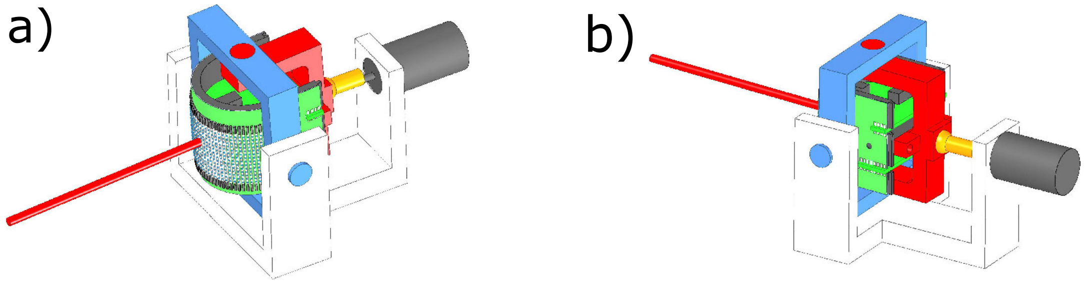


Figure 6.6: a and b) are depicted a system to enable a vibration in 2 dimensions. The sensor is mounted on a kind of a gimball assembly, but the only actuator is a motor mounted on the static frame. An eccentric shaft with a spherical joint allows the connection between the motor and the support of the sensor. The orientation axis of the sensor, shown as a red bar, is moving along a conic surface.

This setup would be very interesting in the case of circular target. If the target is aligned with the eye, with the rotational scanning, no change would appear on the pixel at the center of the rotation, if we have seven photosensors with a hexagonal shape for example (see figure 6.7b). In the same situation but with a slightly misaligned target, a variation would appear in the same photosensor response and the phase compared to the phase of the neighboring photosensor would be enough to detect the orientation of the target. This principle could be useful to detect with a high sensitiviy a mechanical misalignment between a circle placed in front of the sensor.

Another squared layout of the photosensors could also be used to locate this same target (see figure 6.7c). In the aligned situation, the amplitude of the 4 photosensors should have the same amplitude and would have a phase shifted of $\pi/2$ between each orthogonal pixel and π for the pixel located in diagonal. The variation of amplitude and shift would be sufficient to locate the target.

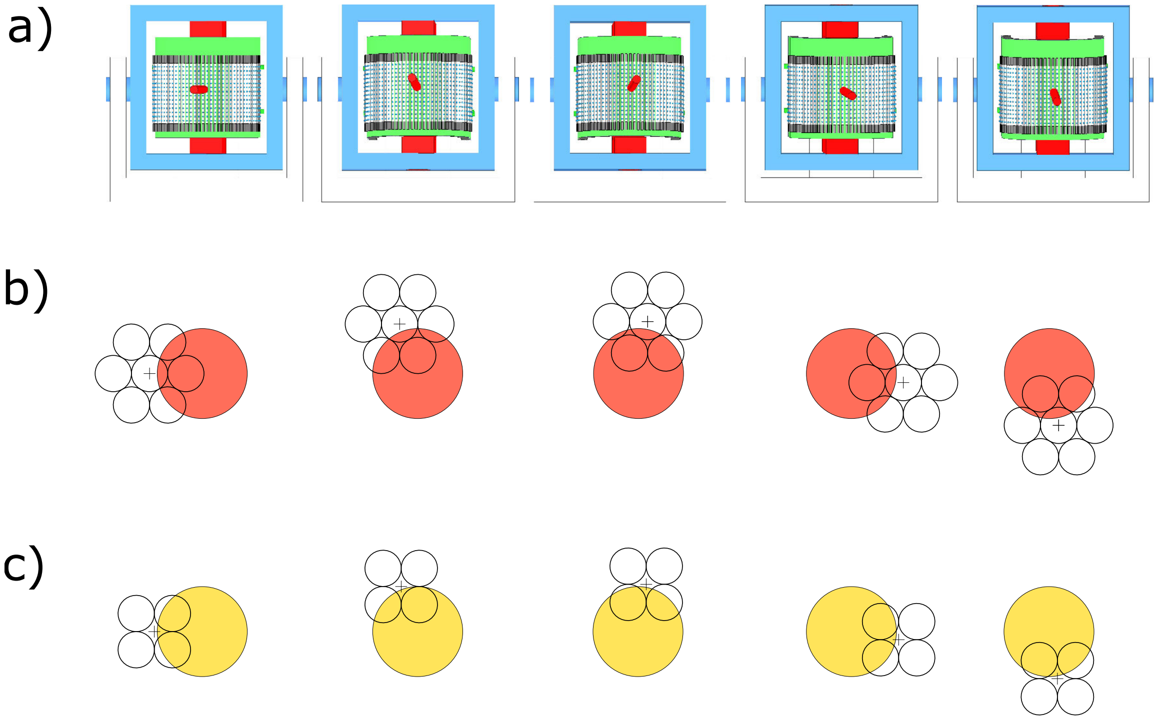


Figure 6.7: a) 5 positions of the previous 2D scanning system. The axis of vision of the sensor describes a circle. b) What see a 2D-array of 7 photosensors with a hexagonal layout for each position. c) In the case of a squared layout of 6 photosensors.

This setup could maybe make it possible to localize other shape of target, like squared or rectangular ones.

In all these situations, the signal processing needs an appropriate design but there are not far from the works done on demodulation and the edge/bar detector.

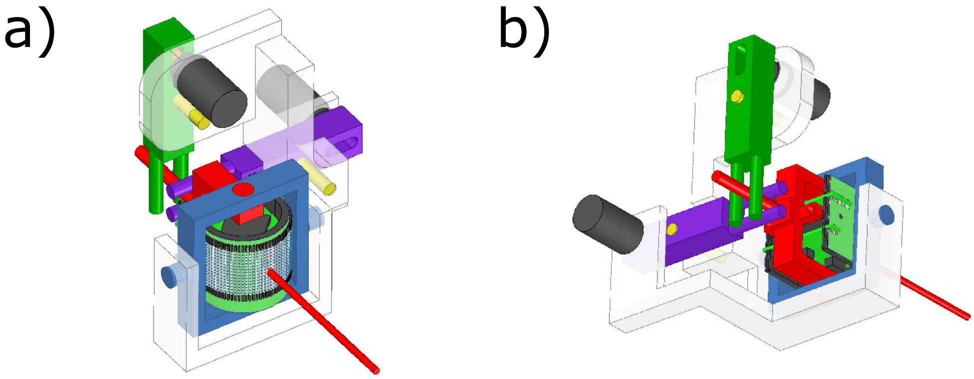


Figure 6.8: 2D scanning with 2 actuators. Both are using an eccentric mechanism to create an alternative rotational movements on 2 dimensions. The phase and the frequency of the two motor can be either identical or different to modulate the orientation of the eye in different manners. a) Front three-quarter view b) Back three-quarter view. In both views, the red bar that goes out of the sensor is the symbolize the visual direction of the sensor

Two actuators for different scanning possibilities A second kind of setup could be to have 2 actuators which would be closer to a bio-inspired solution. Indeed, the insect are endowed with 2 muscles (MOT and MOS) to create retinal movement 1.2.4. To make a visual sensor moves, different systems have already been developed like the gimbal system with 2 serial rotations. But they are presenting the inconvenient property to not have the same dynamics along all the axis of rotation. Another system that was designed to realize a target tracking [Manecy et al., 2016], fits this requirement. It could be inspiring for the 2D vibration objectives, but should be miniaturized to increase the rotational speed needed for scanning system. Another example of a bio-inpired eye was using three actuators, to reproduce the human eye dynamics and control [Liu et al., 2015].

An attempt to implement a 2D scanning sensor is to reproduce the same 1D solution in the orthogonal direction, as shown in figure 6.8. The advantage is to have multiple possibilities of scanning compare to the circular scanning presented for the tracking of a circular target (see figure 6.6). By activating one motor and turning off the vibration of the second one, it would provide the scanning in one direction. It could be the same for the orthogonal direction. The drawbacks of this solution is that you really need the position of the motor of the steady axis to know in which direction you are actually vibrating. By having the two motor scanning at the same frequency, it is possible to replicate the rotational movement previously. Another possibility would be to have one motor turning at a lower speed than the other and the scanning would result in a zigzag trajectory to scan all the visual scene. A specific peak filter on the motor frequency could be enough to discriminate one dimension from the other and then localize a rectangular target for example.

This design would be very interesting from a research point of view as it would enable to explore the 2D active vibration with different possible strategies.

C - Using color vision in target tracking

It is known that bees use color and various spectral sensitivities to locate flowers. Maybe, the use of the color to enhance a contrast is useful in target tracking with low spatial resolution.

In the research of a target or even for stabilization, which where the application developed during this thesis. The color signals were not used for these tasks. It is known that there are 7 to 8 receptors in the fly eyes 1.2.2. During our work, a limited range of the spectral sensitivy was used, mostly the red one, but other sensitivity should be address.

The idea behind is for example to use the green signal, as the environment is mostly green and the target is mostly black or brown. UV sensitivity might also be useful to discriminate object of interest in a cluttered environment. Recognize the color could help in having a more robust target tracking as the target would not be described only by the position of some edges.

6.2.6 Other possible experimental situations

A - Docking

As it was demonstrated, thanks to the localization of 2 edges, a robot is able to position itself with respect to a cylinder. The same principle can be extended to a rectangular pattern on a wall. Figure 6.9b presents the situation where the robot come close to a docking area, then it can

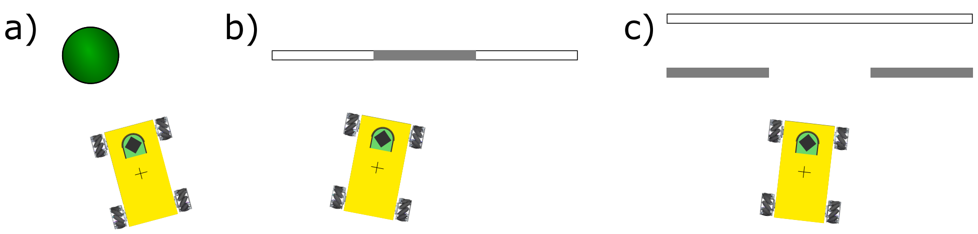


Figure 6.9: Setups that can be realized thanks to the localization of 2 edges a) First possibility explained in chapter 4. It is the cylinder following b) Docking in front of a rectangular pattern c) Crossing an aperture, like a door for example.

use its localization algorithm to position itself with precision in the last phase of the docking. This scenario could be interesting for an autonomous system which has to return to a recharge station. In this situation, the system has to detect the approximate position of the two edges to initialize the localization processing. This difficulty can be overcome if the wall contain only the rectangular shape, as the robot moves in the approaching phase, the motion detector can localize the edges.

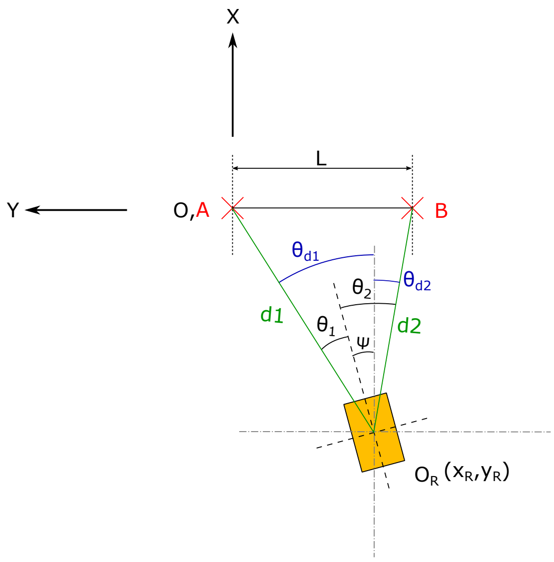


Figure 6.10: A robot approching two points marked A and B separated by a distance L. O_R is the position of the robot, ψ its orientation in the absolute frame.

A simple control ensuring $\theta_1 = \theta_2$ would ensure to end up between the two identified points. The heading of the robot is always pointing in between the 2 points. With the notation given in figure 6.10, the inequality becomes equivalent to:

$$\theta_1 = \theta_2 \quad \equiv \quad \theta_{d1} - \psi = \theta_{d2} + \psi \tag{6.3}$$

knowing that:

$$\begin{cases} \tan(\theta_{d1}) = & \dfrac{-x_R}{-Y_R} \\ \tan(\theta_{d2}) = & \dfrac{-x_R}{L + Y_R} \end{cases} \tag{6.4}$$

Therefore, the equation (6.3) is true when the following is met:

$$\frac{-x_R}{-Y_R} = \frac{-x_R}{L + Y_R} + 2\psi \tag{6.5}$$

This equality is true for $y_R = \frac{-L}{2}; x_R \in \mathbb{R}$ in the case where $\psi = 0$.

Thus, a strategy which maintains only $\theta_1 = \theta_2$ does ensure to finish in the middle of the marker only if the orientation of the robot is perpendicular to the line passing by the 2 points A and B. However, to have the value ψ from only the angular measurements is not possible instantaneously, even with the knowledge of L. Another information is needed, it could be a third beacon or a magnetometer for example.

This task is perfectly known in current state-of-art with any pattern, [Crombez et al., 2015] for example. However, in computer vision, it requires quite extensive computation resources. The solution proposed here, would be simplest and would therefore requires lower computation resources.

B - Navigation through doors (or gap or aperture)

When it comes to the task of crossing a gap, the situation can be seen as 2 edges on each side that have a contrast with the background, as in figure 6.9c. This scenario is very similar to the docking problem and could have the same resolution.

Salaris *et al.* [Salaris et al., 2015] implemented a visual servoing to go through a door presents the difficulty and the best way to achieve it. The control law presented ensure to cross the door axis perpendicularly and in the middle, in order to avoid any contact with the wall. Then, the control law should ensure to go through a gap with the best position to cross the gap safely, *i.e.* without touching the obstacle. The visual cues are the angular position and the height in the image plane of the 2 landmarks. The temporal evolution of the subtended angle, which derived from the angular position, is also used. Salaris *et al.* provided also a comparison of its control law to the human behavior facing the same situation based on previous experiments [Arechavaleta et al., 2008], which shows very similar results.

The difference compared to the target pursuit is the detection of the edges of the doors to initialize the localization processing. A possible solution could be to do a rotation back and forth. For sure, it will be possible to localize edges, but the question "how to detect the contrasts formed by the gap to the others if multiple contrasts are detected?" remains not trivial.

6.3 The bio-inspired approach

What we did during this thesis is to use a bio-inspired visual sensor to perform different tasks. Of course the performances achieved are not always comparable to the computer vision ones. But regarding other metrics as the computational load and the lightness of the bio-inspired solution can also be incomparable as well. The technology is increasing which reduces more and more the today's differences in terms of compactness and payload. But the efficiency is more often in favor of the bio-inspired solution. Indeed, as solving a complex problem is always a matter of trade-offs, nature has optimized its living beings to be the best adapted to their ecosystems.

Being able to design bio-inspired systems which are achieving similar performance to their biological counterparts is also a proof that nature has been well understood. If not, some assumptions can be made and provides another hypothesis for the biologists.

The solution can even contain some uncertainty, as realizing different tasks perfectly at each iteration imply hard constraints on the design which do not cope with power efficiency or small time response. In an open world in constant evolution, make some mistakes and adapt its behavior for the next try seems to be the absolute solution to every problems.

Appendices

A Lexical terms

A.1 Optic Flow

Optic flow can be defined as the motion of different objects or contrasts perceived by an observer. It is characterized by an amplitude which is the speed perceived and the orientation which provide the direction of the motion.

In a steady environment, if the observer is only rotating, the optic flow perceived will only depend on the speed of motion and the axis of rotation. However, for a translational movement, the optic flow will depend on the the speed and the distance of the features viewed. The optic flow field are shown for both cases in figure A.1. It is highlighted that close to the rotational axis, the perceived motion is small whereas is maximum on the orthogonal directions. For the translational movements, it is highlighted that the optic flow is null in the direction of motion.

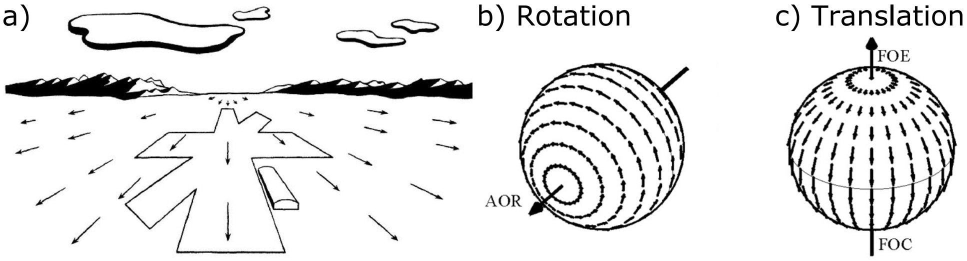


Figure A.1: a) Schematic representation of the optic flow vector field generated on the retina of an agent during a landing phase. Each vector magnitude and direction represents the angular speed of this point of the environment. The point right in front of the agent where the angular speed is zero is called the focus of expansion. Reprinted from [Gibson, 1951].
Optic flow induced during a rotation about a horizontal axis (b) or an upward translation along the vertical body axis (c). Reprinted from [Karmeier et al., 2003].

Koendrick and van Doorn described the optic flow mathematically [Koenderink and van Doorn, 1987]. Their model gives the optic flow output in the case of a moving observer and a fixed object at a position P in the observer's frame. The motion of the observer is decomposed into a translation $\overrightarrow{T}$ and a rotation $\overrightarrow{\Omega}$ relative to an inertial frame. The optic flow ω is:

$$\omega = \frac{\left((\overrightarrow{T} \cdot \hat{p})\hat{p} - \overrightarrow{T}\right)}{\|p\|} - \overrightarrow{\Omega} \times \hat{p} \tag{6}$$

where p is the position vector of P, $\hat{p} = \frac{p}{\|p\|}$ is the unit vector describing the direction of p. It can be noticed that $\|p\|$ represents the distance to the object.

A.2 ON and OFF Contrasts

The definition of an ON contrast is the increasing of the light received by a photoreceptor, meaning a dark surface is seen before a brighter one. An OFF contrast is the opposite as it is the transition from a bright to a darker scene. The two kind of contrasts are depicted in figure A.2.

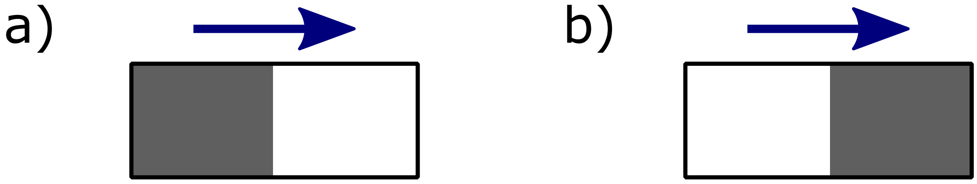


Figure A.2: The blue arrow indicates the sens of motion meaning that in a) an ON contrast is shown whereas in b) it is an OFF contrast.

A.3 Edge and bar

An edge is defined as the transition of two area with different color making a contrast. A bar is the succession of 2 edges in a small part of the field of view. It could also be called a stripe.

Figure A.3: Presentation of 2 field of view seeing (a) an edge and (b) a bar.

B Additional information on chapter 2

B.1 Experimental results of HyperRob attitude disturbance

In this part, we present a study of the influences of attitude disturbances on the position measurement and how it is handled by the robot.

Pure attitude disturbances This experiment has for purpose to establish the efficiency of the gaze stabilization and to demonstrate that the position measurement is insensitive to roll disturbances and roll movements.

To see the influence of roll disturbances on the vision output, perturbations were applied by hand on the left wing, from bottom to up. It should be highlighted that it is the worst attitude perturbation we can meet in real flight condition. Indeed, it is a discontinuous perturbation unlike a wind perturbation. During this test, the position controller was turned off and the robot position was locked with smooth stops to avoid robot's lateral displacement. The results presented on the figure B.4 show that the roll disturbances have a limited influence on the vision output. As the amplitude of the disturbance is about 10°, the corresponding visual error should be around $70mm$ for a planar pattern located at a distance of $400mm$ without any gaze stabilization. However, the visual error shows a peak of around $15mm$ at each disturbance to finally come back to its initial measurement with an accuracy of $\pm 5mm$. So, the effect of the decoupled eye is clearly seen with those results which prove the efficiency of the VOR.

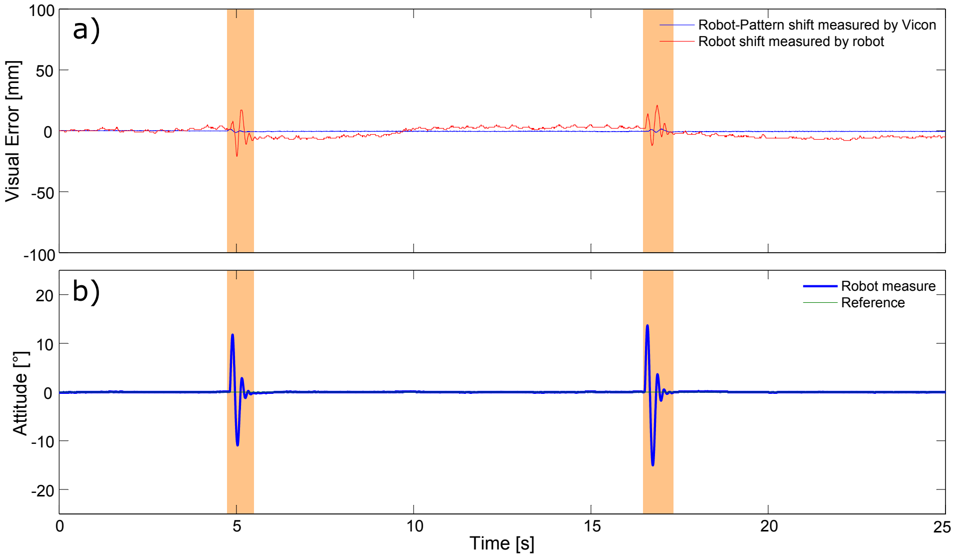


Figure B.4: Roll disturbance rejection with fixed position above a naturally textured pattern. a) The visual error of the robot is displayed in red. Without any disturbances the robot is able to measure its position with a 3-mm noise level. A disturbance implies a transient position error of $\pm 20mm$ due to parasite rotations of the eye because the roll compensation is not perfect. But the position after the disturbances come back to the initial measurement with a 5-mm accuracy. b) Attitude of the robot during disturbances, the disturbances are rejected in less than 1 sec.

