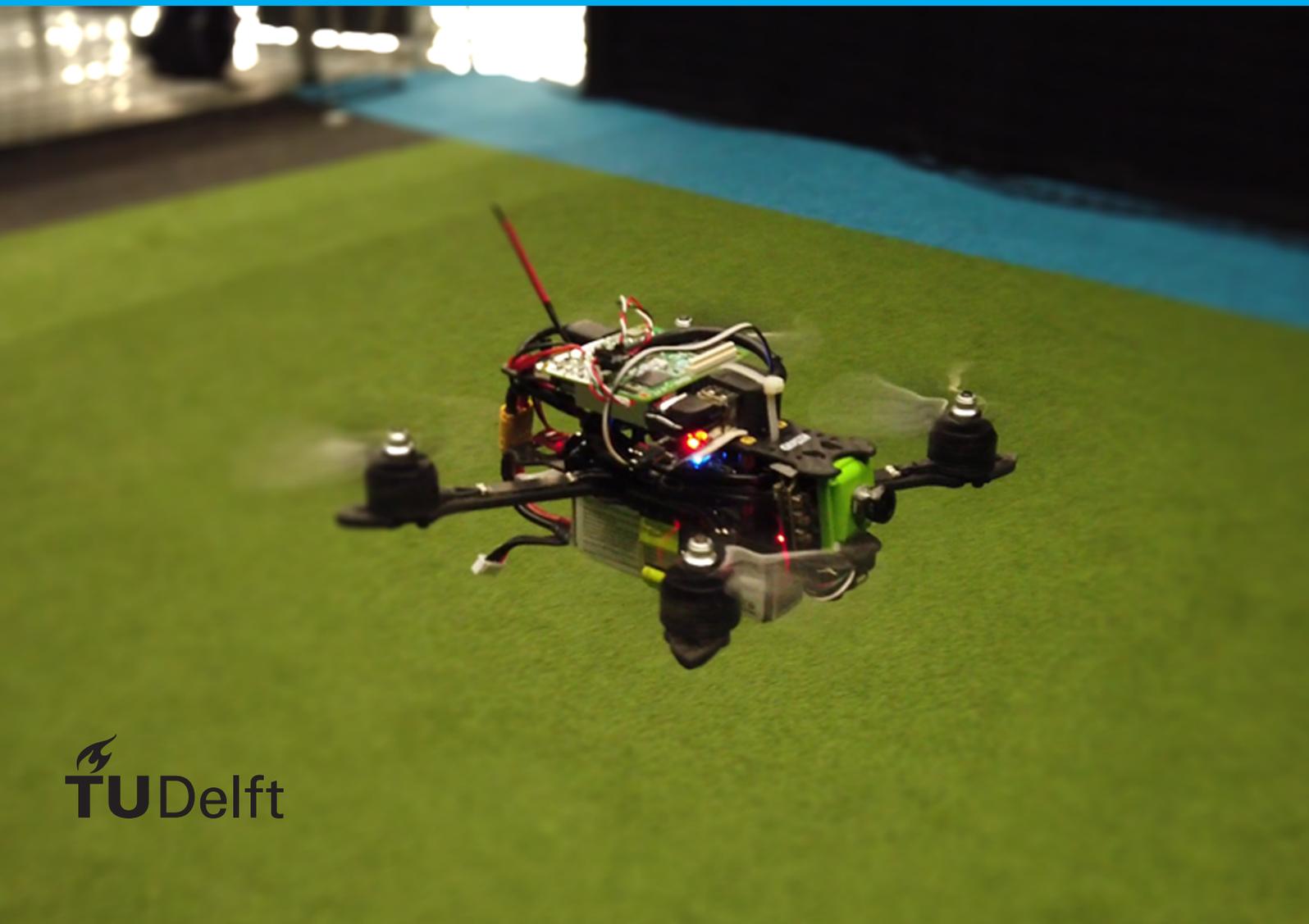# Obstacle Detection and Avoidance onboard an MAV using a Monocular Event-based Camera

MSc. Thesis

R.M. Dinaux

Supervised by: dr. J. Dupeyroux and Prof. G. de Croon

TUDelft

Obstacle Detection and Avoidance onboard an MAV using a Monocular Event-based Camera

MSc. Thesis

by

R.M. Dinaux

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday April 7, 2021 at 14:00.