Attitude disturbances in free flight As a second test, the smooth stops were removed and the position controller was activated to study the influence of roll disturbances on the hovering performances. The attitude perturbations could be seen on the figure B.5c at time $11.5s$, $12.5s$, $14.5s$ and $18s$. As it can be seen, the roll disturbances imply lateral displacements which are quite well measured by the sensor. As a results, even after 4 roll disturbances of different amplitudes, the induced lateral displacements are rejected in figure B.5a in less than 5 seconds and the reference position is reached with a 2-mm accuracy.

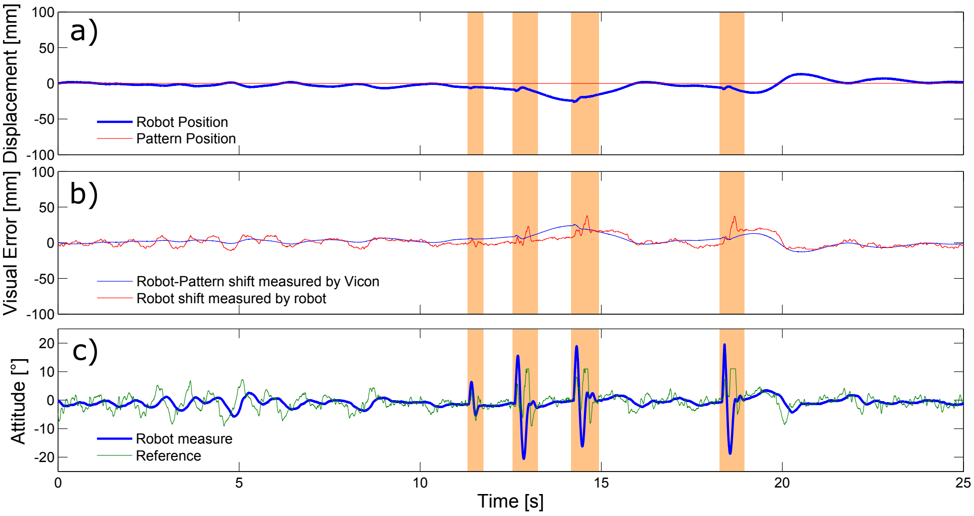


Figure B.5: Roll disturbance rejection with arm free above a naturally textured pattern. a) The pattern position and robot position measured by the Vicon system. We can see that the disturbances are rejected in less than 5 seconds and that any offset are induced by this disturbances. b) Comparison of the robot-pattern shift measured by the Vicon and the robot. As we can see, the robot measure accurately its position even if the accuracy is decreased during transient in attitude. c) Attitude of the robot during the disturbances and the attitude set point yielded by the position controller.

B.2 Zoom on LPU output with translational movement

The non-linear and even non-monotonous measurements depicted in the circled areas in figure 2.22 should be avoided in a localization algorithm. As the strip was constantly moving at the same speed, the output should be linear with a constant slope, which is not the case. It is highlighted by the figure B.6 which depicts this in details. Taking the third pair as an example, between the time $t = 38.5s$ and $t = 41.2s$ (the largest time where the pixel signals are in phase meaning an edge should be seen), θ_i is not monotonous and there is a "bump" at the end, which is interpreted as a back and forth movement. The use of such algorithm on a target tracking would be difficult, because a small number of pair would see the target at the same time.

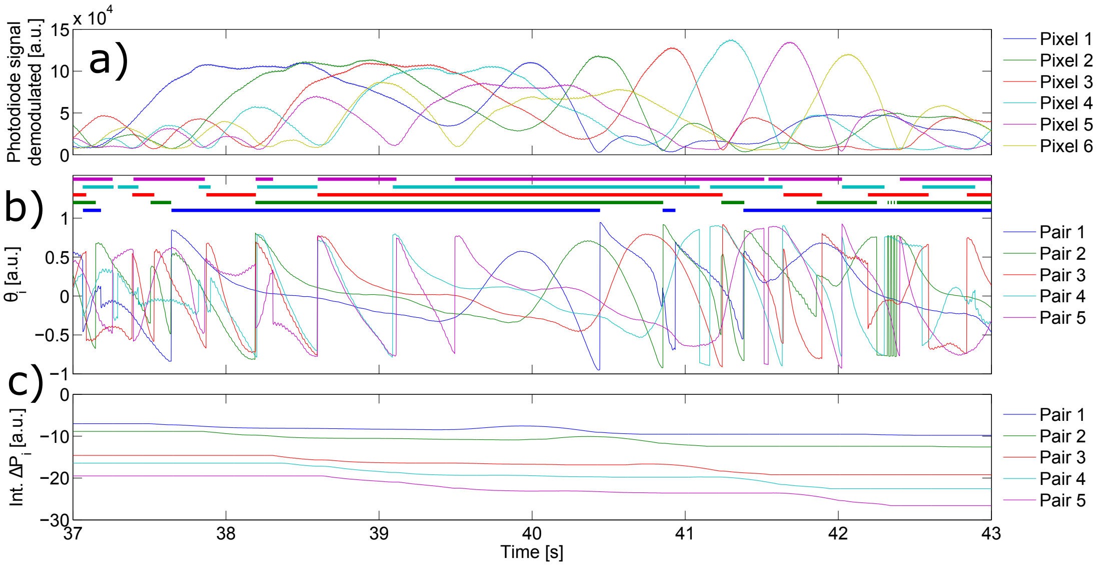


Figure B.6: Data from the experiments between the times $t = 37s$ and $t = 43s$ a) Demodulated signals of the 6 pixels b) The output of the 5 Local Processing Unit θ_i. On top, the colored line are drawn when the pixel signals of the respective pair are in phase c) Integration of each ΔP_i with threshold equal to 80000 (Zoom of the figure 2.22b)

C Additional information on chapter 3

C.1 Adding information on the tracking experiments

As it seems in the figure 3.10 that the subtended angle measurement error is lower when the tracking error is small (meaning that the angular position of the bar is well estimated). The angular position and the subtended angle measurement were plotted according to the theoretical values in figure C.7. It reveals that by removing the point where the tracking error is above 0.5°, the largest subtended angle measurement error are mostly removed as well. It appears however that the angular position of the bar selected are only in the range $[-10, 15]$, which means that the error are larger one one side. The theory of a potential shadow in that position is therefore plausible. It is also interesting to note that the α measurement are mostly overestimated which suggests that this error is not linked with a delay in the processing.

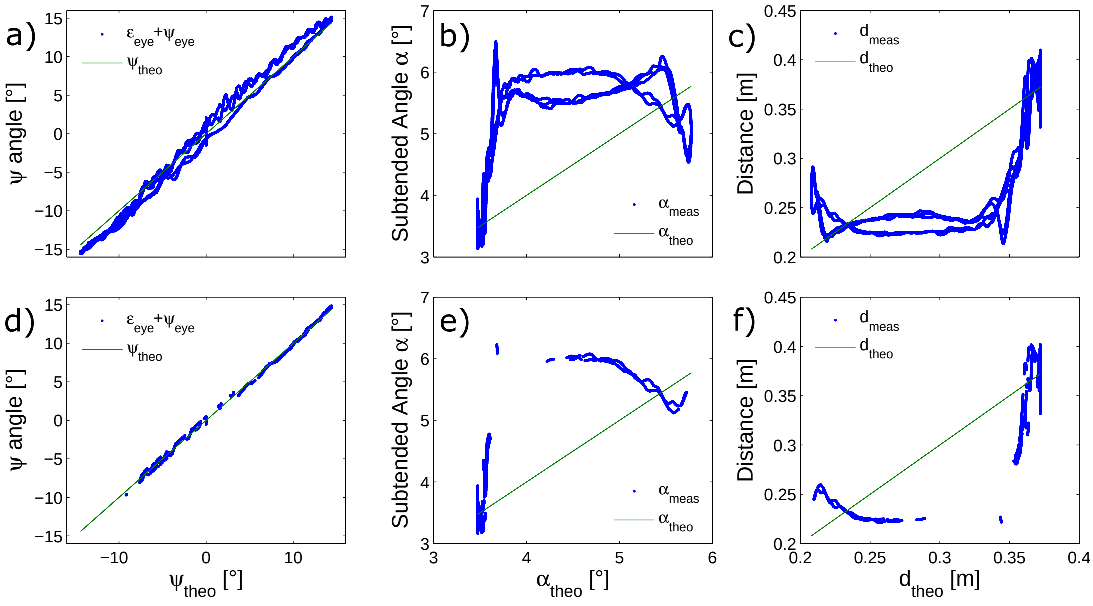


Figure C.7: a-c) Measurements of the tracking experiment, plotted versus the theoretical ones. a) is the orientation, b) the subtended angle and c) the distance. d-f) are the same data but without the values out of a range from -0.5 to 0.5° of the angular position error. It shows that big errors still happen at different positions of the target and some of the subtended angular measurement outliers are avoided.

C.2 Effect of an asymmetry in the contrast of a bar

A bar is defined as the succession of 2 contrasts with in a small spatial angle. In the previous studies, the two contrasts were always considered equivalent. Figure C.8 shows that the output of the difference over the sum of two demodulated pixel signals is not symmetric in the case where the left and the right contrasts are different. The zero-crossing position does not occur at the same position as well.

Considering such evolution in a map would be very difficult. Indeed, it adds another variable to the angular position and the subtended angle. The model of a bar is maybe not the best to

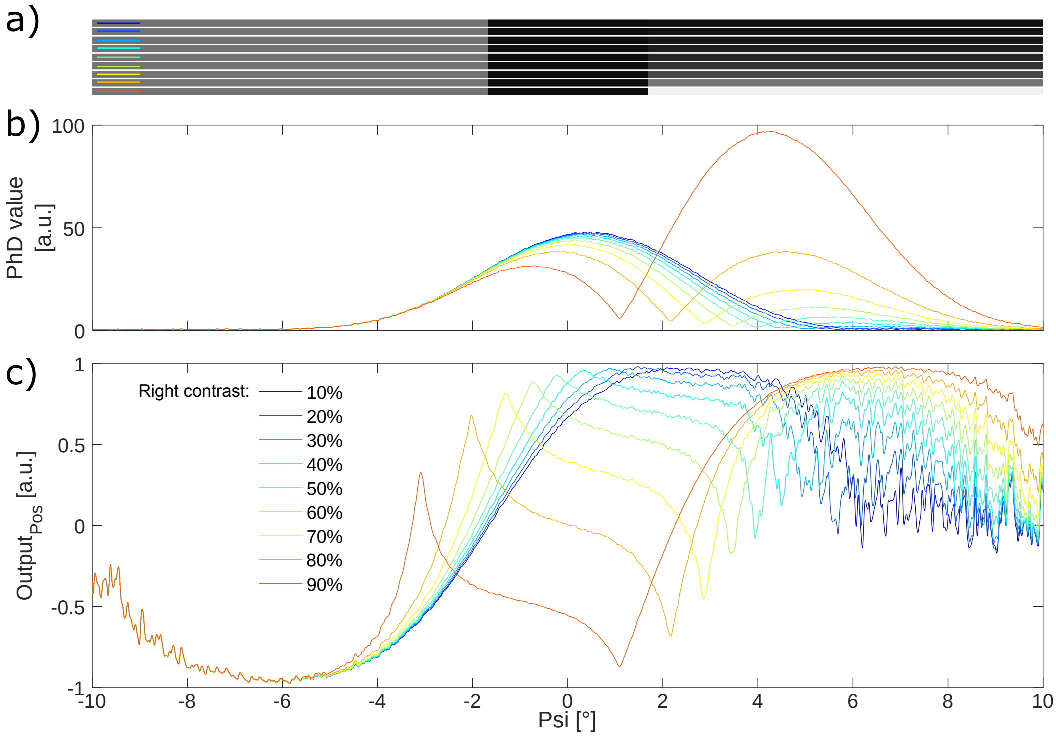


Figure C.8: Simulation of a pair of photosensors rotating in front of a bar subtending an angle of 3.37°. The optical parameters are $\Delta\varphi = \Delta\rho = 4.2°$. The contrast on left is constant and equal 80% whereas the contrasts is varying on the right from 10 to 90% (according to Michelson's definition). a) Simulated contrasts color in front of different background from 10 to 90%. The bar is centered at the 0° position. b) Responses of the second photosensor, which is oriented at $\frac{\Delta\varphi}{2} = 2.1°$. c) Simulated response of the difference over the sum of the two photosensors according to the rotational angle of the eye.

evaluate the succession of multiple contrasts.

D Additional information on chapter 4

D.1 Effect of the vibration amplitude on the Normalized Weighted Sum

Form the variation of amplitude simulated in the section 4.3.2, it can be seen that for small amplitudes of scanning, the maximum of the Gaussian is small. The larger is the scanning amplitude, the higher is the maximum of the Gaussian. But at a value higher than $\Delta\rho/2$, the output is less close to a Gaussian function.

These variations have an impact in the Normalized Weigthed Sum processing. Figure D.9 shows the output of the NWS calculation with different scanning amplitudes. It reveals that for small ones, the output is too noisy to really have a good estimation of the edge position. For larger amplitudes, the output is closer to a staircase than a line. The best compromise is found for a value between 2.1 and 4.2°, which means a value between $\frac{\Delta\rho}{2}$ and $\Delta\rho$.

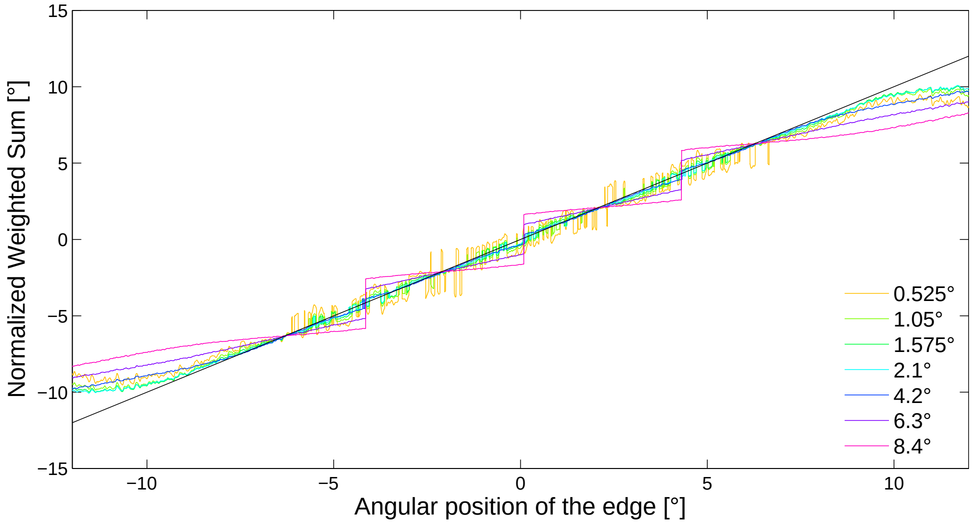


Figure D.9: Effect of the amplitude of the vibration on the output of the NWS processing. The simulation is done using 6 photosensors spaced by an angle of 4.2° and an acceptance angle $\Delta\rho$ also equal to 4.2°. In black is plotted the theoretical response. For small amplitude, the signal-to-noise ratio is too bad and for large amplitude, the input signal is too far from a Gaussian to give a linear response.

D.2 Details about ACEbot hardware

Figure D.10 sums up all the wirings of ACEbot and the communication between the different electronic boards.

D.3 Detailed presentation of the ACEbot control

D.3.4 - Introduction

The omnidirectional mobile robot is a system with three degrees of freedom. This characteristic allows a movement in any directions with any orientations. This is a main advantage compared to mobile robots with traditional wheels (which require a rotational movement before doing a

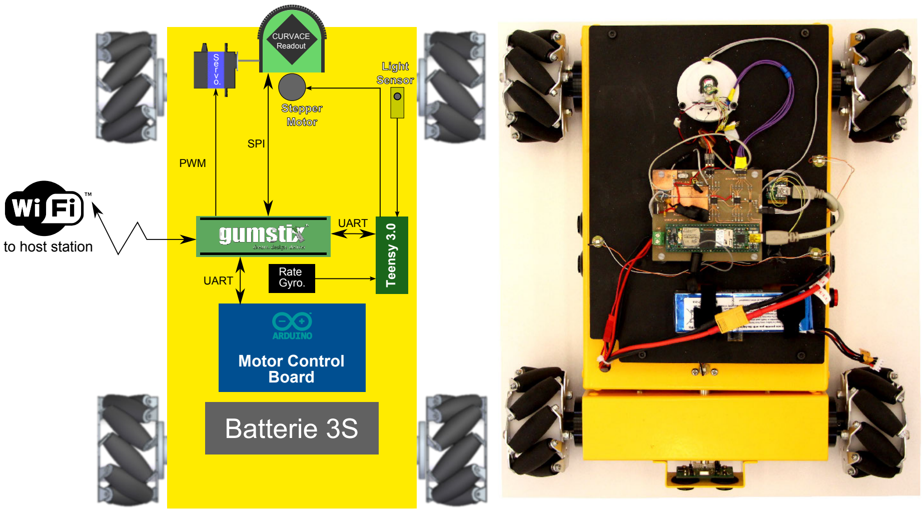


Figure D.10: Electronic architecture of the robot

translational movement).

The Nexus omnidirectional mobile platform is equipped with Mecanum wheels which makes him holonomous.

The control algorithms implemented in mobile robots are generally designed using the kinematic or dynamic model. The application of kinematic model in different control laws is very useful due to its structure that does not depend on dynamic changes in the mobile robot parameters (such as mechanics imperfections or wearing damages [Campion et al., 1996]). This is the main reason why the kinematic model will be used in this work.

D.3.4 - Kinematic model

Presentation The Mecanum wheel mobile platform with mecanum wheels moves in any direction and turn by varying the direction and speed of each wheel. Moving the four wheels in the same direction causes forward/backward movement, running left/right sides in opposite directions causes rotation and running front/rear in opposite directions causes side translation (view Fig. D.11).

Nexus mecanum robot specifications are presented in Table 1.

Table 1: Parameters of Nexus mecanum robot.

Parameters	Description	Value	Units
R_w	Radius of wheel	0.050	m
L	Half length L in axis X	0.15	m
l	Half width l in axis Y	0.15	m
I_w	Wheel's moment of inertia	5 e-4	$kg \cdot m^2$
I_z	Robot's moment of inertia	0.0402	$kg \cdot m^2$
m	Total mass of the robot	4.2	kg
m_w	Mass of the wheel	0.4	kg
D_θ	Viscosity friction coefficient of the wheel		

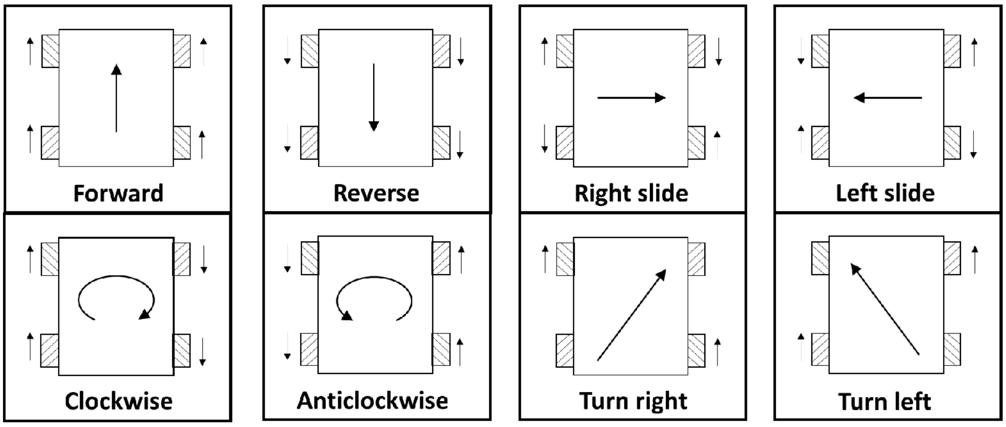


Figure D.11: Co-effect of 4 Mecanum wheels.

To formulate the kinematic model, which is the study of the movements of the mobile robot according to its geometry, it will be assume that the mobile robot is a rigid frame equipped with nondeformable wheels and that it is moving on a horizontal plane (Campion, 1996). The representation of the robot is showed in Fig.D.12.

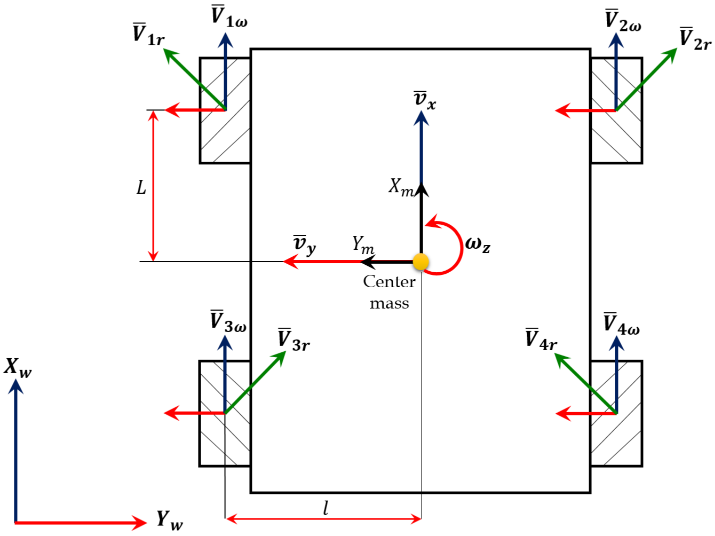


Figure D.12: Diagram of Nexus Mecanum Robot in the plane XY.

The kinematics equation reflects the mapping relationship between the speeds of the four Mecanum wheels and the center speed of the moving mechanism. The qualities of the Jacobian matrix directly reflect the natures and characteristics of the moving body. If the mobile bodyâĂŹs Jacobian matrix of the kinematics equation is not fully ranked, then the moving mechanism will have singularity, and this means the kinematic agencies will lose some degrees of freedom, so that the moving mechanism cannot achieve a full-directional movement.

Each individual wheel contributes to the robot motion and imposes constraints on the robot motion.

For the presentation of the kinematic model, several notations will be used for the robot and wheels parameters, which are indicated in Table 2.

Table 2: Notations for the wheels and the robot.

Notation	Description
$V_{iw}(i = 1,2,3,4)$	Velocity corresponding to wheel rotation.
$V_{ir}(i = 1,2,3,4)$	Tangential velocity of the free roller touching the locomotion surface.
R_w	Radius of omnidirectional wheel.
ω_{iw}	Angular velocity of the omnidirectional wheel.
v_x	Linear speed component of the omnidirectional robot on X.
v_y	Linear speed component of the omnidirectional robot on Y.
ω_z	Angular velocity of the omnidirectional robot.
L	Distance from each wheel shaft to the mass center of the robot on the X axis.
l	Distance from each wheel shaft to the mass center of the robot on the axis Y.

Inverse kinematic model As described in [Tsai et al., 2011], [Tatar et al., 2014] and [Viboonchaicheep et al., 2003], the inverted kinematic model can be expressed thanks to figure D.11. Indeed, the wheels have a rotational speed ω_{iw} which lead to the velocity vector $V_{iw} = R_w.\omega_{iw}$. V_{ir} expressed the tangential rolling speed of the roller touching the ground. Therefore, the speed vectors of the wheels in the global frame can be described as follows:

$$\begin{aligned}
V_{1X} &= V_{1w} + V_{1r} \cdot cos\left(\frac{\pi}{2}\right) \quad , \quad V_{1Y} = V_{1r} \cdot sin\left(\frac{\pi}{2}\right) \\
V_{2X} &= V_{2w} + V_{2r} \cdot cos\left(\frac{\pi}{2}\right) \quad , \quad V_{2Y} = -V_{2r} \cdot sin\left(\frac{\pi}{2}\right) \\
V_{3X} &= V_{3w} + V_{3r} \cdot cos\left(\frac{\pi}{2}\right) \quad , \quad V_{3Y} = -V_{3r} \cdot sin\left(\frac{\pi}{2}\right) \\
V_{4X} &= V_{4w} + V_{4r} \cdot cos\left(\frac{\pi}{2}\right) \quad , \quad V_{4Y} = V_{4r} \cdot sin\left(\frac{\pi}{2}\right)
\end{aligned} \tag{7}$$

Moreover, the movement of the robot described by the translational velocities v_x, v_y and the rotational velocity ω_z expressed at the wheel position leads to the following equation:

$$\begin{aligned}
V_{1X} &= v_x - l \cdot \omega_z \quad , \quad V_{1Y} = v_y + L \cdot \omega_z \\
V_{2X} &= v_x + l \cdot \omega_z \quad , \quad V_{2Y} = v_y + L \cdot \omega_z \\
V_{3X} &= v_x - l \cdot \omega_z \quad , \quad V_{3Y} = v_y - L \cdot \omega_z \\
V_{4X} &= v_x + l \cdot \omega_z \quad , \quad V_{4Y} = v_y - L \cdot \omega_z
\end{aligned} \tag{8}$$

Substituting (8) in (7) gives:

$$\begin{aligned}
V_{1w} &= v_x - v_y - (L + l) \cdot \omega_z \\
V_{2w} &= v_x + v_y + (L + l) \cdot \omega_z \\
V_{3w} &= v_x + v_y - (L + l) \cdot \omega_z \\
V_{4w} &= v_x - v_y + (L + l) \cdot \omega_z
\end{aligned} \tag{9}$$

Then, expressing the equation (9) in a matrix form results as (10), which represents the inverse kinematics equation:

$$V_w = J_0 V_0 \tag{10}$$

where $V_w = [V_{1w}\ V_{2w}\ V_{3w}\ V_{4w}]^T$ is the wheel velocity vector and $V_0 = [v_x\ v_y\ \omega_z]^T$ is the velocity vector in Cartesian coordinates.

The transformation matrix J_0 is equal to:

$$J_0 = \begin{bmatrix} 1 & -1 & -(l+L) \\ 1 & 1 & (l+L) \\ 1 & 1 & -(l+L) \\ 1 & -1 & (l+L) \end{bmatrix} \tag{11}$$

Therefore, from the relation (10), the relation of the wheels angular velocities expressed as a function of the robot velocities is the following:

$$\begin{bmatrix} \omega_{1w} \\ \omega_{2w} \\ \omega_{3w} \\ \omega_{4w} \end{bmatrix} = \frac{1}{R_w} \begin{bmatrix} 1 & -1 & -(l+L) \\ 1 & 1 & (l+L) \\ 1 & 1 & -(l+L) \\ 1 & -1 & (l+L) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} \tag{12}$$

Forward kinematic model In order to express the robot velocity according to the wheel speed, the inverse matrix of J_0 should be determine. However, J_0 is not square, so the pseudo inverse matrix J_0^+ is used instead:

$$J_0^+ = (J_0^T J)^{-1} J_0^T = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -\frac{1}{L+l} & \frac{1}{L+l} & -\frac{1}{L+l} & \frac{1}{L+l} \end{bmatrix} \tag{13}$$

As the relation $J_0^+ \cdot J_0 = I_3$ is verified, J_0^+ can be considered as the inverted model.

So, the forward kinematics equations could be expressed as follows:

$$V_0 = J_0^+ V_w \tag{14}$$

or expressed in details, with the velocities of the omnidirectional robot v_x, v_y and ω_z:

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} = J_0^+ \begin{bmatrix} V_{1w} \\ V_{2w} \\ V_{3w} \\ V_{4w} \end{bmatrix} \tag{15}$$

Considering the relation $V_{iw} = R_w \cdot \omega_{iw}$ the result can be expressed as:

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} = \frac{R_w}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -\frac{1}{L+l} & \frac{1}{L+l} & -\frac{1}{L+l} & \frac{1}{L+l} \end{bmatrix} \begin{bmatrix} \omega_{1w} \\ \omega_{2w} \\ \omega_{3w} \\ \omega_{4w} \end{bmatrix} \tag{16}$$

or as well:

$$\begin{array}{lcl} v_x & = & \frac{R_w}{4}(\dot{\theta}_{1w} + \dot{\theta}_{2w} + \dot{\theta}_{3w} + \dot{\theta}_{4w}) \\ v_y & = & \frac{R_w}{4}(-\dot{\theta}_{1w} + \dot{\theta}_{2w} + \dot{\theta}_{3w} - \dot{\theta}_{4w}) \\ \omega_z & = & \frac{R_w}{4(\mathrm{L+l})}(-\dot{\theta}_{1w} + \dot{\theta}_{2w} - \dot{\theta}_{3w} + \dot{\theta}_{4w}) \end{array} \tag{17}$$

Transformation from Robot frame to Global frame The forward and inverse kinematic models of the robot are formulated in a local frame, *i.e* the robot frame. The position and orientation of the vehicle in the global frame (in the world) is denoted by the vector $[x_w \; y_w \; \psi]^T$. The transformation matrix between the local frame and the global frame is expressed by:

$$\begin{bmatrix} \dot{x}_w \\ \dot{y}_w \\ \dot{\psi}_w \end{bmatrix} = \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} \tag{18}$$

substituting (13) and (18) gives the forward kinematic model:

$$\begin{bmatrix} \dot{x}_w \\ \dot{y}_w \\ \dot{\psi}_w \end{bmatrix} = J^+(\psi)V_w \tag{19}$$

where the transformed matrix $J^+(\psi)$ is defined by:

$$J^+(\psi) = \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot J_0^+ \tag{20}$$

which can be simplified in:

$$J^+(\psi) = \frac{1}{4} \begin{bmatrix} \sqrt{2}sin\psi_1 & \sqrt{2}cos\psi_1 & \sqrt{2}cos\psi_1 & \sqrt{2}sin\psi_1 \\ -\sqrt{2}cos\psi_1 & \sqrt{2}sin\psi_1 & \sqrt{2}sin\psi_1 & -\sqrt{2}cos\psi_1 \\ -\frac{1}{L+l} & \frac{1}{L+l} & -\frac{1}{L+l} & \frac{1}{L+l} \end{bmatrix} \tag{21}$$

and $\psi_1 = \psi + \pi/4$.

Similarly, since the transformation $J(\psi)$ exists its the pseudo inverse matrix $J^+(\psi)$, the inverse kinematics model of the mobile robotics is expressed as follows:

$$V_w = J(\psi) \begin{bmatrix} \dot{x}_w \\ \dot{y}_w \\ \dot{\psi}_w \end{bmatrix} \tag{22}$$

where

$$J(\psi) = \begin{bmatrix} \sqrt{2}sin\psi_1 & -\sqrt{2}cos\psi_1 & -(L+l) \\ \sqrt{2}cos\psi_1 & \sqrt{2}sin\psi_1 & (L+l) \\ \sqrt{2}cos\psi_1 & \sqrt{2}sin\psi_1 & -(L+l) \\ \sqrt{2}sin\psi_1 & -\sqrt{2}cos\psi_1 & (L+l) \end{bmatrix} \tag{23}$$

and

$$J^+J = I_3 \tag{24}$$

Validation of the kinematic model In order to validate the model, an experiment in open loop was made with the robotic plateform. It consists in applying wheel speed commands and compare the model behavior and the real robot one. It should be highlighted that a PI controller was designed in order to assess the wheel speeds, thanks to 4 optical encoders which gives the rotational speed of each wheel. The robot wheel speeds were therefore perfectly controlled.

With the purpose of verifying the robot movements (forward, reverse, left, right, left slide, right slide, clockwise and anticlockwise) a sequence with this kind of movements was implemented, which lasts 120s in total. For each sequence, the speed reference of each wheel is either 1.5, -1.5 or 0 rev/s depending on the desired movement. The experimental results of the speed are represented in figures D.13. The estimated speeds are aligned with the experimental results which shows that the model and the parameters are well estimated and measured.

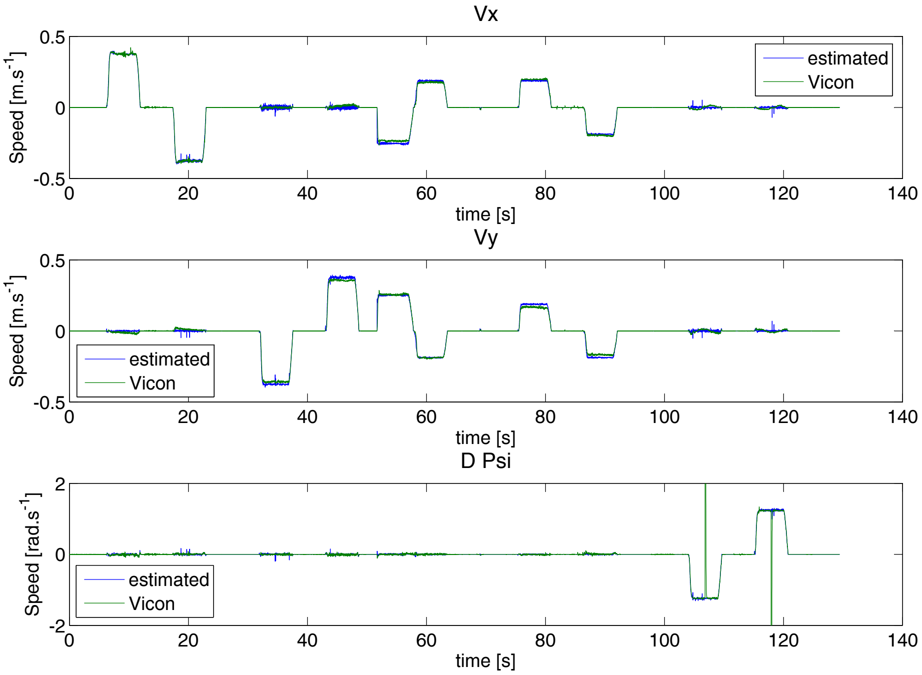


Figure D.13: Speed values of the Mecanum Nexus Robot
The blue line is the estimated value of the kinematic model and the green line is the experimental result measured with motion capture system

D.3.4 - Control law design

Design We consider the control law presented in [Guerrero-Castellanos et al., 2014] to apply at this particular case. It is a feedback linearization method with saturated control.

Let $x = [x_{11}\ x_{12}\ x_{21}\ x_{22}\ x_{31}\ x_{32}]^T = [\int x_w\ x_w\ \int y_w\ y_w\ \int \psi_w\ \psi_w]^T$ be the states vector and $u = [u_1\ u_2\ u_3]^T = \left[\dot{x}_m\ \dot{y}_m\ \dot{\psi}_m\right]^T$ is the control vector. Now the representation in states space of the system (18) in (25).

$$\begin{aligned}
\dot{x}_{11} &= x_{12} \\
\dot{x}_{12} &= u_1 cos(x_{32}) - u_2 sin(x_{32}) \\
\dot{x}_{21} &= x_{22} \\
\dot{x}_{22} &= u_1 sin(x_{32}) + u_2 cos(x_{32}) \\
\dot{x}_{31} &= x_{32} \\
\dot{x}_{32} &= u_3
\end{aligned} \tag{25}$$

With the purpose to delete the non-linear terms in the model (25), we consider the next non-linear control defined in the next step, where r_i are signals with $i = 1, 2, 3$.

$$\begin{aligned}
u_1 &= cos(x_{32}) r_1 + sin(x_{32}) r_2 \\
u_2 &= -sin(x_{32}) r_1 + cos(x_{32}) r_2 \\
u_3 &= r_3
\end{aligned} \tag{26}$$

With (25) and (26) becomes subsystems Σ_i with $i = 1, 2, 3$, where Σ_1 and Σ_2 are the translational movement equations and Σ_3 is the rotational movement equation.

$$\begin{aligned}
\Sigma_1 \quad &:= \begin{cases} \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = r_1 \end{cases} \\
\Sigma_2 \quad &:= \begin{cases} \dot{x}_{21} = x_{22} \\ \dot{x}_{22} = r_2 \end{cases} \\
\Sigma_3 \quad &:= \begin{cases} \dot{x}_{31} = x_{32} \\ \dot{x}_{32} = r_3 \end{cases}
\end{aligned} \tag{27}$$

If we consider a stabilization point in the coordinated plane XY, we could propose a vector with the desired positions $z_d = [z_{1d}\ z_{2d}\ z_{3d}] = [x_d\ y_d\ \psi_d]$ and the vector of actual states $[x\ y\ \psi]$ then expressing the system in the error coordinates e_i gives:

$$\begin{aligned}
e_1 &= z_{1d} - x \quad ; \quad e_{1int} = \int z_{1d} - \int x \\
e_2 &= z_{2d} - y \quad ; \quad e_{2int} = \int z_{2d} - \int y \\
e_3 &= z_{3d} - \psi \quad ; \quad e_{3int} = \int z_{3d} - \int \psi
\end{aligned} \tag{28}$$

$$\begin{aligned}
x_{11} &= e_{1int} \quad ; \quad x_{12} = e_1 \\
x_{21} &= e_{2int} \quad ; \quad x_{22} = e_2 \\
x_{31} &= e_{3int} \quad ; \quad x_{32} = e_3
\end{aligned} \tag{29}$$

On the other hand the saturation function σ_M is defined as:

$$\sigma_M(s) = \begin{cases} s, & \text{if } \mid s \mid < M \\ sign(s) \cdot M, & \text{otherwise} \end{cases} \tag{30}$$

Then the signal control r_i is expressed in (31), where the a_{i1}, a_{i2} are the poles of the system in closed loop if we consider the linear part of the control.

$$r_i = \sigma_{M_{i3}} \Big(\dot{z}_{id} + \sigma_{Mi2} \Big(a_{i1} x_{i2} + \sigma_{Mi1} (a_{i2} x_{i2} + a_{i1} a_{i2} x_{i1}) \Big) \Big) \tag{31}$$

According to (31) the values of control signals are:

$$\begin{aligned}
r_1 &= \sigma_{M_{13}}\left(\dot{z}_{1d}+\sigma_{M12}\Big(a_{11}x_{12}+\sigma_{M11}(a_{12}x_{12}+a_{11}a_{12}x_{11})\Big)\right)\\
r_2 &= \sigma_{M_{23}}\left(\dot{z}_{2d}+\sigma_{M22}\Big(a_{21}x_{22}+\sigma_{M21}(a_{22}x_{22}+a_{21}a_{22}x_{21})\Big)\right)\\
r_3 &= \sigma_{M_{33}}\left(\dot{z}_{3d}+\sigma_{M32}\Big(a_{31}x_{32}+\sigma_{M31}(a_{32}x_{32}+a_{31}a_{32}x_{31})\Big)\right)
\end{aligned} \tag{32}$$

Note that in the case of a stabilization to a given point, the error derivative is zero, which is not true in the case of a tracking task.

Determination of the parameters Using equation (26) and the value of the conditions for the limits, and assuming that the same dynamics in the X and Y axis is needed M_{13} and M_{23} should be equal. The signal control u becomes:

$$\begin{aligned}
\mid u_1 \mid \ &\leq max(M_{13}, M_{23}, \frac{\sqrt{2}}{2}(M_{13}+M_{23}))\\
\mid u_2 \mid \ &\leq max(M_{13}, M_{23}, \frac{\sqrt{2}}{2}((M_{13}+M_{23}))\\
\mid u_3 \mid \ &\leq M_{33}
\end{aligned} \tag{33}$$

Remark 5 *The aim of this remark is to explain how the limits were designed. Using equation (26) and the value of the conditions for the limits, the signal control u becomes:*

$$\begin{aligned}
\mid u_1 \mid \ &\leq max(M_{13}\cdot\cos(x_{33})+M_{23}\cdot\sin(x_{33}))\\
\mid u_2 \mid \ &\leq max(-M_{13}\cdot\sin(x_{33})+M_{23}\cdot\cos(x_{33}))\\
\mid u_3 \mid \ &\leq M_{33}
\end{aligned} \tag{34}$$

To determine the maximum of u_1, let define $f(x) = M_{13}\cdot\cos(x_{33}) + M_{23}\cdot\sin(x_{33})$, with here $x = x_{33}$. Therefore, $f'(x) = -M_{13}\cdot\sin(x) + M_{23}\cdot\cos(x)$. So, to have the maximal value, the equation $f'(x) = 0$ should be solved and $f''(x) <= 0$ should be verified.

The solution of the first equation is:

$$\begin{aligned}
f'(x)=0 \ \Leftrightarrow \ 0 &= -M_{13}\cdot\sin(x)+M_{23}\cdot\cos(x) && \\
\Leftrightarrow \ 1 &= \frac{M_{13}}{M_{23}}\cdot\tan(x) && \textit{if } M_{23}\neq 0 \textit{ and } x\neq\frac{\pi}{2}\\
\Leftrightarrow \ x &= \arctan\left(\frac{M_{23}}{M_{13}}\right) && \textit{if } M_{13}\neq 0
\end{aligned} \tag{35}$$

and the inequality:

$$\begin{aligned}
f''(x)\leq 0 \ \Leftrightarrow \ 0 &\geq -M_{13}\cdot\cos(x)-M_{23}\cdot\sin(x)\\
\Leftrightarrow \ x &\geq \arctan\left(\frac{M_{13}}{M_{23}}\right)
\end{aligned} \tag{36}$$

So, a possible solution is $M_{13} = M_{23}$. It is the reason why the equation 33 could be written.

With the equations (33) and (12) it get the conditions to limit the angular speed of the wheels.

$$\begin{aligned} \mid \dot{\theta}_1 \mid \; & \leq \tfrac{1}{R_w}(u_1 - u_2 - (L+l)u_3) \\ \mid \dot{\theta}_2 \mid \; & \leq \tfrac{1}{R_w}(u_1 + u_2 + (L+l)u_3) \\ \mid \dot{\theta}_3 \mid \; & \leq \tfrac{1}{R_w}(u_1 + u_2 - (L+l)u_3) \\ \mid \dot{\theta}_4 \mid \; & \leq \tfrac{1}{R_w}(u_1 - u_2 + (L+l)u_3) \end{aligned} \tag{37}$$

If we consider that the maximum speed of each wheel is 1.5 rps so $\dot{\theta}_i = 9.4248\ rad/s$.

In this case, to satisfy the equation (37) the limits M_{ij} for $i, j = 1, 2, 3$ are described in (38). The constants for each signal control are considered as $a_{11} = 22.5, a_{12} = 0.0225, a_{21} = 22.5, a_{22} = 0.0225, a_{32} = 45.0, a_{33} = 0$. These values permit that the trajectories are not far of linear region and guaranty the stability of the system.

$$\begin{array}{lll} M_{11} = 0.54 & M_{21} = 0.675 & M_{31} = 0.9360 \\ M_{12} = 2.5515 & M_{22} = 2.835 & M_{32} = 5.256 \\ M_{13} = 7.0425 & M_{23} = 7.0425 & M_{33} = 1.5 \end{array} \tag{38}$$

To sum up, figure shows the control structure with a feedback position given by a motion capture system. This will be replace in the section with the sensor.