Student number:	4298470	
Thesis committee:	Dr. J. Dupeyroux,	TU Delft, supervisor
	Prof. dr. ir. G. de Croon,	TU Delft, supervisor
	Dr. J. F. P. Kooij,	TU Delft, examiner

TUDelft

Executive Summary

Micro Air Vehicles (MAVs) are able to support humans in dangerous operations, such as search and rescue operations at night on unknown terrain. These scenes require a great amount of autonomy from the MAV, as they are often radio and GPS-denied. As MAVs have limited computational resources and energy storage, onboard navigation tasks have to be performed efficient and fast. To address this challenge, this research proposes an approach to visual obstacle detection and avoidance onboard an MAV. The algorithmic approach is based on event-based optic flow, using a monocular event-based camera. This camera captures the apparent motion in the scene, has microsecond latency and very low power consumption, therefore a good fit for onboard navigation tasks. Firstly, a literature study is performed to provide theoretical concepts and foundation for the obstacle avoidance approach. A processing pipeline is designed, based on the use of event-based normal optic flow. This pipeline consists of three sections: course estimation, obstacle detection and obstacle avoidance. A novel course estimation method 'FAITH' is proposed which uses optic flow half-planes along with a fast RANSAC scheme. The object detection method is based on DBSCAN clustering of optic flow vectors, using the time-to-contact and vector location as clustering variables. The performance of these methods is experimentally demonstrated by three experiments: in a simulated environment, offline on real sensor data and online onboard an MAV. As currently no event-based obstacle avoidance datasets are publicly available, a dataset is recorded as supplement to this and future research. Approximately 1350 runs of event-based camera, RADAR, IMU and OptiTrack data are recorded, manually avoiding either a single or two poles using an MAV in the flying arena of the TU Delft. This dataset is used in this research to determine the performance of the course estimation method using real sensor data. The course estimation method is shown to have state-of-the-art accuracy and beyond state-of-the-art computation time on both simulated data and the recorded dataset. The final experiment shows the obstacle detection and avoidance approach integrated onboard an MAV in a real-time obstacle avoidance task. The approach is shown to have a success rate of 80% in a frontal obstacle avoidance task on a low-textured 50-cm wide pole. The contribution of this research is an obstacle detection and avoidance approach using a monocular event-based camera onboard an MAV, along with the novel course estimation algorithm 'FAITH'.

List of Abbreviations

ANN   Artificial Neural Network
APS   Active Pixel Sensor
CIE   Commission Internationale de l'Eclairage
CMOS   Complementary Metal Oxide Semiconductor
CNN   Convolutional Neural Network
CPU   Central Processing Unit
DAVIS   Dynamic and Active-pixel Vision Sensor
DBSCAN   Density Based Spatial Clustering of Applications with Noise
DCMD   Decensing Contralateral Movement Detector
DS   Directionally Selective
DVS   Dynamic Vision Sensor
ECN   Evenly-cascaded Neural Network
EMD   Elementary Motion Detection
FAITH   Fast Iterative Half-plane
FOC   Focus of Convergence
FOE   Focus of Expansion
FOV   Field of View
FPGA   Field-Programmable Gate Array
FPS   Frames Per Second
GPU   Graphics Processing Unit
IMAV   International Micro Air Vehicle Conference and Competition
IMU   Inertial Measurement Unit
LGMD   Lobula Giant Movement Detector
LGN   Lateral Geniculate Nucleus
LIDAR   Light Detection And Ranging
LMC   Large Monopolar Cells
LPTC   Lobula-Plate Tangential Cell
MAV   Micro Air Vehicle
NN   Neural Network
OF   Optic Flow
PD   Proportional Derivative
RANSAC   Random Sample Consensus
RGB-D   Red Green Blue – Depth
ROS   Robot Operating System
SNN   Spiking Neural Network
SNR   Signal to Noise Ratio
SOC   System on a Chip
SWAP   Size Weight And Power
TTC   Time-To-Contact
VSLAM   Visual Simultaneous Localization And Mapping