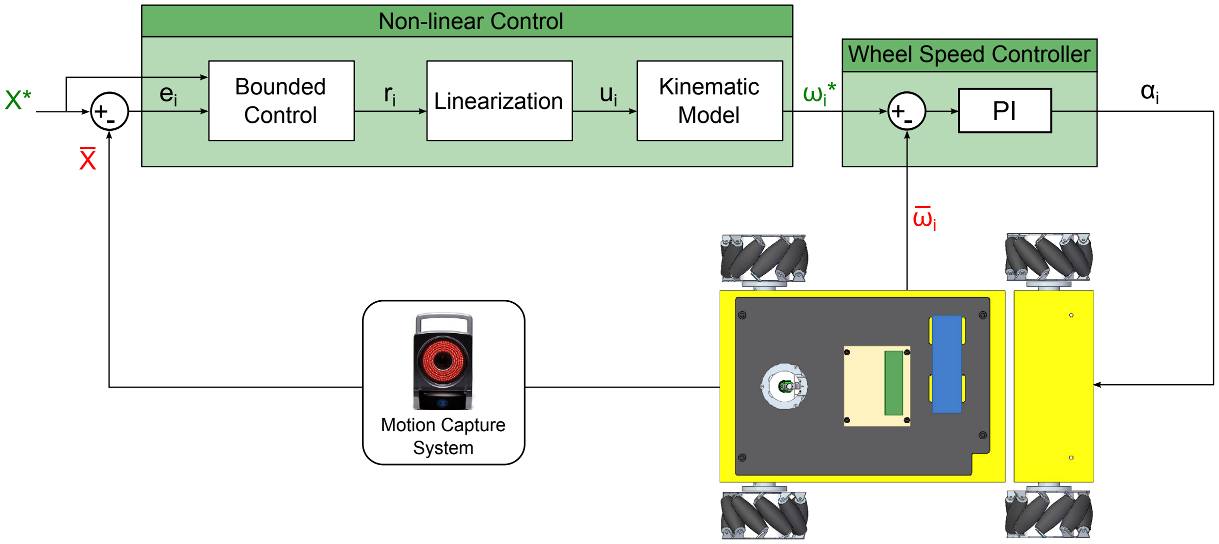


Figure D.14: Robot control architecture

Validation

Stabilization at a given position The first validation experiment shows the robot's capability to reach a an objective position from any arbitrary initial position. Figure D.15 shows the results for one test.

The effect of the saturated control is clearly seen because of the slew rate in the position response in X and Y. Indeed, in a non saturated control, one could expect an exponential or an underdampded response. But here, the saturation is well designed according to the limitation of the actuators.

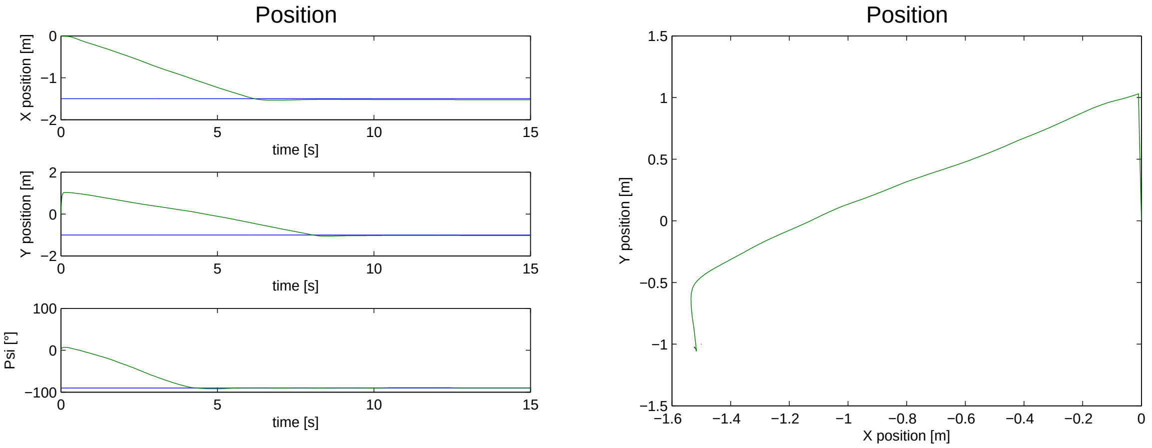


Figure D.15: a, b, c) Position of the robot in the global frame versus time. It is respectively the position in X, Y and yaw position.
d) Position of the robot in the XY plan

Tracking response The second validation consists in following a trajectory, equivalent to following a moving point.

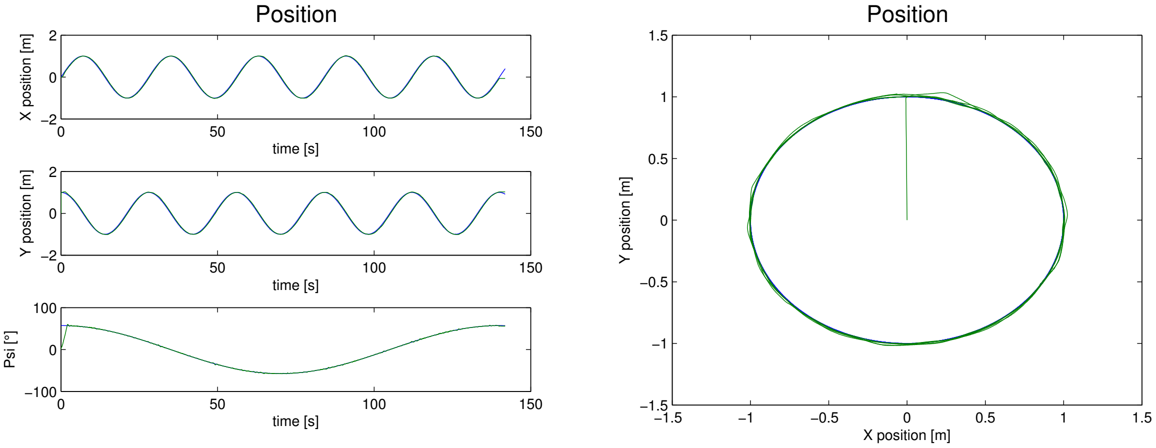


Figure D.16: a, b, c) Position of the robot in the global frame versus time. It is respectively the position in X, Y and yaw position.
d) Position of the robot in the XY plan

Figure D.16 shows the good following behavior of the circle. Even if the orientation of the robot is not tangential to the circle.

D.4 Control Law especially designed for target tracking

As seen in the section 4.7.2A, the target tracking is not the optimal one for this task. Indeed, the controller we reused in this study was firstly designed for position tracking in an absolute frame. Therefore, improvements are possible. We will present here a solution for a following situation with simple robots, without dynamic limitation. The two robots can rotate and go forward, as the pursuer lateral movement are inhibited, it is a relevant scenario compared to the real one presented in chapter 4.

D.4.4 - Theoretical scenario: known target inputs

This scenario is greatly inspired by an exercise found in the book "Mobile Robotics" [Jaulin, 2015].

Scenario explanation Let us consider two robots described by the following state equations:

$$\left\{\begin{array}{lcl} \dot{x}_1 & = & u_1 \cos\theta_1 \\ \dot{y}_1 & = & u_1 \sin\theta_1 \\ \dot{\theta}_1 & = & u_2 \end{array}\right. \quad \text{and} \quad \left\{\begin{array}{lcl} \dot{x}_2 & = & v_1 \cos\theta_2 \\ \dot{y}_2 & = & v_1 \sin\theta_2 \\ \dot{\theta}_2 & = & v_2 \end{array}\right. \tag{39}$$

In this scenario, robot N1 has for objective to follow the robot N2 (see figure D.17). These 2 robots are able to rotate on their own position and go forward and backward. This situation is similar to the one mentioned in section 4.5. Indeed, the target robot cannot do sway movements, as well as the pursuer with the control law used.

In this situation, the hypothesis of known target inputs is made. It is relevant in the case of leader-follower scenario, not really for target tracking but some estimation can be made in order to partially compensate for this unknown situation.

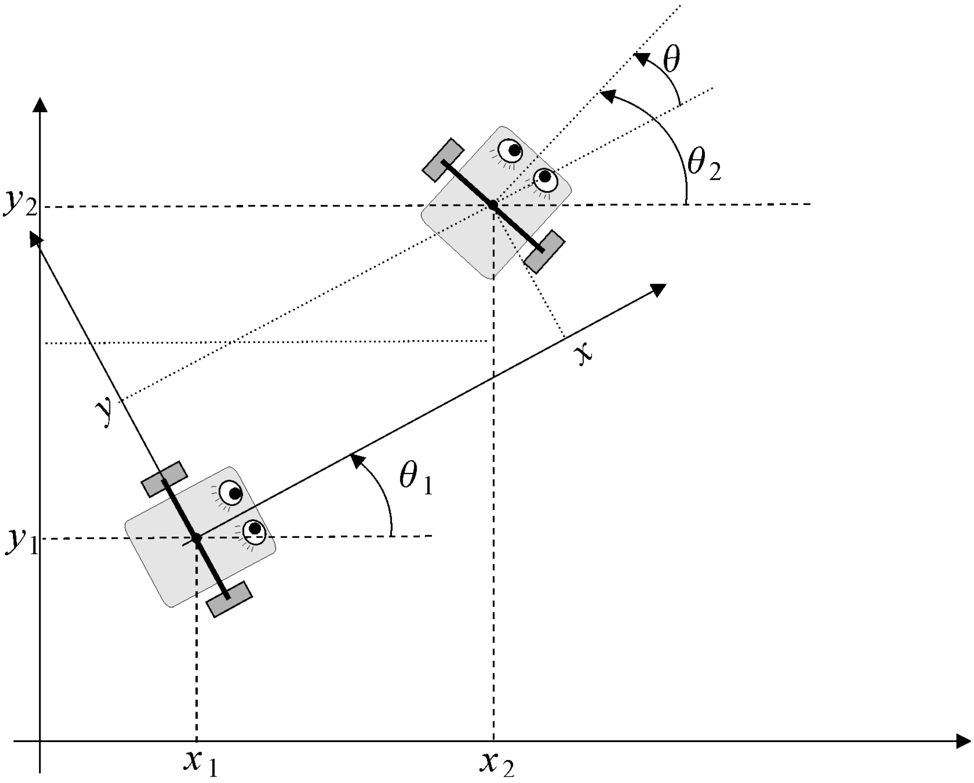


Figure D.17: Scheme of the scenario, a robot N1 follows a target robot N2. (copied from [Jaulin, 2015])

Problem formulation Let $X = [x, y, \theta]$ be the position of robot N2 in the coordinate system of the robot N1. It is expressed as follows:

$$X = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = R_1 \cdot \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ \theta_2 - \theta_1 \end{bmatrix} \tag{40}$$

where R_1 is the rotation matrix from the absolute frame to the robot N1 one.

$$R_1 = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 \\ -\sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{41}$$

The system can therefore be derived:

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = R_1 \cdot \begin{bmatrix} \dot{x}_2 - \dot{x}_1 \\ \dot{y}_2 - \dot{y}_1 \\ \dot{\theta}_2 - \dot{\theta}_1 \end{bmatrix} + \dot{\theta}_1 \cdot \begin{bmatrix} -\sin\theta_1 & \cos\theta_1 & 0 \\ -\cos\theta_1 & -\sin\theta_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ \theta_2 - \theta_1 \end{bmatrix} \tag{42}$$

with simplification using equations (39), which gives the expression of $\dot{x}_2$, $\dot{x}_1$, $\dot{y}_2$, $\dot{y}_1$, $\dot{\theta}_2$ and $\dot{\theta}_1$, and (40), for the expression of the matrix $\begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ \theta_2 - \theta_1 \end{bmatrix}$, it leads to:

$$\dot{X} = R_1 \cdot \begin{bmatrix} v_1 \cos\theta_2 - u_1 \cos\theta_1 \\ v_1 \sin\theta_2 - u_1 \sin\theta_1 \\ v_2 - u_2 \end{bmatrix} + u_2 \cdot \begin{bmatrix} -\sin\theta_1 & \cos\theta_1 & 0 \\ -\cos\theta_1 & -\sin\theta_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot R_1^{-1} \cdot \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \tag{43}$$

$$\dot{X} = \begin{bmatrix} v_1 \cdot \cos(\theta_2 - \theta_1) - u_1 \\ v_1 \cdot \sin(\theta_2 - \theta_1) \\ v_2 - u_2 \end{bmatrix} + u_2 \cdot \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \tag{44}$$

It could be expressed also as follows:

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -1 & y \\ 0 & -x \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{45}$$

Finally, we have the expression of the dynamics of the system depending on the inputs of both robots and their relative positions.

Controller Design The dynamics of the system to control is expressed in equation (45). The objective is to control the position of the robot N2 relative to the robot N1, whatever the orientation of the robot N2. This position is completely described by the state x and y. In the controller design, only these two states will be considered for the closed-loop control.

With the non linear transformation expressed in equation (46), the system can be assimilated to $\dot{Y} = U$, which is an integrator.

$$\begin{bmatrix} -1 & y \\ 0 & -x \end{bmatrix}^{-1} \cdot \left(\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} - \begin{bmatrix} v_1 \cdot \cos(\theta) \\ v_1 \cdot \sin(\theta) \end{bmatrix} \right) = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{46}$$

Therefore, it is possible to have:

$$\dot{Y} = a_0 \cdot (W - Y) + \dot{W} \tag{47}$$

with W, the variable expressing the setpoints, *i.e.* the relative position between the target and the pursuer. Therefore, if we set the error ϵ of the system is equal to $W - Y$. The equation becomes:

$$0 = a_0 \cdot \epsilon + \dot{\epsilon} \tag{48}$$

Let $P(s)$ be the polynomial expressing the dynamics the complete system, with its corrector.

$$P(s) = s + a_0 \tag{49}$$

With a pole placement method, the parameters of the controller can be set and the desired dynamics of the system can be achieved.

Finally, the expression of the output of the controller is:

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} -1 & -y/x \\ 0 & -1/x \end{bmatrix} \cdot \left(\begin{bmatrix} a_0 \cdot (w_1 - x) + \dot{w}_1 \\ a_0 \cdot (w_2 - y) + \dot{w}_2 \end{bmatrix} - \begin{bmatrix} v_1 \cdot \cos(\theta) \\ v_1 \cdot \sin(\theta) \end{bmatrix} \right) \tag{50}$$

The dynamics of the response can be adjusted by varying the value of a_0. The values w_1 and w_2 are the setpoints relative position of the target and the pursuer. As we do not want any difference of speed between the target and the pursuer, both the values $\dot{w}_1$ and $\dot{w}_2$ will be set to 0. This designed controller is depending on the target's inputs, or expressed differently its absolute speed expressed in the robot N1 frame.

D.4.4 - Comparison of 2 controllers: simulated results

Definition of the controllers The controller presented previously in section D.4.4, will be further named Omniscient Controller, because the inputs of the target are considered known.

In the experiment made, a similar controller was designed, taking also in account the limit of the pursuer robot. It is similar to this one with the hypothesis of null speed of the target. It will be later called Position Controller as only the relative position of the target is known.

Presentation of the tracking results A simulation gives the tracking response of the robot N1 with the two controllers for the same target position. In this example, the target is moving on a S-shaped trajectory at a constant forward speed (see figure D.18) and the reference setpoints are given to follow the target at a constant distance of $7m$ and an orientation of 0°.

On the figure D.18, the two controllers seems to give a very similar response. However, by looking at the figure D.19, it can be seen that the Position Controller does not converge to the correct distance. The orientation is not constant also, though it is equal to the setpoint on average.

In a pursuit scenario, the knowledge of the the target input are unknown. Therefore, the response of the Omniscient Controller is only theoretical. The hypothesis of null inputs is seen to be a solution that enable the pursuit. It can be sufficient in certain situation, as it enables to keep the target close to the line-of-sight and at a close distance to the one desired, even if not

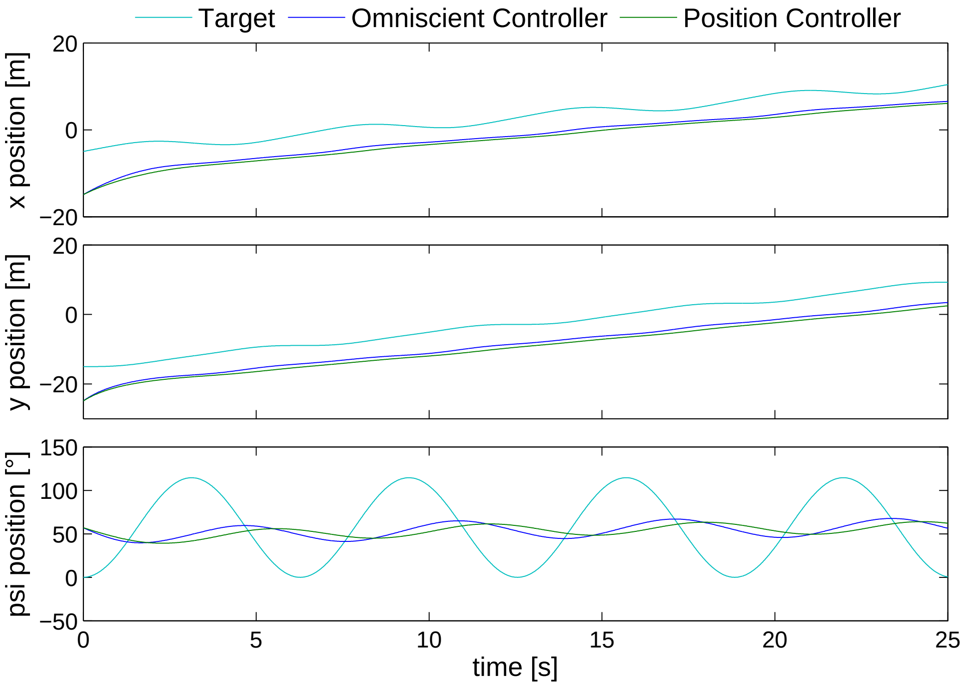


Figure D.18: Pursuit of the robot N2 (*i.e.* the target) by the robot N1 (*i.e.* the pursuer). The pose of the robots in the absolute frame are described with on a) the positions along the x axis, on b) the positions along the y axis and on c) the orientations. In light blue is the target pose and in blue and green are the pose of the pursuer with the Omniscient Controller and the Position Controller respectively

exact. It has been shown in section 4.5 that it is enough to keep the target in the FOV and to follow it. However, it is seen that it is not the optimal response. The response could be improved if the speed of the target could be either measured or estimated.

D.4.4 - Using a non-linear observer for a velocity estimator

In the experimental setup, the measured variables were the distance to the target noted d and the orientation of the target in the pursuer frame noted ψ.

These two variables are expressed according to the state variables as follows:

$$\left\{ \begin{array}{lcl} d & = & \sqrt{x^2+y^2} \\ \psi & = & \arctan 2\left(\frac{y}{x}\right) - \theta_1 \end{array} \right. \tag{51}$$

In the same manner, the expression of x and y according to d and ψ and their derivative:

$$\left\{ \begin{array}{lcl} x & = & d \cdot \cos\psi \\ y & = & d \cdot \sin\psi \end{array} \right. \quad \text{and} \quad \left\{ \begin{array}{lcl} \dot{x} & = & \dot{d} \cdot \cos\psi - d\dot{\psi} \cdot \sin\psi \\ \dot{y} & = & \dot{d} \cdot \sin\psi + d\dot{\psi} \cdot \cos\psi \end{array} \right. \tag{52}$$

Replacing the expression in the state equation (45), it leads to:

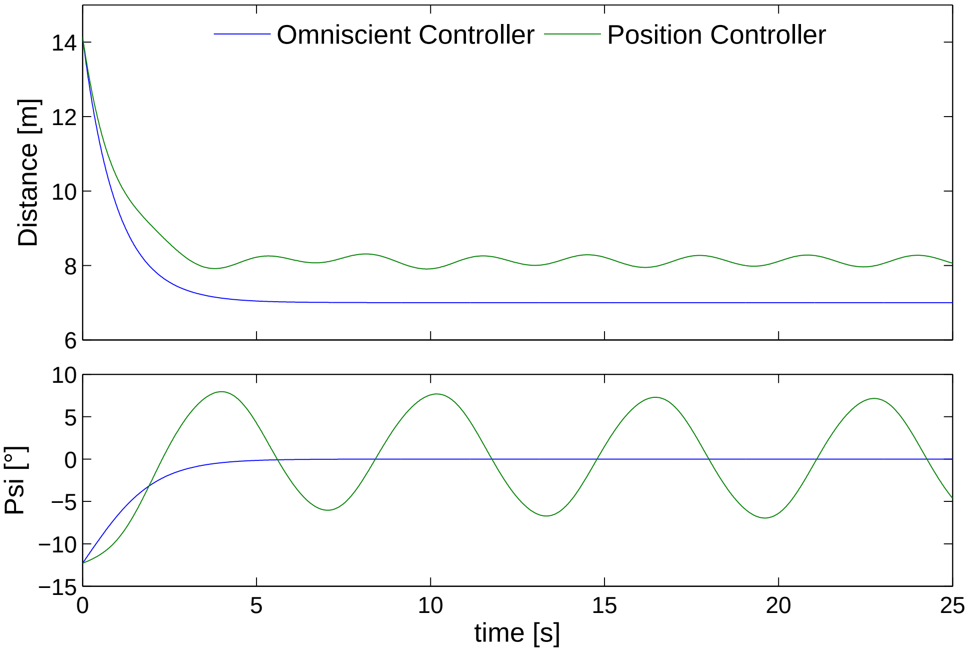


Figure D.19: Pursuit of the robot N2 (*i.e.* the target) by the robot N1 (*i.e.* the pursuer). Here are depicted the relative distance between the robots and the orientation of the target relative to the pursuer. In blue and green are the response of the pursuer with the Omniscient Controller and the Position Controller respectively.

$$\begin{bmatrix} \dot{d} \\ d\dot{\psi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\cos\psi & 0 \\ \sin\psi & -d \\ 0 & \theta - 1 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} \cos(\theta-\psi) & 0 \\ \sin(\theta-\psi) & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{53}$$

Then, the same method as presented in [Das et al., 2001, Das et al., 2002] the state vector to be estimated is given by:

$$\dot{\mathbf{X}} = f(\mathbf{X}, U, \mathbf{N})$$

$$\begin{bmatrix} \dot{\theta}_2 \\ \dot{v}_1 \\ \dot{v}_2 \\ \dot{d} \\ \dot{\psi} \\ \dot{\theta}_1 \end{bmatrix} = \begin{bmatrix} v_2 \\ 0 \\ 0 \\ -u_1 \cdot \cos(\psi) + v_1 \cdot \cos(\theta - \psi) \\ \frac{u_1 \cdot \sin(\psi) + v_1 \cdot \sin(\theta - \psi)}{d} - u_2 \\ u_2 \end{bmatrix} + \mathbf{N}(t) \tag{54}$$

where $U = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$ is the control input variable and $\mathbf{N}(t)$, the process noise. It is assumed that $\dot{v}_1 \simeq 0$ and $\dot{\mathbf{N}} \simeq 0$. Moreover, the noise is assumed to follow a Gaussian distribution with a zero mean and a covariance $\mathbf{Q}$.