Contents

1

Introduction to Research

Autonomous navigation for mobile robots, including tasks such as obstacle avoidance, is considered one of the top ten technological challenges of our time [13]. These challenges occur in both ground and areal robotics. Specifically, Micro Air Vehicles (MAVs) are gaining traction in a variety of applications. MAVs are able to fly in challenging environments, due to their agile movement and small size. For example, when considering a disaster aftermath or a fire in a building with an unknown map, it is often considered dangerous to send humans into the scene. A fully autonomous MAV is able to enter and navigate in the struck area, providing a safer and faster alternative to sending in humans. Unfortunately, MAVs are endowed with highly restricted power capacity, and extremely limited computational resources. Their use is also hampered by the risk of GPS failure indoor, magnetometer disturbances because of surrounding ferrous materials (e.g., buildings or infrastructure) and IMU drift over time. The Size, Weight and Power (SWAP) of the MAV are therefore highly restricted. As visual sensors are light-weight and more energy efficient than active sensors, they are often used onboard MAVs for visual odometry and navigation. MAVs are often equipped with cameras for which both the temporal ($30 - 60$ fps on average) and the visual resolutions are limited. These embedded cameras have greatly contributed to the reduction of navigation failure, but their use remains limited by the low computational resources available onboard. It is therefore crucial to determine fast and efficient methods to allow MAVs to autonomously navigate and avoid obstacles.

The recent developments in neuromorphic systems represent a promising opportunity for autonomous obstacle avoidance and navigation onboard robots, in particular for MAVs. In this respect, event-based cameras were first released in 2008 by Lichtensteiner et al. [8]. Unlike conventional cameras which output images at a fixed frame-rate, event-based cameras produce a stream of asynchronous and independent events reporting changes in brightness at the pixel level. Therefore, these cameras inherently capture the apparent motion. The intensity change threshold, which triggers the pixel, is user-defined. Event-based cameras offer a high dynamic range ($> 120 \, dB$) along with a high temporal resolution (in the range of microseconds). These advantages make event-based cameras inherently insensitive to classical visual artifacts such as motion blur or the tunnel effect. As a result, these cameras provide accurate visual information at extremely high speed, making them suitable for aerial robotics, including MAV. These cameras can therefore be used for various tasks including obstacle avoidance. The characteristics of this camera are discussed extensively in Appendix A, Chapter 4.1.

This research is conducted in the context of the preparation for the International Micro Air Vehicle Conference and Competition (IMAV) in November 2020, which is postponed due to the COVID-19 crisis to November 2021. In this competition, international research teams address challenges, pushing the boundaries of MAV technology currently available. This edition focuses on 'search and rescue' scenarios, such as described above. One of the competition challenges, which is used as inspiration for this research, requires the MAV to fly through a room filled with smoke and moving poles. This challenging environment pushes the capabilities of autonomous MAV navigation. To narrow down the scope of the research, the primary focus of this research is on static obstacle avoidance. This thesis provides a report of the design of a visual obstacle detection and avoidance system on an MAV, using a monocular event-based camera with fully onboard processing. As visual course estimation is a crucial process in visual obstacle avoidance tasks, this research also provides a novel solution to this open challenge.

The central research question of this thesis is:

> What approach can be used to perform monocular event-based obstacle detection and avoidance onboard an MAV?

The main research objective and contribution is as follows:

> To contribute to the development of obstacle avoidance systems for MAVs using only onboard processing; by providing an obstacle detection and avoidance system using a monocular event-based camera, along with a novel course estimation algorithm.

The following structure is used in this thesis report. The research paper on the novel course estimation method is the main body of content. The course estimation, obstacle detection and obstacle avoidance methods are introduced in Chapter 2. This chapter discusses the methodology and system design choices. After this brief introduction to the sub-systems, the paper is presented in Chapter 3. Afterwards, Chapter 4 draws a final conclusion on the outcome of the research thesis. Lastly, recommendations for future research are given in Chapter 5.

Appendix A contains the literature review, which is conducted before the start of the preliminary experiments. This literature review discusses relevant previous research in depth and is used as theoretical foundation for this research. The first section of the literature study reviews many biological concepts regarding visual obstacle avoidance. During the preliminary experiments, the research focus shifted from a bio-inspired design to an algorithmic approach as this showed promising results. Therefore, the second section of the research sub-questions in the literature study are of higher relevance to the final research outcome. It is recommended to read the synthesis of the second half of the literature study as supplementary information on the system design choices made.

2

Introduction to Methodology and System Design

Performing visual obstacle detection and avoidance is a task with a high level of abstraction. To achieve autonomous obstacle avoidance in this research, several sub-systems are designed which interpret visual cues and fulfill lower-level functions. This chapter gives background information on the assumptions and decisions made in the design of these sub-systems. The proposed novel course estimation method is summarized in a scientific paper, which uses the obstacle detection and avoidance sub-systems to demonstrate its performance onboard an MAV. This chapter therefore gives an introduction to the relevant topics discussed in the scientific paper. Firstly, three important considerations regarding the sub-system design, resulting from the literature study and preliminary experiments, are discussed.

- **Primary obstacle**
  Firstly, the literature study and first experiments with the event-based camera give insight into the capabilities and limitations of the sensor. Due to the complex task of visual obstacle avoidance and the small amount of previous research on event-based solutions to this task, this research focuses on high-contrast obstacles. An example of such an object is an orange pole in the flying arena at the TU Delft. In line with the research challenges discussed in the research introduction (Chapter 1), a static 50-cm wide pole is chosen as primary obstacle to avoid for this research.

- **Algorithmic approach**
  Secondly, the research takes an algorithmic approach to the obstacle avoidance task. The literature study showed many learning-based approaches using (Convolutional) Neural Networks, Spiking Neural Networks (SNN) and others. These solutions often require heavy computational resources, which are highly restricted onboard. Therefore, these methods do not fit the design requirements of this research and a lightweight algorithmic approach is taken.

- **Sub-systems**
  The third consideration regards the visual cues used in the obstacle avoidance task. The literature study shows that optic flow is a commonly used visual property, which allows monocular visual systems to extract information about the environment and perform visual odometry. Therefore optic flow is used as the primary visual source of information. The event-surface method for estimating optic flow by Benosman et al. [1] shows robust performance and generates sparse normal optic flow. This method is selected as most suitable for this research, due to its robustness and availability. The literature study also concluded that previous research on visual course estimation (i.e., based on the focus of expansion) using sparse normal flow is limited. Therefore a comparison between different methods is made and a novel method is proposed. Using this proposed method, a clustering-based obstacle detection sub-system is designed. Lastly, a straightforward obstacle avoidance method using the detected obstacle location and MAV course is designed.

These considerations are taken into account in the obstacle detection and avoidance system design. The following section will discuss the methodology used to design the sub-systems. Afterwards, Section 2.2 discusses additional background on the sub-system design assumptions and decisions.

2.1. Methodology

Several sub-systems are designed to perform lower-level functions in the obstacle avoidance task. To systematically design and verify these sub-systems, each is subject to the same methodology. This methodology consists of four parts:

1. **Consult literature study and select method(s)**
   Firstly, the previous research on the topic of the sub-system is reviewed. From this previous research, the method which suits the system design requirements best is selected, to be implemented and tested in MATLAB. In the course estimation sub-system design, no clear best solution can be selected from the available methods. A novel method is proposed and benchmarked against three state-of-the-art methods. This focus of expansion estimation method and its performance benchmark is extensively described in the scientific paper (Chapter 3).