The system output with sensor noise is given by:

$$\mathbf{Z}(t) = \mathbf{h}(\mathbf{X}) + \eta(t) = \begin{bmatrix} d \\ \psi \end{bmatrix} \tag{55}$$

where η is also a noise source assumed to be zero-mean Gaussian with a covariance $\mathbf{R}$

Therefore, an Extended Kalmann Filter algorithm can be expressed to estimate the state $\hat{\mathbf{X}}$ and its covariance P. An estimation of the output can be made as follows:

$$\hat{\mathbf{Z}} = \mathbf{H} \cdot \hat{\mathbf{X}} \quad \text{with} \quad \mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{56}$$

Estimate the target inputs seems to be the best option. The EKF should be completely designed for our application, the matrices $\mathbf{Q}$ and $\mathbf{R}$ should be identified. Moreover, a controller should be designed to take the limit of the robot dynamics in consideration as it has been done previously with the bounded control.

E Additional information on chapter 5

E.1 Experimental pursuit with saccadic eye controller submitted to Ambient Light Variations

The same experiments with the light variation, similar to what was done in section 4.5.6, was carried out with the Saccadic Eye Controller. The experimental conditions are identical, it means that the three tests of figure E.20 were performed in a single experimental run, starting in the dark, where the target was detected and followed up to the end of the arena before returning close to the starting-point. The light was turned on when the target came to a stop. The pursuit was repeated and the blinds of the robotic arena were finally opened for the last pursuit.

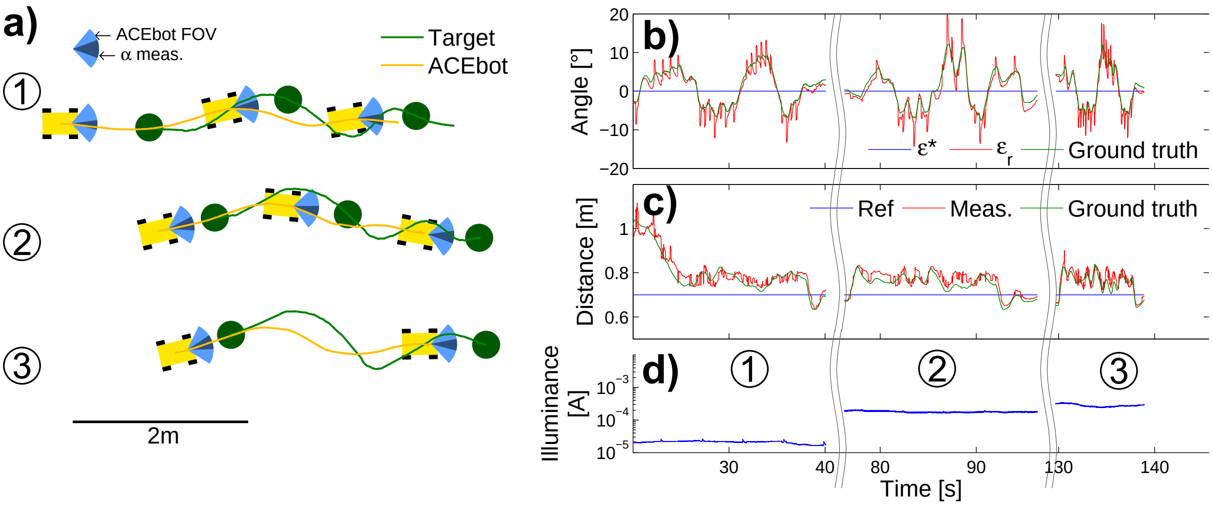


Figure E.20: Three different following trajectories under different ambient light conditions. a) The trajectories of the center of inertia of the target and the pursuer ACEbot are presented in green and orange, respectively. The robots are drawn every $8s$, with the full FOV of ACEbot in light blue, and the measured subtended angle of the target in dark blue. b) The orientation of the target in the FOV of ACEbot with the setpoint ϵ^* in blue, the measured angle ϵ_r in red and the ground truth in green. c) The distance from the target, the measured distance and the ground truth value in blue, red and green, respectively. d) Dynamic responses of the light sensor shown in fig. 4.5a, reflecting the changes in the ambient lighting, which measured 100 Lux in ①, 780 Lux in ② and 1300 Lux in ③. These responses show the robustness of the visual processing system with respect to several light levels.

Figure E.20, it can be seen that the ACEbot always kept close to the Target, while keeping it in its line of sight. The dark blue triangle shows the measured subtended angle.

During this experiment, the target was remotely controlled by hand, giving similar trajectories and ACEbot, after detecting the Target, was able to follow it unfailingly. It can be seen from figure E.20 that the distance was accurately estimated and remained constant, although slightly above the reference value. These performance are very similar the one obtain in section 4.5.6 without the decoupled eye.

Bibliography

[Arechavaleta et al., 2008] Arechavaleta, G., Laumond, J.-P., Hicheur, H., and Berthoz, A. (2008). On the nonholonomic nature of human locomotion. *Autonomous Robots*, 25(1):25–35.

[Bagheri et al., 2015] Bagheri, Z., Wiederman, S. D., Cazzolato, B., Grainger, S., and O'Carroll, D. C. (2015). Robustness and real-time performance of an insect inspired target tracking algorithm under natural conditions. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 97–102.

[Baldi and Heiligenberg, 1988] Baldi, P. and Heiligenberg, W. (1988). How sensory maps could enhance resolution through ordered arrangements of broadly tuned receivers. *Biological cybernetics*, 59(4-5):313–318.

[Barron and Srinivasan, 2006] Barron, A. and Srinivasan, M. V. (2006). Visual regulation of ground speed and headwind compensation in freely flying honey bees (apis mellifera l.). *Journal of Experimental Biology*, 209(5):978–984.

[Basiri et al., 2016] Basiri, M., Schill, F., Lima, P., and Floreano, D. (2016). On-board relative bearing estimation for teams of drones using sound. *IEEE Robotics and Automation Letters*, 1(2):820–827.

[Benson et al., 2009] Benson, J., Luke, G., Wright, C., and Barrett, S. (2009). Pre-blurred spatial sampling can lead to hyperacuity. In *Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop, 2009. DSP/SPE 2009. IEEE 13th*, pages 570 –575.

[Blanes, 1986] Blanes, C. (1986). *Appareil visuel élémentaire pour la navigation a vue d'un robot mobile autonome.* Master thesis in Neurosciences (DEA in French), Neurosciences, Univ. Aix-Marseille II, Marseille.

[Boeddeker and Egelhaaf, 2005] Boeddeker, N. and Egelhaaf, M. (2005). A single control system for smooth and saccade-like pursuit in blowflies. *Journal of Experimental Biology*, 208(8):1563–1572.

[Boeddeker et al., 2003] Boeddeker, N., Kern, R., and Egelhaaf, M. (2003). Chasing a dummy target: smooth pursuit and velocity control in male blowflies. *Proceedings of the Royal Society of London B: Biological Sciences*, 270(1513):393–399.

[Bosnak et al., 2012] Bosnak, M., Matko, D., and Blazic, S. (2012). Quadrocopter hovering using position-estimation information from inertial sensors and a high-delay video system. *Journal of Intelligent Robotic Systems*, 67(1):43–60.

[Bouabdallah et al., 2004] Bouabdallah, S., Murrieri, P., and Siegwart, R. (2004). Design and control of an indoor micro quadrotor. In *International Conference on Robotics and Automation (IEEE ICRA)*, volume 5, pages 4393–4398.

[Bouguet, 2001] Bouguet, J.-Y. (2001). Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, 5(1-10):4.

[Briod et al., 2016] Briod, A., Zufferey, J.-C., and Floreano, D. (2016). A method for ego-motion estimation in micro-hovering platforms flying in very cluttered environments. *Autonomous Robots*, 40(5):789–803.

[Bristeau et al., 2011] Bristeau, P.-J., Callou, F., Vissière, D., Petit, N., et al. (2011). The navigation and control technology inside the ar. drone micro uav. In *18th IFAC world congress*, volume 18, pages 1477–1484.

[Brückner et al., 2006] Brückner, A., Duparré, J., Bräuer, A., and Tünnermann, A. (2006). Artificial compound eye applying hyperacuity. *Optics Express*, 14(25):12076–12084.

[Campion et al., 1996] Campion, G., D'Andrea-Novel, B., and Bastin, G. (1996). Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 12(1):47–62.

[Carpenter, 1988] Carpenter, R. (1988). *Movements of the eyes (2nd rev. & enlarged ed.)*. Pion Limited.

[Carrillo et al., 2012] Carrillo, L., Flores, G., Sanahuja, G., and Lozano, R. (2012). Quad-rotor switching control: An application for the task of path following. In *American Control Conference (ACC)*, pages 4637–4642, Fairmont Queen Elizabeth, Montréal, Canada.

[Collett and Land, 1975] Collett, T. and Land, M. (1975). Visual control of flight behaviour in the hoverfly syritta pipiens l. *Journal of Comparative Physiology*, 99(1):1–66.

[Collett, 1980] Collett, T. S. (1980). Angular tracking and the optomotor response an analysis of visual reflex interaction in a hoverfly. *Journal of comparative physiology*, 140(2):145–158.

[Collett and Land, 1978] Collett, T. S. and Land, M. F. (1978). How hoverflies compute interception courses. *Journal of comparative physiology*, 125(3):191–204.

[Colonnier et al., 2015a] Colonnier, F., Manecy, A., Juston, R., Mallot, H., Leitel, R., Floreano, D., and Viollet, S. (2015a). A small-scale hyperacute compound eye featuring active eye tremor: application to visual stabilization, target tracking, and short-range odometry. *Bioinspiration & Biomimetics*, 10(2):026002.

[Colonnier et al., 2015b] Colonnier, F., Manecy, A., Juston, R., and Viollet, S. (2015b). *Visual Odometry and Low Optic Flow Measurement by Means of a Vibrating Artificial Compound Eye*, pages 153–163. Springer International Publishing, Cham.

[Colonnier et al., 2017] Colonnier, F., Ramírez-Martínez, S., Ruffier, F., and Viollet, S. (2017). Visual micro-scanning makes a sighted robot capable of target following behavior. *(In preparation)*.

[Conroy et al., 2009] Conroy, J., Gremillion, G., Ranganathan, B., and Humbert, J. S. (2009). Implementation of wide-field integration of optic flow forÂăautonomous quadrotor navigation. *Autonomous Robots*, 27(3):189.

[Conroy and Humbert, 2013] Conroy, J. and Humbert, J. S. (2013). Structure from motion in computationally constrained systems. In *Proc. SPIE, Micro- and Nanotechnology Sensors, Systems, and Applications V*, volume 8725, pages 87251G–87251G–12.

[Cowan et al., 2003] Cowan, N., Shakerina, O., Vidal, R., and Sastry, S. (2003). Vision-based follow-the-leader. In *Intelligent Robots and Systems (IROS). Proceedings. IEEE/RSJ International Conference on*, volume 2, pages 1796–1801. IEEE.

[Crombez et al., 2015] Crombez, N., Caron, G., and Mouaddib, E. M. (2015). Photometric gaussian mixtures based visual servoing. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5486–5491.

[Cronin et al., 2014] Cronin, T. W., Johnsen, S., Marshall, N. J., and Warrant, E. J. (2014). *Visual ecology*. Princeton University Press.

[Dagan et al., 2004] Dagan, E., Mano, O., Stein, G. P., and Shashua, A. (2004). Forward collision warning with a single camera. In *IEEE Intelligent Vehicles Symposium*, pages 37–42.

[Das et al., 2002] Das, A. K., Fierro, R., Kumar, V., Ostrowski, J. P., Spletzer, J., and Taylor, C. J. (2002). A vision-based formation control framework. *IEEE Transactions on Robotics and Automation*, 18(5):813–825.

[Das et al., 2001] Das, A. K., Fierro, R., Kumar, V., Southall, B., Spletzer, J., and Taylor, C. J. (2001). Real-time vision-based control of a nonholonomic mobile robot. In *Robotics and Automation. Proceedings ICRA. IEEE International Conference on*, volume 2, pages 1714–1719 vol.2.

[Davis et al., 2009] Davis, J., Barrett, S., Wright, C., and Wilcox, M. (2009). A bio-inspired apposition compound eye machine vision sensor system. *Bioinspiration & Biomimetics*, 4(4):046002.

[Delbruck and Mead, 1994] Delbruck, T. and Mead, C. A. (1994). Adaptive photoreceptor with wide dynamic range. In *Proceedings of IEEE International Symposium on Circuits and Systems - ISCAS '94*, volume 4, pages 339–342 vol.4.

[Douglass and Strausfeld, 1996] Douglass, J. K. and Strausfeld, N. J. (1996). Visual motion-detection circuits in flies: Parallel direction- and non-direction-sensitive pathways between the medulla and lobula plate. *Journal of Neuroscience*, 16(15):4551–4562.

[Dunbier et al., 2012] Dunbier, J. R., Wiederman, S. D., and O'Carroll, D. C. (2012). Predictive response facilitation to moving targets in an insect neuron. In *10th International Congress of Neuroethology*, number 234.

[Engel et al., 2012] Engel, J., Sturm, J., and Cremers, D. (2012). Accurate figure flying with a quadrocopter using onboard visual and inertial sensing. In *Workshop on Visual Control of*

Mobile Robotos (ViCoMoR) at the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, volume 320, page 240, Vilamoura, Protugal.

[Expert et al., 2012] Expert, F., Roubieu, F., and Ruffier, F. (2012). Interpolation based "time of travel" scheme in a visual motion sensor using a small 2d retina. In *IEEE Sensors Conference*, pages 2231–2234, Taipei, Taiwan.

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

[Expert et al., 2011] Expert, F., Viollet, S., and Ruffier, F. (2011). Outdoor field performances of insect-based visual motion sensors. *Journal of Field Robotics*, 28(4):529–541.

[Faessler et al., 2014] Faessler, M., Mueggler, E., Schwabe, K., and Scaramuzza, D. (2014). A monocular pose estimation system based on infrared leds. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 907–913.

[Floreano et al., 2013] Floreano, D., Pericet-Camara, R., Viollet, S., Ruffier, F., Brückner, A., Leitel, R., Buss, W., Menouni, M., Expert, F., Juston, R., Dobrzynski, M., L'Eplattenier, G., Recktenwald, F., Mallot, H., and Franceschini, N. (2013). Miniature curved artificial compound eyes. *Proc Natl Acad Sci U S A*, 110(23):9267–9272.

[Forster et al., 2014] Forster, C., Pizzoli, M., and Scaramuzza, D. (2014). SVO: Fast semi-direct monocular visual odometry. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 15–22.

[Fortey and Chatterton, 2003] Fortey, R. and Chatterton, B. (2003). A devonian trilobite with an eyeshade. *Science*, 301(5640):1689–1689.

[Franceschini, 2014] Franceschini, N. (2014). Small brains, smart machines: From fly vision to robot vision and back again. *Proceedings of the IEEE*, 102(5):751–781.

[Franceschini and Chagneux, 1997] Franceschini, N. and Chagneux, R. (1997). Repetitive scanning in the fly compound eye. In *Göttingen Neurobiology Report*, volume 2. Thieme.

[Franceschini et al., 1991] Franceschini, N., Chagneux, R., Kirschfeldand, K., and Mücke, A. (1991). Vergence eye movements in flies. In *Göttingen Neurobiology Report*, volume 1. Thieme.

[Franceschini and Kirschfeld, 1971] Franceschini, N. and Kirschfeld, K. (1971). Les phénomènes de pseudopupille dans l'œil composé dedrosophila. *Kybernetik*, 9(5):159–182.

[Franceschini et al., 1992] Franceschini, N., Pichon, J. M., and Blanes, C. (1992). From insect vision to robot vision. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 337 (1281):283–294.

[Franceschini et al., 1989] Franceschini, N., Riehle, A., and Nestour, A. L. (1989). *Facets of Vision*, chapter 17 : Directionally Selective Motion Detection by Insect Neurons, pages 360–390. Springer, Berlin, Eds : D.G. Stavenga, R.C. Hardie.

[Frost et al., 2016] Frost, S., Gorospe, G., and Teubert, C. (2016). Compound eye sensor for real-time aircraft wing deflection measurement. In *AIAA Guidance, Navigation, and Control Conference (AIAA SciTech)*, San Diego, California, USA. American Institute of Aeronautics and Astronautics.

[Gibson, 1951] Gibson, J. J. (1951). The perception of the visual world. *The American Journal of Psychology*, 64(3):440–444.

[Gollisch, 2009] Gollisch, T. (2009). Throwing a glance at the neural code: Rapid information transmission in the visual system. *HFSP Journal*, 3(1):36–46. PMID: 19649155.

[Gomez-Balderas et al., 2014] Gomez-Balderas, J., Salazar, S., Guerrero, J., and Lozano, R. (2014). Vision-based autonomous hovering for a miniature quad-rotor. *Robotica*, 32:43–61.

[Gonzalez-Bellido et al., 2016] Gonzalez-Bellido, P. T., Fabian, S. T., and Nordström, K. (2016). Target detection in insects: optical, neural and behavioral optimizations. *Current Opinion in Neurobiology*, 41:122 – 128. Microcircuit computation and evolution.

[Götz, 1965] Götz, K. G. (1965). Die optischen übertragungseigenschaften der komplexaugen von drosophila. *Kybernetik*, 2(5):215–221.

[Goulard et al., 2015] Goulard, R., Julien-Laferriere, A., Fleuriet, J., Vercher, J.-L., and Viollet, S. (2015). Behavioural evidence for a visual and proprioceptive control of head roll in hoverflies (episyrphus balteatus). *Journal of Experimental Biology*, 218(23):3777–3787.

[Gu et al., 2005] Gu, Y., Oberwinkler, J., Postma, M., and Hardie, R. (2005). Mechanisms of light adaptation in drosophila photoreceptors. *Current biology*, 15(13):1228–1234.

[Guerrero-Castellanos et al., 2014] Guerrero-Castellanos, J., Villarreal-Cervantes, M., Sánchez-Santana, J., and Ramírez-Martínez, S. (2014). Trajectory tracking of a mobile robot (3,0) by means of bounded control. *Revista Iberoamericana de Automática e Informática Industrial {RIAI}*, 11(4):426 – 434.

[Hassenstein and Reichardt, 1956] Hassenstein, B. and Reichardt, W. (1956). Systemtheoretische Analyse der Zeit-, Reihenfolgen-, und Vorzeichenauswertung bei der Bewegungsperzeption des Rüsselkafers. *Chlorophanus. Zeitschrift für Naturforschung*, 11:513–524.

[Hateren and Schilstra, 1999] Hateren, J. and Schilstra, C. (1999). Blowfly flight and optic flow. ii. head movements during flight. *Journal of Experimental Biology*, 202(11):1491–1500.

[Hausen, 1982] Hausen, K. (1982). Motion sensitive interneurons in the optomotor system of the fly. *Biological Cybernetics*, 45(2):143–156.

[Heiligenberg, 1987] Heiligenberg, W. (1987). Central processing of sensory information in electric fish. *Journal of Comparative Physiology A*, 161(4):621–631.

[Heisenberg and Wolf, 1979] Heisenberg, M. and Wolf, R. (1979). On the fine structure of yaw torque in visual flight orientation ofdrosophila melanogaster. *Journal of comparative physiology*, 130(2):113–130.

[Heisenberg and Wolf, 1984] Heisenberg, M. and Wolf, R. (1984). *Vision in Drosophila: genetics of microbehavior. Studies of brain function.* New York: Springer-Verlag.

[Hengstenberg, 1972] Hengstenberg, R. (1972). Eye movements in the housefly musca domestica. In Wehner, R., editor, *Information Processing in the Visual Systems of Arthropods*, pages 93–96. Springer Berlin Heidelberg.

[Hengstenberg, 1988] Hengstenberg, R. (1988). Mechanosensory control of compensatory head roll during flight in the blowfly calliphora erythrocephala meig. *Journal of Comparative Physiology A*, 163:159–168.

[Hengstenberg, 1993] Hengstenberg, R. (1993). Multisensory control in insect oculomotor systems. *Rev Oculomot Res*, 5:285–298.

[Herissé et al., 2012] Herissé, B., Hamel, T., Mahony, R., and Russotto, F.-X. (2012). Landing a vtol unmanned aerial vehicle on a moving platform using optical flow. *Robotics, IEEE Transactions on*, 28(1):77–89.

[Holzmann et al., 2016] Holzmann, T., Fraundorfer, F., and Bischof, H. (2016). Direct stereo visual odometry based on lines. In *Proceedings of the 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, Rom, Italy.

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

[Hongler et al., 2003] Hongler, M. O., de Meneses, Y. L., Beyeler, A., and Jacot, J. (2003). The resonant retina: exploiting vibration noise to optimally detect edges in an image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1051–1062.

[Horn and Schunck, 1981] Horn, B. K. and Schunck, B. G. (1981). Determining optical flow. *Artificial Intelligence*, 17(1):185 – 203.

[Horridge, 1977] Horridge, G. A. (1977). The compound eye of insects. *Scientific American*, 237:108–120.

[Humbert et al., 2007] Humbert, J. S., Hyslop, A., and Chinn, M. (2007). Experimental validation of wide-field integration methods for autonomous navigation. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2144–2149.

[Jaulin, 2015] Jaulin, L. (2015). *Mobile Robotics.* ISTE editions, 1st edition.

[Jung and Sukhatme, 2004] Jung, B. and Sukhatme, G. S. (2004). Detecting moving objects using a single camera on a mobile robot in an outdoor environment. In *International Conference on Intelligent Autonomous Systems*, pages 980–987.

[Juston, 2013] Juston, R. (2013). *From elementary eye to artificial compound eye : Application to robot stabilization in hover.* PhD thesis, Aix-Marseille Université.

[Juston et al., 2014] Juston, R., Kerhuel, L., Franceschini, N., and Viollet, S. (2014). Hyperacute edge and bar detection in a bioinspired optical position sensing device. *Mechatronics, IEEE/ASME Transactions on*, 19(3):1025–1034.

[Juston and Viollet, 2012] Juston, R. and Viollet, S. (2012). A miniature bio-inspired position sensing device for the control of micro-aerial robots. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1118–1124.

[Kaps and Schmid, 1996] Kaps, F. and Schmid, A. (1996). Mechanism and possible behavioural relevance of retinal movements in the ctenid spider cupiennius salei. *Journal of Experimental Biology*, 199(Pt 11):2451–2458.

[Karmeier et al., 2003] Karmeier, K., Krapp, H. G., and Egelhaaf, M. (2003). Robustness of the tuning of fly visual interneurons to rotatory optic flow. *Journal of Neurophysiology*, 90(3):1626–1634.

[Kendoul et al., 2009] Kendoul, F., Nonami, K., Fantoni, I., and Lozano, R. (2009). An adaptive vision-based autopilot for mini flying machines guidance, navigation and control. *Autonomous Robots*, 27(3):165–188.

[Kerhuel, 2009] Kerhuel, L. (2009). *Minimalist optical sensors & biomimetic oculomotor reflexes (Application to aerial robots)*. PhD thesis, UniversitÃľ des sciences et techniques de Montpellier 2.

[Kerhuel et al., 2010] Kerhuel, L., Viollet, S., and Franceschini, N. (2010). Steering by gazing: An efficient biomimetic control strategy for visually guided micro aerial vehicles. *Robotics, IEEE Transactions on*, 26(2):307–319.

[Kerhuel et al., 2012] Kerhuel, L., Viollet, S., and Franceschini, N. (2012). The vodka sensor: A bio-inspired hyperacute optical position sensing device. *Sensors Journal, IEEE*, 12(2):315–324.

[Kern et al., 2012] Kern, R., Boeddeker, N., Dittmar, L., and Egelhaaf, M. (2012). Blowfly flight characteristics are shaped by environmental features and controlled by optic flow information. *Journal of Experimental Biology*, 215(14):2501–2514.

[Kirschfeld, 1994] Kirschfeld, K. (1994). Tracking of small objects in front of a textured background by insects and vertebrates: phenomena and neuronal basis. *Biological cybernetics*, 70(5):407–415.

[Kirschfeld and Franceschini, 1969] Kirschfeld, K. and Franceschini, N. (1969). Ein mechanismus zur steuerung des lichtflusses in den rhabdomeren des komplexauges von musca. *Kybernetik*, 6(1):13–22.

[Koenderink and van Doorn, 1987] Koenderink, J. J. and van Doorn, A. J. (1987). Facts on optic flow. *Biological Cybernetics*, 56(4):247–254.

[Kwon et al., 2005] Kwon, H., Yoon, Y., Park, J. B., and Kak, A. C. (2005). Person tracking with a mobile robot using two uncalibrated independently moving cameras. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2877–2883.

[Lagorce et al., 2015] Lagorce, X., Meyer, C., Ieng, S. H., Filliat, D., and Benosman, R. (2015). Asynchronous event-based multikernel algorithm for high-speed visual features tracking. *IEEE Transactions on Neural Networks and Learning Systems*, 26(8):1710–1720.

[Land, 1982] Land, M. (1982). Scanning eye movements in a heteropod mollusc. *Journal of Experimental Biology*, 96(1):427–430.

[Land, 1992] Land, M. (1992). Visual tracking and pursuit: Humans and arthropods compared. *Journal of Insect Physiology*, 38(12):939 – 951.

[Land and Collett, 1974] Land, M. and Collett, T. (1974). Chasing behaviour of houseflies (fannia canicularis). *Journal of Comparative Physiology*, 89(4):331–357.

[Land and Nilsson, 2012] Land, M. and Nilsson, D.-E. (2012). *Animal eyes (second edition)*. Oxford University Press.

[Land, 1969] Land, M. F. (1969). Movements of the retinae of jumping spiders (salticidae: Dendryphantinae) in response to visual stimuli. *Journal of Experimental Biology*, 51(2):471–493.

[Land, 1997] Land, M. F. (1997). Visual acuity in insects. *Annual Review of Entomology*, 42(1):147–177. PMID: 15012311.

[Laughlin and Weckström, 1993] Laughlin, S. B. and Weckström, M. (1993). Fast and slow photoreceptors — a comparative study of the functional diversity of coding and conductances in the diptera. *Journal of Comparative Physiology A*, 172(5):593–609.

[Lichtsteiner et al., 2006] Lichtsteiner, P., Posch, C., and Delbruck, T. (2006). A 128 x 128 120db 30mw asynchronous vision sensor that responds to relative intensity change. In *IEEE International Solid State Circuits Conference - Digest of Technical Papers*, pages 2060–2069.

[Lichtsteiner et al., 2008] Lichtsteiner, P., Posch, C., and Delbruck, T. (2008). A 128 × 128 120 db 15 956 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576.

[Lim et al., 2012] Lim, H., Lee, H., and H.J., K. (2012). Onboard flight control of a micro quadrotor using single strapdown optical flow sensor. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 495–500, Vilamoura, Algarve, Portugal.

[Lindemann et al., 2008] Lindemann, J. P., Weiss, H., Möller, R., and Egelhaaf, M. (2008). Saccadic flight strategy facilitates collision avoidance: closed-loop performance of a cyberfly. *Biological Cybernetics*, 98(3):213.

[Ling and Shen, 2015] Ling, Y. and Shen, S. (2015). Dense visual-inertial odometry for tracking of aggressive motions. In *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 576–583.

[Liu et al., 2015] Liu, H., Luo, J., Wu, P., Xie, S., and Li, H. (2015). Symmetric kullback-leibler metric based tracking behaviors for bioinspired robotic eyes. *Applied Bionics and Biomechanics*, 2015:11.

[Liu et al., 2016] Liu, H., Moeys, D. P., Das, G., Neil, D., Liu, S. C., and Delbrück, T. (2016). Combined frame- and event-based detection and tracking. In *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2511–2514.

[Lucas et al., 1981] Lucas, B. D., Kanade, T., et al. (1981). An iterative image registration technique with an application to stereo vision. In *IJCAI*, volume 81, pages 67,4–679, Vancouver, Canada.

[Luke et al., 2012] Luke, G., Wright, C., and Barrett, S. (2012). A multiaperture bioinspired sensor with hyperacuity. *Sensors Journal, IEEE*, 12(2):308–314.

[Ma et al., 2013] Ma, K. Y., Chirarattananon, P., Fuller, S. B., and Wood, R. J. (2013). Controlled flight of a biologically inspired, insect-scale robot. *Science*, 340(6132):603–607.

[Mafrica et al., 2015] Mafrica, S., Godiot, S., Menouni, M., Boyron, M., Expert, F., Juston, R., Marchand, N., Ruffier, F., and Viollet, S. (2015). A bio-inspired analog silicon retina with michaelis-menten auto-adaptive pixels sensitive to small and large changes in light. *Opt. Express*, 23(5):5614–5635.

[Mafrica et al., 2016] Mafrica, S., Servel, A., and Ruffier, F. (2016). Minimalistic optic flow sensors applied to indoor and outdoor visual guidance and odometry on a car-like robot. *Bioinspiration & Biomimetics*, 11(6):066007.

[Mahony et al., 2008] Mahony, R., Hamel, T., and Pflimlin, J.-M. (2008). Nonlinear complementary filters on the special orthogonal group. *Automatic Control, IEEE Transactions on*, 5(5):1203–1218.

[Maisak et al., 2013] Maisak, M. S., Haag, J., Ammer, G., Serbe, E., Meier, M., Leonhardt, A., Schilling, T., Bahl, A., Rubin, G. M., Nern, A., B.J., D., D.F., R., E., H., and A., B. (2013). A directional tuning map of drosophila elementary motion detectors. *Nature*, 500(7461):212–216.

[Manecy, 2014] Manecy, A. (2014). *RT-MaG Project, http://www.gipsa-lap.fr/projet/RT-MaG/*.

[Manecy, 2015] Manecy, A. (2015). *Bio-inspired visual strategies: application to stabilization of a micro UAV and to traget tracking*. Theses, EEATS.

[Manecy et al., 2016] Manecy, A., Diperi, J., Boyron, M., Marchand, N., and Viollet, S. (2016). A novel hyperacute gimbal eye to implement precise hovering and target tracking on a quadrotor. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3212–3218.

[Manecy et al., 2015] Manecy, A., Marchand, N., Ruffier, F., and Viollet, S. (2015). X4-mag: A low-cost open-source micro-quadrotor and its linux-based controller. *International Journal of Micro Air Vehicles*, 7(2):89–110.

[Mariottini et al., 2009] Mariottini, G. L., Morbidi, F., Prattichizzo, D., Valk, N. V., Michael, N., Pappas, G., and Daniilidis, K. (2009). Vision-based localization for leader-follower formation control. *IEEE Transactions on Robotics*, 25(6):1431–1438.

[Maxim et al., 2008] Maxim, P. M., Hettiarachchi, S., Spears, W. M., Spears, D. F., Hamann, J., Kunkel, T., and Speiser, C. (2008). Trilateration localization for multi-robot teams. In *Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics, Special Session on MultiAgent Robotic Systems (ICINCO 08).*

[Metzen et al., 2015] Metzen, M. G., Jamali, M., Carriot, J., Avila-Akerberg, O., Cullen, K. E., and Chacron, M. J. (2015). Coding of envelopes by correlated but not single-neuron activity requires neural variability. *Proceedings of the National Academy of Sciences*, 112(15):4791–4796.

[Mischiati et al., 2015] Mischiati, M., Lin, H.-T., Herold, P., Imler, E., Olberg, R., and Leonardo, A. (2015). Internal models direct dragonfly interception steering. *Nature*, 517:333–338.

[Mkrtchyan et al., 2009] Mkrtchyan, A., Schultz, R., and Semke, W. (2009). Vision-based autopilot implementation using a quadrotor helicopter. In *AIAA Infotech@ Aerospace Conference*, page 1831, Seattle, Washington.

[Mura and Franceschini, 1996] Mura, F. and Franceschini, N. (1996). Obstacle avoidance in a terrestrial mobile robot provided with a scanning retina. In *Intelligent Vehicles Symposium, Proceedings of the 1996 IEEE*, pages 47–52.

[Mura and Shimoyama, 1998] Mura, F. and Shimoyama, I. (1998). Visual guidance of a small mobile robot using active, biologically-inspired, eye movements. In *Robotics and Automation. Proceedings. 1998 IEEE International Conference on*, volume 3, pages 1859–1864.

[Murtaugh and Criel, 1966] Murtaugh, S. and Criel, H. (1966). Fundamentals of proportional navigation. *Spectrum, IEEE*, 3(12):75–85.

[Nilsson, 2009] Nilsson, D.-E. (2009). The evolution of eyes and visually guided behaviour. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 364(1531):2833–2847.

[Nitsche et al., 2015] Nitsche, M., Krajnik, T., Cizek, P., Mejail, M., Duckett, T., et al. (2015). Whycon: an efficient, marker-based localization system. In *Workshop on Aerial Open-source Robotics, 28 September 2015, Hamburg.*

[Nityananda et al., 2016] Nityananda, V., Tarawneh, G., Rosner, R., Nicolas, J., Crichton, S., and Read, J. (2016). Insect stereopsis demonstrated using a 3d insect cinema. *Scientific reports*, 6:18718.

[Northrop, 2000] Northrop, R. B. (2000). *Introduction to dynamic modeling of neuro-sensory systems.* CRC Press.

[Olberg et al., 2000] Olberg, R., Worthington, A., and Venator, K. (2000). Prey pursuit and interception in dragonflies. *Journal of Comparative Physiology A*, 186(2):155–162.

[Olberg, 2012] Olberg, R. M. (2012). Visual control of prey-capture flight in dragonflies. *Current opinion in neurobiology*, 22(2):267–271.

[Olberg et al., 2007] Olberg, R. M., Seaman, R. C., Coats, M. I., and Henry, A. F. (2007). Eye movements and target fixation during dragonfly prey-interception flights. *Journal of Comparative Physiology A*, 193(7):685–693.

[Orchard et al., 2013] Orchard, G., Benosman, R., Etienne-Cummings, R., and Thakor, N. V. (2013). A spiking neural network architecture for visual motion estimation. In *2013 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 298–301.

[Pericet-Camara et al., 2011] Pericet-Camara, R., Dobrzynski, M., L'Eplattenier, G., Zufferey, J.-C., Expert, F., Juston, R., Ruffier, F., Franceschini, N., Viollet, S., Menouni, M., Godiot, S., Brückner, A., Buss, W., Leitel, R., Recktenwald, F., Yuan, C., Mallot, H., and Floreano, D. (2011). CURVACE - CURVed Artificial Compound Eyes. In *Proceedings of the 2nd European Future Technologies Conference and Exhibition 2011 (FET 11)*, volume 7, pages 308–309, Hamburg, Germany.

[Perry and Desplan, 2016] Perry, M. W. and Desplan, C. (2016). Love spots. *Current Biology*, 26(12):R484 – R485.

[Portelli et al., 2010] Portelli, G., Ruffier, F., and Franceschini, N. (2010). Honeybees change their height to restore their optic flow. *Journal of Comparative Physiology A*, 196(4):307–313.

[Posch, 2012] Posch, C. (2012). Bio-inspired vision. *Journal of Instrumentation*, 7(01):C01054.

[Posch et al., 2008] Posch, C., Matolin, D., and Wohlgenannt, R. (2008). An asynchronous time-based image sensor. In *2008 IEEE International Symposium on Circuits and Systems*, pages 2130–2133.

[Posch et al., 2011] Posch, C., Matolin, D., and Wohlgenannt, R. (2011). A qvga 143 db dynamic range frame-free pwm image sensor with lossless pixel-level video compression and time-domain cds. *IEEE Journal of Solid-State Circuits*, 46(1):259–275.

[Pritchard, 1966] Pritchard, G. (1966). On the morphology of the compound eyes of dragonflies (odonata: Anisoptera), with special reference to their role in prey capture. *Proceedings of the Royal Entomological Society of London. Series A, General Entomology*, 41(1-3):1–8.

[Prokopowicz and Cooper, 1995] Prokopowicz, P. N. and Cooper, P. R. (1995). The dynamic retina: Contrast and motion detection for active vision. *International Journal of Computer Vision*, 16(3):191–204.

[Raharijaona et al., 2015] Raharijaona, T., Mignon, P., Juston, R., Kerhuel, L., and Viollet, S. (2015). Hypercube: A small lensless position sensing device for the tracking of flickering infrared leds. *Sensors*, 15(7):16484.

[Riley et al., 2008] Riley, D., Harmann, W., Barrett, S., and Wright, C. (2008). Musca domestica inspired machine vision sensor with hyperacuity. *Bioinspiration & Biomimetics*, 3(2):026003 (13pp).

[Roberts et al., 2012] Roberts, J., Stirling, T., Zufferey, J.-C., and Floreano, D. (2012). 3-d relative positioning sensor for indoor flying robots. *Autonomous Robots*, 33(1):5–20.

[Roberts et al., 2015] Roberts, T., Staras, K., Husbands, P., and Philippides, A. (2015). *Entraining and Copying of Temporal Correlations in Dissociated Cultured Neurons*, pages 223–226. Springer International Publishing, Cham.

[Rolfs, 2009] Rolfs, M. (2009). Microsaccades: Small steps on a long way. *Vision Research*, 49(20):2415 – 2441.

[Ross et al., 2012] Ross, R., Devlin, J., and Wang, S. (2012). Toward refocused optical mouse sensors for outdoor optical flow odometry. *IEEE Sensors Journal*, 12(6):1925–1932.

[Rossel, 1980] Rossel, S. (1980). Foveal fixation and tracking in the praying mantis. *Journal of comparative physiology*, 139(4):307–331.

[Roubieu et al., 2011] Roubieu, F., Expert, F., M, B., B-J., F., Viollet, S., and Ruffier, F. (2011). A novel 1-gram insect based device measuring visual motion along 5 optical directions. In *International Conference on Sensors (IEEE Sensors)*, Limerick, Ireland.

[Roubieu et al., 2014] Roubieu, F., Serres, J., Colonnier, F., Franceschini, N., Viollet, S., and Ruffier, F. (2014). A biomimetic vision-based hovercraft accounts for beesâĂŹ complex behaviour in various corridors. *Bioinspiration & Biomimetics*, 9(3):036003.

[Roubieu et al., 2012] Roubieu, F. L., Expert, F., Sabiron, G., and Ruffier, F. (2012). Two-directional 1-g visual motion sensor inspired by the fly's eye. *IEEE Sensors Journal*, 13 (3):1025–1035.

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

[Ruffier and Franceschini, 2014] Ruffier, F. and Franceschini, N. (2014). Optic flow regulation in unsteady environments: A tethered mav achieves terrain following and targeted landing over a moving platform. *Journal of Intelligent & Robotic Systems*, pages 1–19.

[Ruffier et al., 2003] Ruffier, F., Viollet, S., Amic, S., and Franceschini, N. (2003). Bio-inspired optical flow circuits for the visual guidance of micro-air vehicles. *Proceeding of IEEE International Symposium on Circuits and Systems (ISCAS)*, 3:846–849.

[Salaris et al., 2015] Salaris, P., Vassallo, C., Souères, P., and Laumond, J. P. (2015). The geometry of confocal curves for passing through a door. *IEEE Transactions on Robotics*, 31(5):1180–1193.

[Sandeman, 1978] Sandeman, D. (1978). Eye-scanning during walking in the crab leptograpsus variegatus. *Journal of comparative physiology*, 124(3):249–257.

[Santos-Victor and Sandini, 1997] Santos-Victor, J. and Sandini, G. (1997). Embedded visual behaviors for navigation. *Robotics and Autonomous Systems*, 19(3):299 – 313.

[Saska, 2015] Saska, M. (2015). Mav-swarms: Unmanned aerial vehicles stabilized along a given path using onboard relative localization. In *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, pages 894–903.

[Scaramuzza and Fraundorfer, 2011] Scaramuzza, D. and Fraundorfer, F. (2011). Visual odometry [tutorial]. *IEEE Robotics Automation Magazine*, 18(4):80–92.

[Schilstra and Hateren, 1999] Schilstra, C. and Hateren, J. (1999). Blowfly flight and optic flow. i. thorax kinematics and flight dynamics. *Journal of Experimental Biology*, 202(11):1481–1490.

[Serres et al., 2008] Serres, J., Masson, G., Ruffier, F., and Franceschini, N. (2008). A bee in the corridor: centering and wall-following. *Naturwissenschaften*, 95(12):1181–1187.

[Shaw, 1981] Shaw, S. R. (1981). *Neurones without impulses: their significance for vertebrate and invertebrate nervous systems*, volume 6, chapter Anatomy and physiology of identified non-spiking cells in the photoreceptor-lamina complex of the compound eye of insects, especially Diptera, pages 61–116. Cambridge University Press.

[Shen et al., 2015] Shen, S., Michael, N., and Kumar, V. (2015). Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft mavs. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5303–5310.

[Shen et al., 2013] Shen, S., Mulgaonkar, Y., Michael, N., and Kumar, V. (2013). Vision-based state estimation for autonomous rotorcraft mavs in complex environments. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1758–1764, Karlsruhe, Germany.

[Shishika et al., 2015] Shishika, D., Yim, J., and Paley, D. (2015). Bio-inspired pursuit with autonomous hovercraft using lyapunov-based control. In *American Control Conference (ACC), 2015*, pages 3107–3113. IEEE.

[Snyder, 1979] Snyder, A. (1979). *Handbook of Sensory Physiology*, chapter Physics of Vision in Compound eyes, pages 225–313. Springer-Verlag Berlin Heidelberg New York.

[Song et al., 2013] Song, Y. M., Xie, Y., Malyarchuk, V., Xiao, J., Jung, I., Choi, K., Liu, Z., Park, H., Lu, C., Kim, R., Li, R., Crozier, K. B., Huang, Y., and Rogers, J. A. (2013). Digital cameras with designs inspired by the arthropod eye. *Nature*, 497(7447):95–99.