2. **Perform simulated experiments**
   With the selected method for the sub-system, simulated experiments are performed to review the performance of the chosen method. An event-based camera simulator ESIM [9] is used and multiple scenes are designed to test the methods offline in a controlled environment. These simulated experiments allow to fully control the experiment variables (e.g. the environment and MAV trajectory). This validates whether the chosen concept for this sub-system is suitable for implementation.

3. **Perform experiments on dataset using real sensors**
   After validating the performance of the sub-system on simulated data, its performance is reviewed using a dataset containing real sensor data. This experiment shows whether the chosen method performs sufficiently on data from a physical sensor. This data originates from a dataset which is recorded as supplementary material to this project and future research. It consists of ~ 1350 MAV obstacle avoidance trials of event-based camera, RADAR, 1920x1080 frame-based video, IMU, and OptiTrack data. Section 2.2.2 provides additional information on this dataset.

4. **Perform real-time onboard experiments**
   Lastly the method is implemented in C++ using the Robot Operating System (ROS) framework [11]. It is added to the final processing pipeline and tested in an obstacle avoidance task. This experiment shows whether the chosen method is implementable onboard an MAV and is able to run real-time.

This concludes the methodology for this research. The considerations from the literature study and the four steps described above are applied to each sub-system. This results in a systematically verified and validated obstacle avoidance system. The following section will discuss the design considerations of each sub-system separately.

2.2. System Design

As described in the introduction to this chapter, optic flow is used as the primary source of information about the environment and the ego-motion of the MAV. Following the algorithmic approach, four sub-systems based on optic flow are designed. Figure 2.1 shows this processing pipeline in a flow diagram. Firstly, the optic flow is generated and derotated using the rotational rates from the inertial measurement unit (IMU). Secondly, the course of the MAV is determined using the FAITH method to estimate the focus of expansion (FOE). Afterwards, the Time-To-Contact (TTC) is estimated and used to cluster the optic flow into object and background clusters. Lastly, these clusters are used with the FOE estimation to determine a roll command, which is given as input to the MAV control loop. The four sub-systems are discussed as optic flow estimation (Section 2.2.1), focus of expansion estimation (Section 2.2.2), the obstacle detection (Section 2.2.3) and obstacle avoidance strategy (Section 2.2.4).