[Srinivasan, 1994] Srinivasan, M. (1994). An image-interpolation technique for the computation of optic flow and egomotion. *Biological Cybernetics*, 71:401–415.

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

[Srinivasan et al., 1991] Srinivasan, M. V., Lehrer, M., Kirchner, W. H., and Zhang, S. W. (1991). Range perception through apparent image speed in freely flying honeybees. *Visual Neuroscience*, 6(5):519–535.

[Stavenga, 2003] Stavenga, D. G. (2003). Angular and spectral sensitivity of fly photoreceptors. i. integrated facet lens and rhabdomere optics. *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, 189(1):1–17.

[Strausfeld, 1976] Strausfeld, N. J. (1976). *Atlas of an insect brain.* Springer-Verlag.

[Tatar et al., 2014] Tatar, M., Popovici, C., Mandru, D., Ardelean, I., and Plesa, A. (2014). Design and development of an autonomous omni-directional mobile robot with mecanum wheels. In *Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on.*

[Taylor and Krapp, 2007] Taylor, G. K. and Krapp, H. G. (2007). Sensory systems and flight stability: What do insects measure and why? In Casas, J. and Simpson, S., editors, *Insect Mechanics and Control*, volume 34 of *Advances in Insect Physiology*, pages 231 – 316. Academic Press.

[Tomasi and Kanade, 1992] Tomasi, C. and Kanade, T. (1992). Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154.

[Tsai et al., 2011] Tsai, C.-C., Tai, F.-C., and Lee, Y.-R. (2011). Motion controller design and embedded realization for mecanum wheeled omnidirectional robots. In *Proceedings of the 8th World Congress on Intelligent Control and Automation*, pages 546–551.

[Tuthill et al., 2013] Tuthill, J., Nern, A., Holtz, S., Rubin, G., and Reiser, M. (2013). Contributions of the 12 neuron classes in the fly lamina to motion vision. *Neuron*, 79(1):128 – 140.

[Viboonchaicheep et al., 2003] Viboonchaicheep, P., Shimada, A., and Kosaka, Y. (2003). Position rectification control for mecanum wheeled omni-directional vehicles. *Industrial Electronics Society, 2003. IECON '03. The 29th Annual Conference of the IEEE*, 1:854–859.

[Viollet, 2014] Viollet, S. (2014). Vibrating makes for better seeing: from the fly's micro eye movements to hyperacute visual sensors. *Frontiers in Bioengineering and Biotechnology*, 2(9).

[Viollet and Franceschini, 1999] Viollet, S. and Franceschini, N. (1999). Biologically-inspired visual scanning sensor for stabilization and tracking. In *Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on*, volume 1, pages 204–209vol.1.

[Viollet and Franceschini, 2010] Viollet, S. and Franceschini, N. (2010). A hyperacute optical position sensor based on biomimetic retinal micro-scanning. *Sensors and Actuators A: Physical*, 160(1-2):60 – 68.

[Viollet et al., 2014] Viollet, S., Godiot, S., Leitel, R., Buss, W., Breugnon, P., Menouni, M., Juston, R., Expert, F., Colonnier, F., L'Eplattenier, G., Brückner, A., Kraze, F., Mallot, H., Franceschini, N., Pericet-Camara, R., Ruffier, F., and Floreano, D. (2014). Hardware architecture and cutting-edge assembly process of a tiny curved compound eye. *Sensors*, 14(11):21702–21721.

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

[von Frisch, 1967] von Frisch, K. (1967). *The Dance Language and Orientation of Bees.* Harvard University Press.

[Wardill et al., 2015] Wardill, T. J., Knowles, K., Barlow, L., Tapia, G., Nordström, K., Olberg, R. M., and Gonzalez-Bellido, P. T. (2015). The killer fly hunger games: target size and speed predict decision to pursuit. *Brain, behavior and evolution*, 86(1):28–37.

[Wenzel et al., 2010] Wenzel, K., Rosset, P., and Zell, A. (2010). Low-cost visual tracking of a landing place and hovering flight control with a microcontroller. *Journal of Intelligent Robotic Systems*, 57:297–311.

[Wenzel et al., 2012] Wenzel, K. E., Masselli, A., and Zell, A. (2012). Visual tracking and following of a quadrocopter by another quadrocopter. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 4993–4998. IEEE.

[Westheimer, 1975] Westheimer, G. (1975). Editorial: Visual acuity and hyperacuity. *Investigative Ophthalmology & Visual Science*, 14(8):570–2.

[Westheimer, 1981] Westheimer, G. (1981). *Visual hyperacuity.* Sensory Physiology 1. Springer-Verlag.

[Westheimer, 2009] Westheimer, G. (2009). Hyperacuity. *Encyclopedia of Neuroscience*, 5:45–50.

[Wiederman and O'Carroll, 2011] Wiederman, S. D. and O'Carroll, D. C. (2011). Discrimination of features in natural scenes by a dragonfly neuron. *Journal of Neuroscience*, 31(19):7141–7144.

[Wilcox and Thelen, 1999] Wilcox, M. and Thelen, Jr., D. (1999). A retina with parallel input and pulsed output, extracting high-resolution information. *Neural Networks, IEEE Transactions on*, 10(3):574–583.

[Willis et al., 2012] Willis, K., Brockmeyer, E., Hudson, S., and Poupyrev, I. (2012). Printed optics: 3d printing of embedded optical elements for interactive devices. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 589–598, New York, NY, USA. ACM.

[Wright and Barrett, 2013] Wright, C. and Barrett, S. (2013). Chapter 1 - biomimetic vision sensors. In Lakhtakia, A. and Martín-Palma, R. J., editors, *Engineered Biomimicry*, pages 1 – 36. Elsevier, Boston.

[Wu et al., 2013] Wu, Y., Lim, J., and Yang, M. H. (2013). Online object tracking: A benchmark. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conferenceon*, pages 2411–2418.

[Yang et al., 2013] Yang, S., Scherer, S., and Zell, A. (2013). An onboard monocular vision system for autonomous takeoff, hovering and landing of a micro aerial vehicle. *Journal of Intelligent Robotic Systems*, 69:499–515.

[Zhang et al., 2009] Zhang, T., Kang, Y., Achtelik, M., Kühnlenz, K., and Buss, M. (2009). Autonomous hovering of a vision/imu guided quadrotor. In *International Conference on Mechatronics and Automation (IEEE ICRA)*, pages 2870–2875, Changchun, China.

[Zingg et al., 2010] Zingg, S., Scaramuzza, D., Weiss, S., and Siegwart, R. (2010). Mav navigation through indoor corridors using optical flow. In *2010 IEEE International Conference on Robotics and Automation*, pages 3361–3368.

[Zufferey and Floreano, 2006] Zufferey, J.-C. and Floreano, D. (2006). Fly-inspired visual steering of ultralight indoor aircraft. *IEEE Transactions on Robotics*, 22(1):137–146.

Résumé de la thèse en français

A Introduction

La robotique aérienne est capable aujourd'hui de plus en plus de prouesses techniques. Hier, les applications militaires étaient les premières à recourir des aéronefs sans pilotes capables de suivre une trajectoire définie. Avec l'avènement des smartphones, combinant à la fois capteurs et puissance calculatoire embarquée, il a été possible de réduire massivement les coûts grâce à des économies d'échelle. Des quadrirotors sont apparues avec des applications industrielles telles que l'inspection d'ouvrages d'art, la photographie aérienne et l'agriculture. Il n'est quasiment plus nécessaire d'envoyer des grimpeurs pour monter sur les pilonnes électriques ou de grues pour faire des prises de vue en hauteur.

Mais pour que la prise en main de tels engins soit possible par des non initiés qui les voient comme des outils, pour prendre une photo par exemple. L'autonomie de tel système doit être la plus grande possible, de la stabilisation par rapport à un point GPS jusqu'à suivre un sportif tout en évitant les obstacles et autres objets en mouvement. Dans des environnements encombrés, recevoir un signal GPS est impossible. Dès lors, des solutions de stabilisation par caméra sont apparues. C'est une solution qui est assez gourmande en ressources calculatoires. Il y a encore peu le rafraichissement était également un frein à l'exploitation des dynamiques rapides des quadrirotors. Mais traiter toutes les images reçues à haut débit en temps réel reste un réel challenge.

Ainsi, s'inspirer du vivant peut permettre de trouver des solutions à la fois efficaces et minimales. En effet, les insectes volants tels que les mouches, les abeilles ou les libellules sont capables de performances de vol assez extraordinaire pour le roboticien d'aujourd'hui. Ces animaux peuvent naviguer, éviter obstacles et prédateurs, chasser et même indiquer une source de nourriture à leurs congénères. Ils sont capable de remplir ces tâches malgré leurs contraintes énergétiques.

Au cours de cette thèse, nous avons doté l'œil composé artificiel CurvACE, qui reprend certaines caractéristiques de l'œil de la mouche, d'un système de vibrations mécaniques afin de lui appliquer des micro-mouvements à hautes fréquences ($50Hz$) (voir figure A.1). J'ai contribué au développement de nouveaux algorithmes de traitement des signaux visuels afin d'effectuer une stabilisation visuelle d'un robot aérien et un suivi de cible par un robot mobile.

B Un œil composé artificiel doté d'une vibration permettant de stabiliser un robot volant

D'après les travaux de L. Kerhuel [Kerhuel et al., 2012], il a été démontré qu'à partir de deux photorécepteurs soumis à une vibration, on peut localiser un front de contraste et une barre,

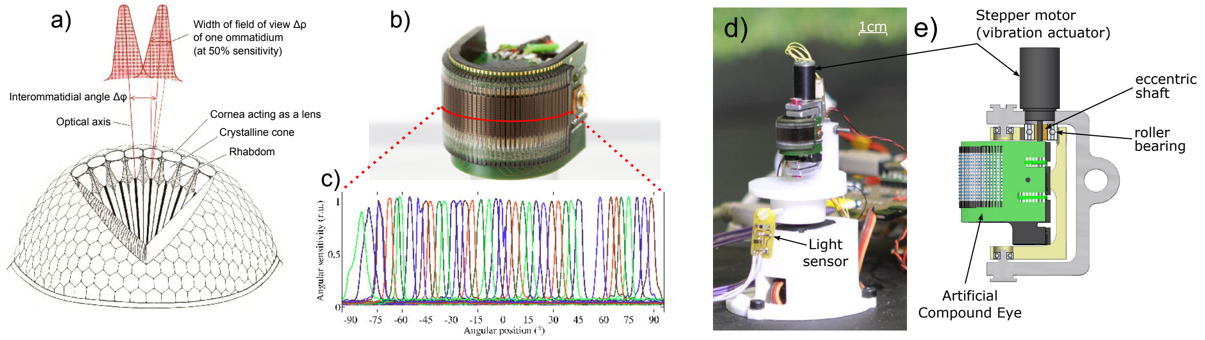


Figure A.1: a) Schéma d'un oeil composé, montrant sa structure et ses caractéristiques optiques principales, à savoir, la sensibilité angulaire gaussienne et l'angle interommatidial $\Delta\varphi$ (adaptée de [Horridge, 1977]). b) Photo du capteur CurvACE. c) Sensibilité angulaire de l'ensemble des photorécepteurs d'une ligne (extrait de [Floreano et al., 2013]). d-e) Active CurvACE équipé de son système de vibration, un moteur pas-à-pas avec un arbre eccentrique.

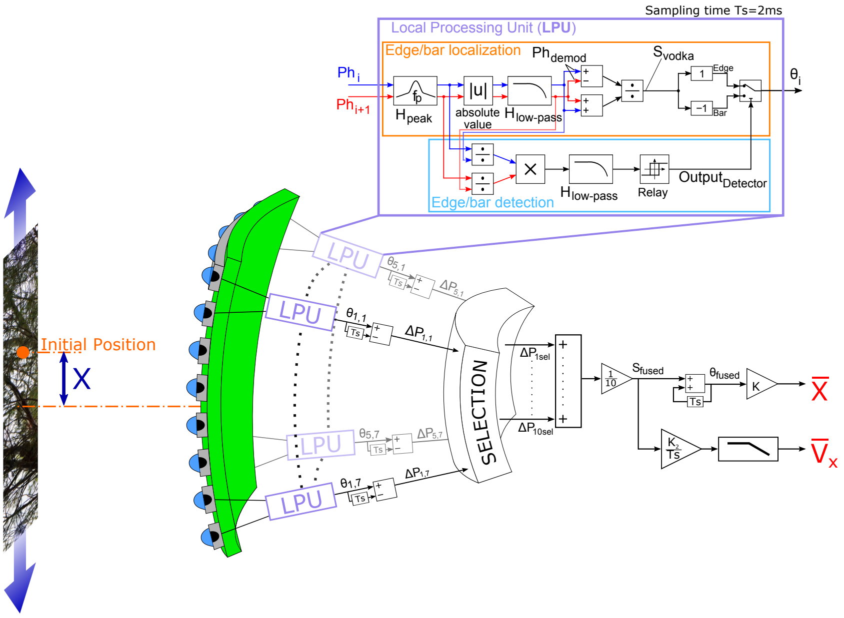


Figure B.2: Algorithme de traitement visuel à partir des signaux de 8×5 photorécepteurs. 35 "Local Processing Unit" (LPU) permettent de traiter les signaux par paire afin de donner une mesure de déplacement local. Chaque LPU est composé d'une partie filtrage pour démoduler les signaux de chaque pixel. Ensuite, la division sur la somme est effectuée et un gain est appliqué en fonction du détecteur front/bar. La fusion des sorties des LPU se fait grâce à un processus de sélection basée sur la somme des signaux démodulés. Seules les 10 LPU ayant la plus grande somme sont utilisées dans le calcul de la sortie S_{fused} qui en est une moyenne. Les données de vitesse $\bar{V}_x$ et de position $\bar{X}$ sont calculées directement à partir du signal S_{fused}. Modifié de [Colonnier et al., 2015a].

la barre étant une succession de deux contrastes équivalents. Le traitement appliqué est une démodulation des signaux visuels; cela correspond à effectuer un détecteur d'enveloppe après un filtrage passe-bande. Puis, la différence des deux signaux démodulés sur leur somme, appelé principe *VODKA* permet d'obtenir une caractéristique en tangente hyperbolique par rapport à la position d'un contraste. En revanche, la caractéristique est plus complexe et non monotone dans le cas d'une barre. Cependant, la détection de ces 2 types de contrastes permet d'obtenir une sortie monotone et quasi linéaire en appliquant un gain correspondant [Juston et al., 2014]. L'ensemble de ces traitements est résumé figure B.2 et appelé "Local Processing Unit" (LPU).

Nous avons donc travaillé à l'élaboration d'un algorithme fusionnant les mesures reçues de 35 LPU afin d'obtenir une localisation de différents contrastes en même temps et ainsi se localiser par rapport à une surface plane texturée. La fusion est effectuée à partir des 10 paires qui voient les contrastes les plus élevés. Une moyenne des 10 mesures du mouvement angulaire observées à chaque pas de temps est intégrée pour donner un déplacement global par rapport à la position initiale. Une mise à l'échelle est effectuée suivant l'hypothèse que la distance à la surface plane est connue et constante (voir figure B.2).

Cet algorithme a été testé sur un robot nommé HyperRob ayant pour objectif de se stabiliser à une position précise au-dessus d'un panneau contrasté (ici une photo de branches de pins). Le robot est accroché au bout d'un bras afin de se déplacer latéralement suivant une trajectoire circulaire dans le plan azimutal. Son œil découplé du corps permet de compenser une rotation du robot en roulis et ainsi, grâce à un réflexe de type vestibulo oculaire, de maintenir la direction de visée toujours verticale. Cela permet au capteur de toujours être soumis à une translation et à une distance constante du panneau (voir figure B.3 pour une photo du robot).

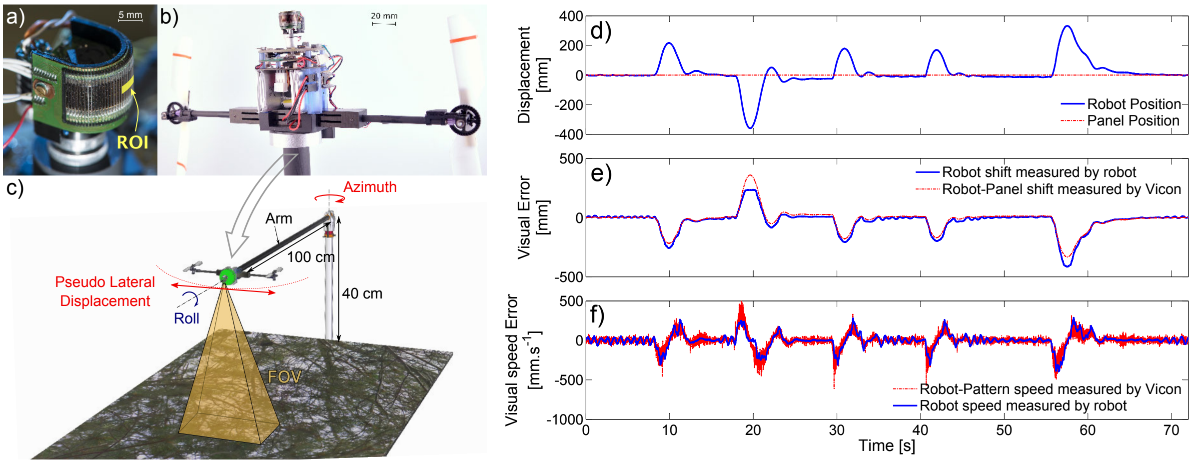


Figure B.3: a) Photo de CurvACE actif montrant la région d'intérêt utilisée pour l'expérience, composée de 40 photorécepteurs (8 × 5). (Copyright P. Psaïla) b) Photo du robot HyperRob équipé du capteur CurvACE actif. c) Setup expérimentale avec le robot fixé au bout d'un bras rotatif. Le champ de vision du capteur de 33.6° × 20.2° est présenté. d-f) Résultats du rejet de perturbations. Celles-ci sont appliquées à la main sur le bras et le robot revient automatiquement à sa position initiale. d) Déplacement du robot en bleu par rapport au panneau immobile. e) Mesure du déplacement vue par le robot en bleu et calculé à partir du système de capture du mouvement Vicon. f) Mesure de la vitesse estimé par le robot en bleu et en rouge par le système Vicon.

Le robot est ainsi capable de rejeter des perturbations en position (voir figure B.3) et de se stabiliser par rapport à un sol en mouvement. Des tests ont été effectués avec succès pour montrer la robustesse du système vis-à-vis des hypothèses avec une stabilisation au-dessus d'un terrain en pente de 18.5° et également d'un terrain avec de faibles reliefs. De plus, il a été montré qu'avec cet algorithme, le robot est capable de suivre une cible si celle-ci présente des contrastes plus élevés que l'arrière-plan.

Malgré ces belles performances, notre algorithme souffre de quelques limitations. En effet, pour des mesures locales, il n'est pas possible d'assurer une mesure monotone et linéaire par rapport à tout type d'environnements visuels. Ainsi, si l'on souhaite réduire le champ visuel, avoir une estimation fiable du déplacement avec moins de paires devient impossible. Développer une application impliquant une mesure de position de plusieurs éléments à différentes distances est également très difficile.

C Estimation de l'angle sous-tendu d'une barre afin d'obtenir une mesure linéaire de la position

Les limitations entrevues dans la partie précédentes sont en parties dues au fait que la localisation de la barre avec le traitement $VODKA$ présente un profil non linéaire en fonction de l'angle sous-tendu de la barre (c'est-à-dire l'angle que forme la barre dans le champ visuel).

Une calibration de cette caractéristique a été réalisée, à la fois en simulation et avec le capteur CurvACE actif, pour obtenir une "map" des réponses en fonction de la position angulaire et de l'angle sous-tendu.

Effectuer une transposition de la "map" obtenues permet d'avoir la position ψ en fonction de la sortie S_{vodka} et de l'angle sous-tendu α. Ainsi, en utilisant les mesures S_{vodka} de deux paires voisines, il est possible d'obtenir une estimation de ces deux paramètres. En effet, on obtient deux fonctions donnant ψ en fonction de α. En trouvant la variable α qui permet d'annuler la différence de ces deux fonctions, on résout le système et on retrouve ψ en utilisant l'une des deux fonctions.

En simulation, les estimations obtenues ont une erreur inférieure à 1°, à la fois pour l'angle sous-tendu α et pour la position angulaire ψ sur des plages relativement larges, respectivement $\psi \in [-15° : 15°]$ pour 6 photorécepteurs et $\alpha \in [5° : 9°]$ correspondant à une distance comprise dans l'intervalle $[16 : 29cm]$ pour une barre de diamètre $25mm$.

Avec le capteur CurvACE actif, la réponse de la paire centrale est quasi linéaire pour chacun des angles sous-tendus, ce qui assure une estimation fiable de la position angulaire de la barre. La précision de l'estimation de l'angle sous-tendu de la barre revient donc à la sensibilité de la mesure des paires voisines. On a donc pu observer qu'en condition statique, l'estimation était relativement fiable. Mais lors d'une expérience de tracking d'une barre en mouvement, l'estimation de l'angle sous-tendu a été assez imprécise, surtout dans les phases où celui-ci évoluait rapidement. Plusieurs hypothèses peuvent expliquer cette mauvaise estimation. Le fait que l'angle de vue change peut faire apparaitre des ombres derrière la barre, ce qui crée une dissymétrie entre les contrastes à gauche et droite de la barre. Cette dissymétrie change la réponse de chacune des paires, ce qui ne correspond plus aux conditions de calibration. La vitesse de la barre a également une influence sur la démodulation, ce qui par la suite peut entrainer une variation par rapport aux conditions de calibration.

Il apparait néanmoins qu'une telle estimation est possible, mais il faut alors des conditions bien contrôlées et un capteur ayant un bon rapport signal sur bruit. Il est également montré que l'utilisation de seulement 2 photorécepteurs est insuffisante pour estimer la position d'une barre avec son angle sous-tendu. Mais une manière plus efficace doit être trouvée pour réaliser une telle performance dans n'importe quelles conditions.