2.2.1. Optic flow

The results of the literature study (Appendix A, Section 4.2.2) show that optic flow is a broadly used principle in visual obstacle avoidance systems. It also shows that, when using a monocular camera, motion based visual depth cues can be used to infer relative distances. As SWAP limitations of the MAV platform restrict the design to using a monocular camera, motion based cues are most suitable for this task. Therefore, event-based optic flow is used to derive visual depth cues. The event-based camera lends itself well for the estimation of apparent motion, as it inherently detects brightness change on the pixel-level. Following the conclusions from the
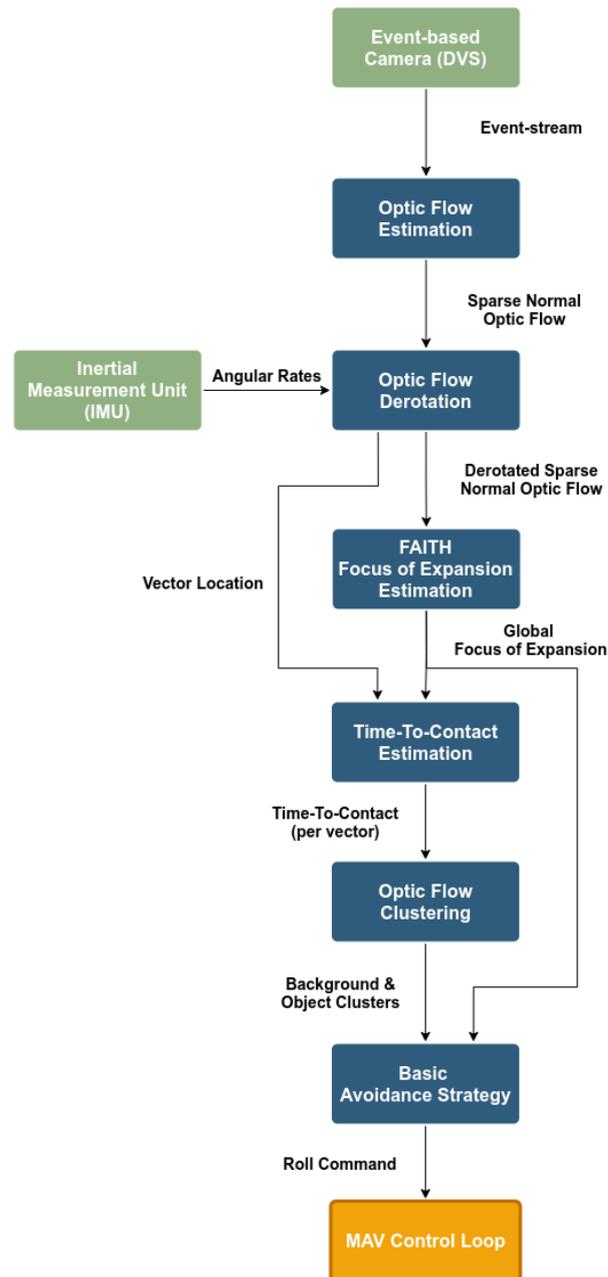
Figure 2.1: Processing flow for the obstacle avoidance system. Green elements are sensors onboard the MAV, blue elements are processing steps and the yellow element is the MAV control loop. The arrows with labels show the output from functions and their connection to other functions.
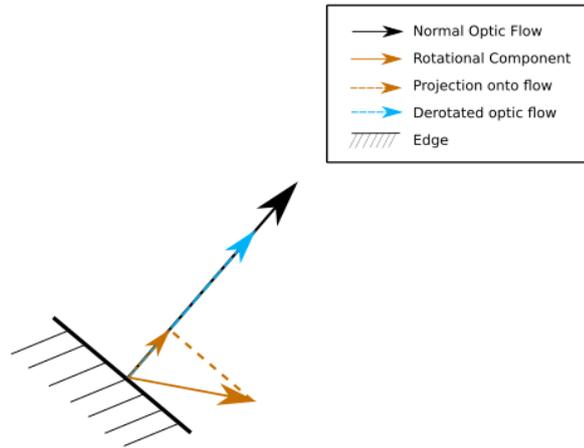
Figure 2.2: Derotating the normal optic flow, using the rotational compontent of the optic flow. This rotational component is estimated using the angular rates from the IMU. The rotational component is projected and subsequently subtracted from the normal optic flow, resulting in derotated normal optic flow.

synthesis of the literature study, an event-surface method is used in this research. This method, proposed by Benosman et al. [1], uses the spatio-temporal representation of the event cloud. By estimating the parameters of the slope of a group of events, the method is able to estimate sparse normal optic flow. This optic flow is normal to edges (normal optic flow) as the aperture problem limits the estimation to local optic flow estimation. As described in the literature review, optic flow consists of a translational and rotational component. As only the translational component contains useful depth cues, the rotational component is filtered out using the angular rates from the inertial measurement unit (IMU). The normal optic flow is derotated by using the projection of the rotational component onto the normal optic flow vector. This projection of the rotational component is subtracted from the normal optic flow vector (Fig. 2.2). The resulting derotated optic flow is used as input for the focus of expansion sub-system, discussed in the following section.

2.2.2. Focus of Expansion

Visual odometry is a crucial process in autonomous visual obstacle avoidance. The egomotion of the MAV, relative to the object, determines whether the MAV is on a collision course. This research uses a property of optic flow, the focus of expansion (FOE), to determine the course of the MAV. As described in the literature study (Appendix A, Section 4.2.3), this is a singular point from which the optic flow expands, assuming the scene is static and the motion of the observer is purely translational (Fig. 2.3). Therefore, the optic flow at this point is zero and indicates the course of the MAV under these assumptions. Estimating the FOE is a challenging task as optic flow estimation can be noisy, sparse and often only generates normal optic flow. The restricted computational resources of the MAV also highly limit the amount of processing which can be used for this visual odometry task.

In this study, the novel FAITH (FAst ITerative Half-plane) method is proposed to determine the course of the MAV. This is achieved with a fast RANSAC-based algorithm which determines the FOE based on optic flow half-planes. The performance of the proposed method is demonstrated using three methods, following the methodology described in Chapter 2. For the sake of experimental simplification, the motion of the MAV is bounded to translation in the horizontal plane (Fig. 2.4) and within the camera FOV. Firstly, the proposed method is benchmarked against three recent state-of-the-art focus of expansion implementations in a virtual environment, simulated with the ESIM event-based camera simulator [9]. Secondly, these methods are benchmarked on real sensor data using the dataset collected in the indoor flying arena, equipped with an OptiTrack system. Lastly, the proposed method is implemented onboard an MAV and tested real-time in an obstacle avoidance task.

Simulated experiment

The methods are first benchmarked in a simulated environment such that the experiment parameters (e.g., the MAV trajectory) are fully controlled. Figure 2.5 shows a comparison between simulated event-based camera data and real sensor data. Although the ESIM contains simulated Gaussian noise on the contrast threshold, the simulated event-stream still shows a higher amount of texture (e.g., on the steel pillars in Fig. 2.5-B)

Figure 2.3: Example of optic flow from straight motion towards a tree in simulation. The blue arrows represent optic flow vectors. The red cross indicates the focus of expansion. Image adapted from G. de Croon [4].
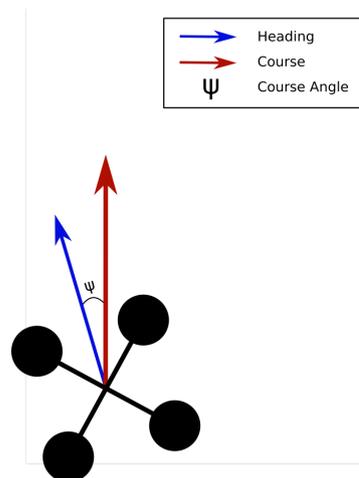

Figure 2.4: Definition of the course angle in the MAV body reference frame. The blue arrow represents the heading of the MAV, the red arrow the course and ψ the angle between these vectors.
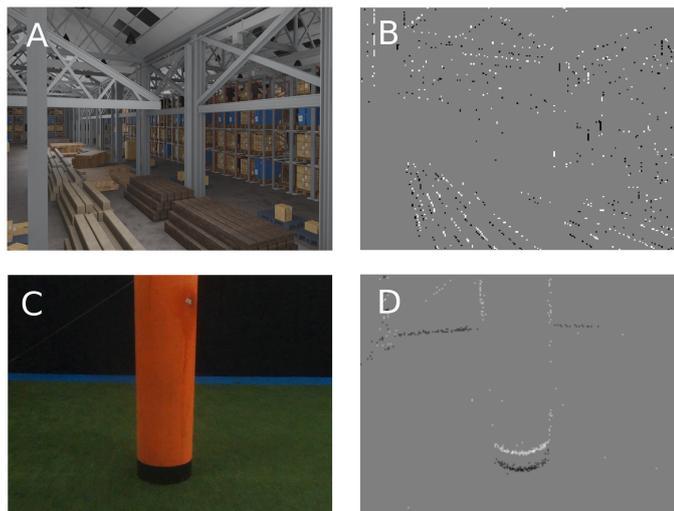
Figure 2.5: Comparing the simulated DVS output from the ESIM versus real output from a DVS240C on-board an MAV: **(A)** Simulated environment of a wood warehouse in Blender, **(B)** Simulated event-image from output of ESIM, **(C)** Image of a pole in the flying arena, captured by the video camera onboard the MAV, **(D)** Event-image from the output of a real DVS240C onboard the MAV.

compared to the real sensor data. This also results in a higher density of optic flow vectors, as the used event-surface method is more effective with higher texture. Therefore, the proposed FOE estimation method is also benchmarked on real sensor data and onboard the MAV to show performance in realistic conditions.