D Localisation d'un front avec 3 photorécepteurs

La phase de calibration, évoquée précédemment, était fastidieuse sur un grand champ visuel et une potentielle source d'erreur. Sur la base de ces résultats, j'ai conclu que la différence sur la somme d'une paire de photorécepteurs n'est peut-être pas la meilleure solution dans le cas d'une barre. La linéarité reste néanmoins un point à améliorer à la fois dans le cas d'une barre et dans celui d'un front.

C'est à la suite de ce constat que j'ai proposé une autre manière de procédé afin de localiser un contraste. Les travaux d'Heiligenberg sur le poisson electrique se sont avérés très intéressants. Ils montrent que la somme pondérée d'une ligne de récepteurs sensoriels ayant une sensibilité spatiale gaussienne à un stimuli est une approximation linéaire de la position de ce stimuli [Heiligenberg, 1987, Baldi and Heiligenberg, 1988]. J'ai adapté la méthode à une application visuelle afin d'être robuste aux variations de contrastes et d'intensité lumineuse. De plus, il s'avère que 3 photorécepteurs seulement sont nécessaires à la localisation d'un seul front à chaque instant. Le front étant visible uniquement par ces 3 photorécepteurs. Ainsi, le calcul de la somme pondérée normalisée (appelée NWS) pour chacun des triplets est le suivant :

$$NWS(n) = \Delta\varphi \cdot \frac{\sum\limits_{k=n-1}^{n+1} k \cdot Ph_D(k)}{\sum\limits_{k=n-1}^{n+1} Ph_D(k)} + bias \tag{57}$$

où n correspond à l'indice du photorécepteur central du triplet concerné, Ph_D le signal démodulé de chaque photorécepteurs et le biais correspond à une variable de calibration pour positionner le 0 du champ visuel.

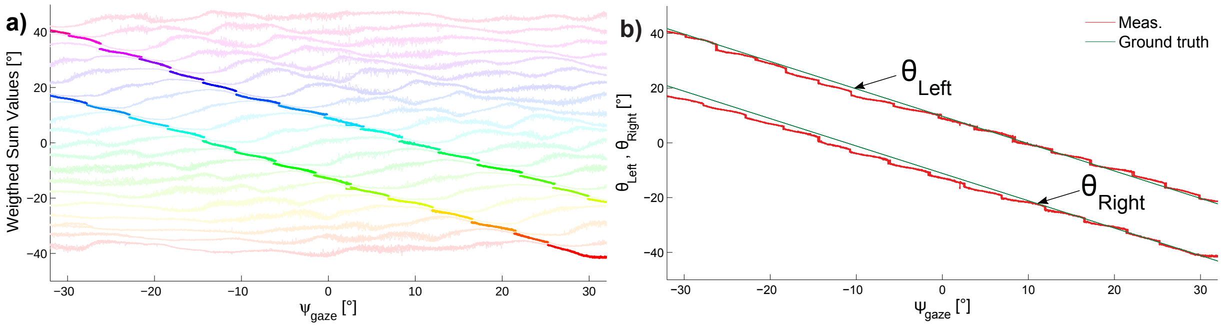


Figure D.4: a) Réponses des sommes pondérées normalisées (NWS) des 21 triplets en fontion de l'orientation de l'oeil (ψ_{gaze}) par rapport à deux front de contrastes. La partie surlignée met en avant les parties utilisée après l'étape de sélection b) Mesures fusionnées donnant la localization des deux contrastes (en rouge) par rapport à la mesure théorique (en vert).

De plus, afin de connaitre le triplet à prendre en compte, il faut assurer un critère de sélection.

Ce critère $C(n)$ est calculé pour le triplet qui voit le contraste au temps t, ainsi que ses 2 voisins et le maximum permet de connaitre le triplet à utiliser au temps $t+1$.

$$C(n) = Ph_D(n-1) + 2 \cdot Ph_D(n) + Ph_D(n+1) \tag{58}$$

Ce processus de sélection implique de connaitre la position initiale du contraste et de faire une hypothèse de continuité. C'est-à-dire que entre deux pas de temps, le contraste vue par un triplet ne peut pas avoir effectué un déplacement supérieur à $2\Delta\varphi$.

E Application à un suivi de cible

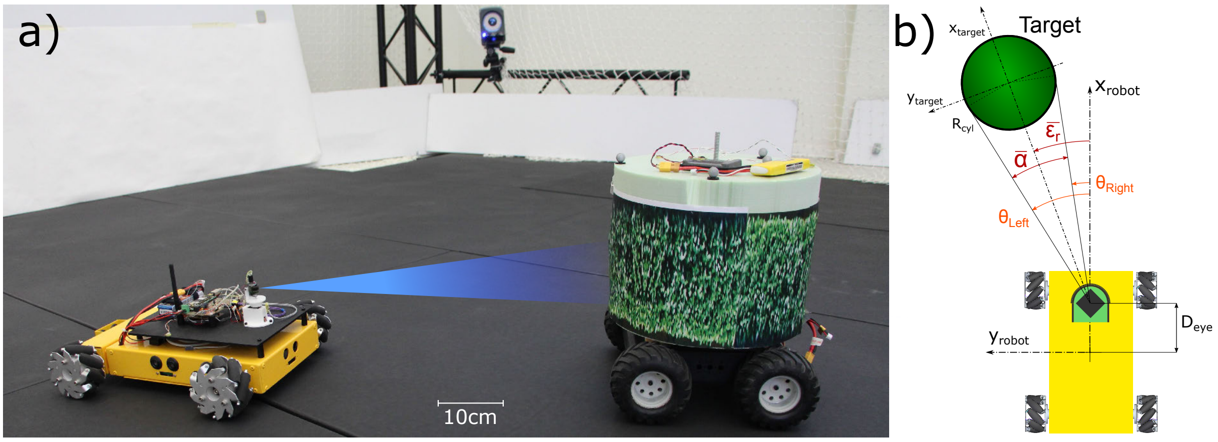


Figure E.5: a) Présentation du robot ACEbot (à gauche) suivant la cible, un cylindre texturé monté sur un rover (à droite), à une distance constante. b) Schéma de l'ensemble des paramètres utilisés dans la stratégie de commande.

Active CurvACE a ainsi été monté sur un robot nommé ACEbot afin d'effectuer une tâche de suivi de cible (voir figure E.5). Ici, la cible étant un cylindre de diamètre $32cm$ posée sur un rover. Les fronts formés par les bords du cylindre seront donc localisé grâce à l'algorithme de la somme pondérée, évoqué précédemment. Sachant que le diamètre du cylindre est connue, il est possible à partir de la position angulaire des deux contrastes d'obtenir la distance au cylindre. Un asservissement a donc été réalisé, de manière à maintenir la cible au centre du champ visuel et à une distance constante. Le suivi de la cible a été démontré sous des conditions d'éclairages variables avec une bonne précision (voir figure E.6).

On peut noter qu'aucune estimation de la vitesse de la cible n'est utilisée dans la loi de commande du robot, ce qui peut engendrer un temps de réaction assez grand lorsque la cible commence à bouger et une erreur statique en condition stabilisée.

F Comportement bio-inspiré : suivi de cible, détection et interception

La loi de commande d'ACEbot permet de maintenir la cible au centre du champ visuel ainsi qu'une distance quasi constante. En reproduisant des trajectoires de la femelle syrphe *Syritta pipens L.* observés [Collett and Land, 1975], on s'aperçoit qu'ACEbot suit une trajectoire très

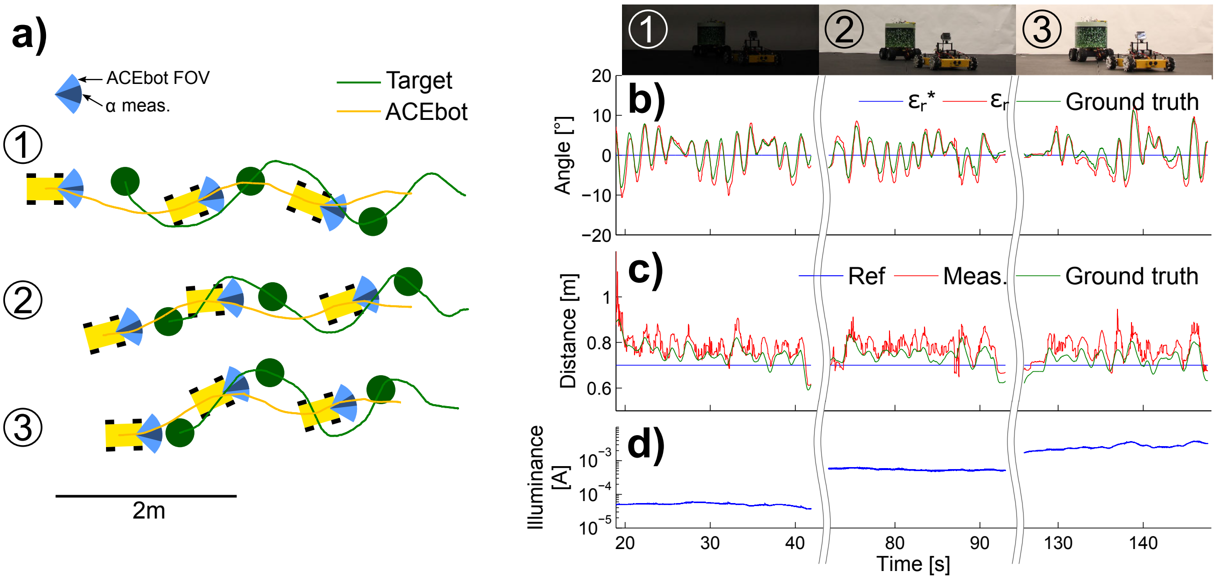


Figure E.6: Trois trajectoires de suivi de cible suivant différentes conditions lumineuses. a) Trajectoires des centres de la cible et de ACEbot respectivement en vert et orange. Les positions des robots est indiquées toutes les $8s$ avec le champ visuel du capteur en bleu ciel et l'angle sous-tendu de la cible mesurée par le capteur en bleu foncé. b) Erreur rétinienne de consigne, mesurée et calculée à partir des positions des robots en fonction du temps, respectivement en bleu, rouge et vert. c) Distance de référence, mesurée et calculée, respectivement en bleu, rouge et vert, toutes tracées en fonction du temps. d) Luminosité ambiente mesurée par un capteur de lumière embarqué sur le robot (voir fig. A.1d) pour les 3 phases. L'intensité moyenne perçue correspond à 100, 780 et $1500 Lux$.

proche de celle du mâle. Le profil de la distance à la cible en fonction du temps est lui aussi similaire. Ainsi, la stratégie adopté par le robot semble être très proche de celle du diptère. Il semble donc que ce dernier ne fait aucune estimation sur la future position de la femelle et cherche uniquement à maintenir celle-ci au centre de son champ de vision. Le maintien de la distance se fait probablement en maintenant l'angle sous-tendu de la cible constant. Mais cette hypothèse n'est pas évidente si l'on tient compte du fait que la femelle n'est pas une boule.

Une expérience a également été menée avec un œil découplé afin de bénéficier d'une dynamique plus rapide de celui-ci comparé au robot. Cependant, la précision de la mesure est très affectée par des mouvements rapides. En effet, l'adaptation à la lumière, intrinsèque au capteur, ainsi que les différents filtres temporels, ne permettent pas une mesure de position correcte en étant soumis à des mouvements rapides. C'est pourquoi un contrôle saccadé de l'œil a été préféré, une inhibition a été intégrée afin de ne pas bouger l'œil le temps que les filtres convergent de nouveau à l'image des saccades observées chez le singe ou la mante religieuse [Kirschfeld, 1994]. Mais, ce temps d'inhibition étant trop grand, il ne permet pas d'augmenter suffisamment la vitesse moyenne de rotation et ne présente donc que peu d'avantages dans notre application.

Durant cette thèse, j'ai aussi travaillé sur la capacité à détecter une cible en mouvement avant de la suivre. Sachant que l'algorithme de localisation nécessite de connaître la position initiale des photorécepteurs, cela permet d'éviter une initialisation manuelle. La stratégie utilisée s'appuie sur le fait que chaque photorécepteur est sensible à une variation de luminosité. Ainsi,

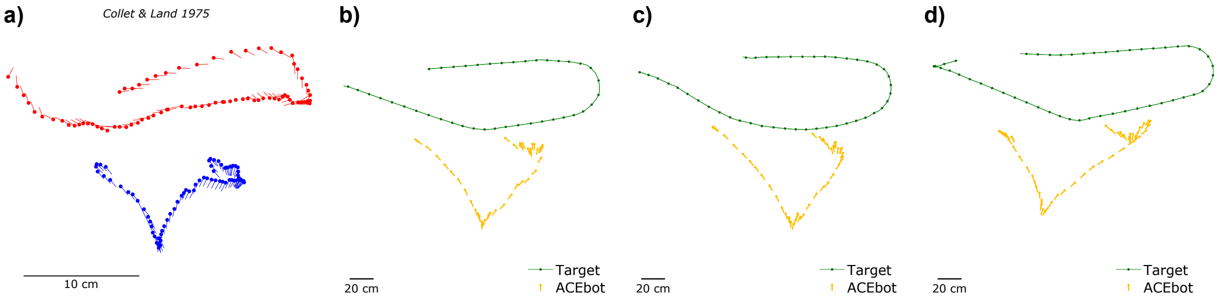


Figure F.7: Comparaison entre une trajectoire observée dans la phase d'accouplement des syrphes *Syritta pipens L.* (a) avec celles de suivi de cible d'ACEbot (b, c et d). a) Les positions de la femelle et du mâle sont notées respectivement en bleu et rouge avec un trait pour indiquer l'orientation du corps. L'écart temporel entre 2 positions est de $40ms$. b-d) En vert, la trajectoire de la cible et en jaune, celle de ACEbot avec son orientation. Les positions sont tracées toutes les $400ms$.

lorsque la cible bouge à une vitesse suffisante devant le capteur et que deux contrastes sont clairement identifiés, alors la poursuite peut démarrer. Un filtrage passe haut est tout de même réalisé afin de supprimer la composante continue et un filtrage passe-bas afin de réduire le bruit de mesure. Cette stratégie est proche des observations faites notamment sur la 'mouche tueuse' (*Coenosia attenuata*), qui montrent que leur probabilité de décollage est corrélé avec le rapport de la vitesse de la cible sur sa taille. En effet, si la cible est trop loin alors la vitesse perçue est insuffisante pour obtenir un signal dépassant le seuil fixé et de plus son angle sous-tendu ne dépassera $2\Delta\varphi$. Il faut donc que la cible soit à proximité et avec une vitesse suffisante.

De plus, nous avons également pu montrer qu'en modifiant la consigne de distance, il est possible d'effectuer une interception de la cible. Une variation de l'angle de l'œil permet aussi d'optimiser la distance parcourue avant interception. Un travail plus important est néanmoins nécessaire afin de reproduire les trajectoires plus évolués de la libellule.

G Conclusion

Cette thèse présente des travaux utilisant un œil composé soumis à une vibration périodique lui permettant de localiser des contrastes avec une hyperacuité. Il a ainsi été possible à un robot aérien de se stabiliser au-dessus d'un panneau texturé avec une grande précision. Dans le cas d'une barre, la localisation est plus complexe et l'estimation de l'angle sous-tendu grâce à une calibration permet d'améliorer la linéarité. Mais, la mesure de celui-ci reste imprécise quand la cible bouge et que les conditions diffèrent des conditions de calibrations. J'ai ainsi proposé d'améliorer la linéarité de la réponse du traitement tout en évitant un recours à une étape de calibration. Un nouvel algorithme basé sur une somme pondérée a été mise au point dans le cas d'un front. Cette technique a été appliquée dans le cas d'un suivi de cible avec le maintien d'une distance constante avec celle-ci. Les trajectoires du robot se sont avérées similaires à celle du syrphe lorsque celui-ci suit une femelle. La stratégie du maintien de la cible au centre du champ visuel sans recourir à une estimation de la vitesse de la cible semble donc être la solution adoptée par le diptère.

Finalement, il a été montré que la vibration peut être utilisée pour améliorer l'acuité visuelle d'un œil composé artificiel. Cette technique a montré un grand intérêt car elle permet de localiser un contraste avec précision sans recourir à un traitement gourmand en ressources calculatoire. Mais son application dans des environnements plus complexes reste délicate. Reprendre les expérimentations sur la mouche permettrait de mieux comprendre les situations où elle fait appel à cette possibilité. La façon dont elle traite l'information visuelle reste vraisemblablement difficile à obtenir mais utiliser une autre méthode de démodulation pourrait permettre d'obtenir la différenciation des contrastes ON-OFF inexistante aujourd'hui. Il reste encore beaucoup à découvrir quant à l'utilisation de ces micro-mouvements rétiniens, à l'image de ceux observés chez l'homme où leur utilité est encore sujet à débat [Rolfs, 2009].

Hyperacute artificial compound eye: Robotic applications to stabilization and pursuit

ABSTRACT: Based on several studies, the fly retina is submitted to periodic micro-movements. Several sensors were designed in the lab mimicking this principle. It was therefore established that this vibration could be used in the localization of contrasts with precision and endowed the fly vision with hyperacuity. Inspired by the fly compound eye properties, such as the Angular sensitivity and the periodic scanning, the sensors were able to localize a contrast very precisely over a small field of view limited to only two pixels.

In this thesis, an artificial compound eye endowed with a wide field of view was used. First, an algorithm that fused the local position measurements of different photosensor (pixel + lens) pairs is proposed. It enables a robot named HyperRob to hover above a naturally textured pattern.

Localizing a contrast precisely over the entire field of view remains difficult with this first solution. But, using 2 pairs of photosensors, a second algorithm allows having, in the case of a bar, a more linear position measurement and its subtended angle too. A calibration process was involved to have a map of the pair measurements relative to the angular position and subtended angle of the bar. It showed some good results, especially in steady conditions, but also a dependency on the contrasts seen and the illuminance in respect to the calibration setup.

Therefore, an effort was done in order to avoid a calibration process. A third algorithm was suggested using previous works of Heiligenberg and Baldi. They established that an array of Gaussian receptive field can provide a linear estimation of a stimulus position, thanks to a weighted sum calculation. Here, this approximation is modified to be robust to ambient lighting and contrast variations. An application to a target pursuit was made with a mobile robot named ACEbot. It was able to reproduce pursuit behavior similar to the hoverfly. An interception behavior was also showed.

Finally, an artificial compound eye with a coarse spatial resolution can be endowed with hyperactuity and enable a robot to follow a target with precision. In this thesis, a step forward has been made toward bio-inspired target localization and pursuit, allowing a better understanding of the strategy used by winged insects.

KEYWORDS: Bio-Inspiration, Vision, Hyperacuity, Robotics, Stabilization, Pursuit, Interception

Œil composé artificiel doté d'hyperacuité : Applications robotiques à la stabilisation et à la poursuite

RÉSUMÉ: Plusieurs études ont montré que la rétine de la mouche est soumise à des micro-mouvements périodiques. Différents capteurs inspirés par ces observations ont été réalisés au laboratoire. Ils ont pu démontrer que ces vibrations pouvaient être utiles dans une tâche de localisation de contrastes avec précision et ainsi doter la mouche d'une hyperacuité visuelle. Inspirés par les propriétés optiques des yeux composés, les capteurs visuels reproduisaient la sensibilité angulaire gaussienne et la vibration périodique de l'œil de la mouche pour localiser un contraste très précisément avec un champ visuel réduit à deux pixels.

Dans cette thèse, un œil composé artificiel programmable appelé CurvACE, doté d'un large champ de vision, est utilisé. Tout d'abord, un nouvel algorithme permet de fusionner les mesures visuelles de position des contrastes issus de différentes paires d'ommatidies (pixel et lentille). Grâce à cet algorithme, le robot HyperRob est capable de se stabiliser au-dessus d'un panneau texturé.

Localiser un contraste de manière linéaire sur l'ensemble du champ visuel demeure toutefois difficile avec cette première solution. Ainsi, en utilisant 2 paires de photorécepteurs, un deuxième algorithme permet de localiser une barre de façon plus linéaire en ajoutant la mesure de l'angle sous-tendu. Cet algorithme implique une calibration de l'ensemble des réponses de chaque paire en fonction de la position angulaire et de l'angle sous-tendu de la barre. De bons résultats sont obtenus, surtout en statique, malgré une certaine dépendance aux contrastes et à la luminosité ambiante vis-à-vis des conditions de calibrations.

Ainsi, afin d'éviter un processus de calibration, un troisième algorithme qui s'appuie sur les travaux d'Heiligenberg et Baldi, a été proposé. Ces auteurs ont montré que la somme pondérée de plusieurs capteurs ayant un champ récepteur gaussien pouvait fournir une estimation linéaire de la position d'un stimulus. Nous avons, pour la première fois, appliqué une variante de ce principe à un œil composé artificiel afin d'être robuste aux variations de luminosité ambiante et de contraste. Une application à la poursuite d'une cible a été effectuée avec un robot mobile nommé ACEbot. Nous avons pu reproduire un comportement similaire à celui observé chez le syrphe. Le robot a aussi montré qu'il était capable d'effectuer des manœuvres d'interceptions.

Finalement, un œil composé artificiel dont la résolution intrinsèque est faible, peut être doté d'une hyperacuité visuelle et permettre de suivre une cible avec précision. Ces travaux ont ainsi conduit à proposer des stratégies bio-inspirées pour la localisation et la poursuite de cible et de mieux comprendre les stratégies utilisées par les insectes ailés.

MOTS CLÉS: Bio-Inspiration, Vision, Hyperacuité, Robotique, Stabilisation, Suivi de cible, Interception