Following the simulation experiment workflow (Fig. 2.6), firstly four 3D environments are generated in the open-source software Blender [3]. These environments represent different scenarios in which the FOE estimation method is tested. The MAV trajectories, which are also generated in Blender, are chosen such that FOE covers a widespread selection of course angles in the field of view ($-30°$ to $30°$). These include straight and sinusoid trajectories to test the robustness of the methods. Next, the 3D model and trajectories are fed into the simulator, which generates a ROS bag containing the simulated event-stream. This ROS bag is given as input to the event-surface optic flow estimation method, which outputs a .csv file with optic flow vectors and timestamps. The FOE estimation methods are implemented in MATLAB and temporal slices of optic flow are given as input to this implementation. This results in an FOE estimation for each time-slice. With the ground truth trajectory from Blender and the FOE estimations, a course estimation error is calculated. For four different environments, 100 trajectories are simulated which lead to the results presented in the research paper (Chapter 3).

Dataset experiment

The simulated experiment uses an event-based camera simulator to test the performance of the FAITH method offline in a controlled environment. The results from these experiments show state-of-the-art performance. To determine the performance on real event-based camera data, it is tested on a dataset which is recorded as supplementary material to this thesis. As no datasets on MAV obstacle avoidance using an event-based camera are publicly available, a new dataset was recorded for the purpose of this and future research. The research project of graduate student Nikhil Wessendorp [12] is closely related to this research, by using a RADAR instead of an event-based camera for a similar obstacle avoidance task. Therefore, this dataset is recorded in collaboration with his work. An MAV platform is equipped with an event-based camera (DVS240), a 24-GHz radar sensor, a Full-HD RGB camera and a 6-axes IMU. The obstacles in the flying arena of the TU Delft consist of one or two 50-cm wide poles, of which the ground truth location is known. The MAV is manually flown, avoiding the obstacles, while the ground truth attitude and location is recorded using the OptiTrack system. Approximately 1350 runs are recorded, after which the data from all sensors is synchronised and stored in ROS bags. Figure 2.7 shows an example of a sample from the dataset. The specifications of this project are published in a dataset paper, and the data will be publicly available on the 4TU servers. Access to the dataset and paper can be gained via the GitHub repository [6].

As the event-based camera output and ground truth MAV attitude and location are recorded in this dataset, it is used to test the FAITH method performance offline on real sensor data. Similar to the simulated exper-
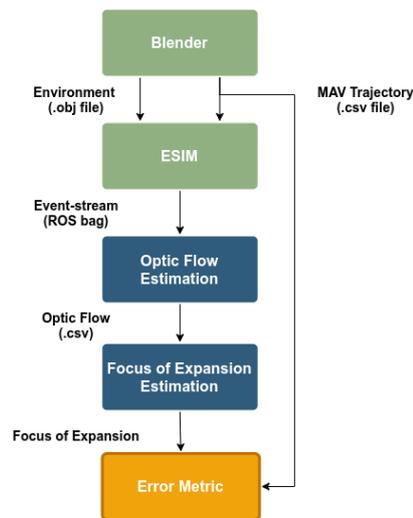
Figure 2.6: The workflow for testing the four FOE estimation methods in the simulated environment. The 'ESIM: Open Event Camera Simulator' [9] is used on a 3D rendered environment and MAV trajectory from Blender. Afterwards, the event-surface method is used to estimate optic flow and the four methods are tested on this output.
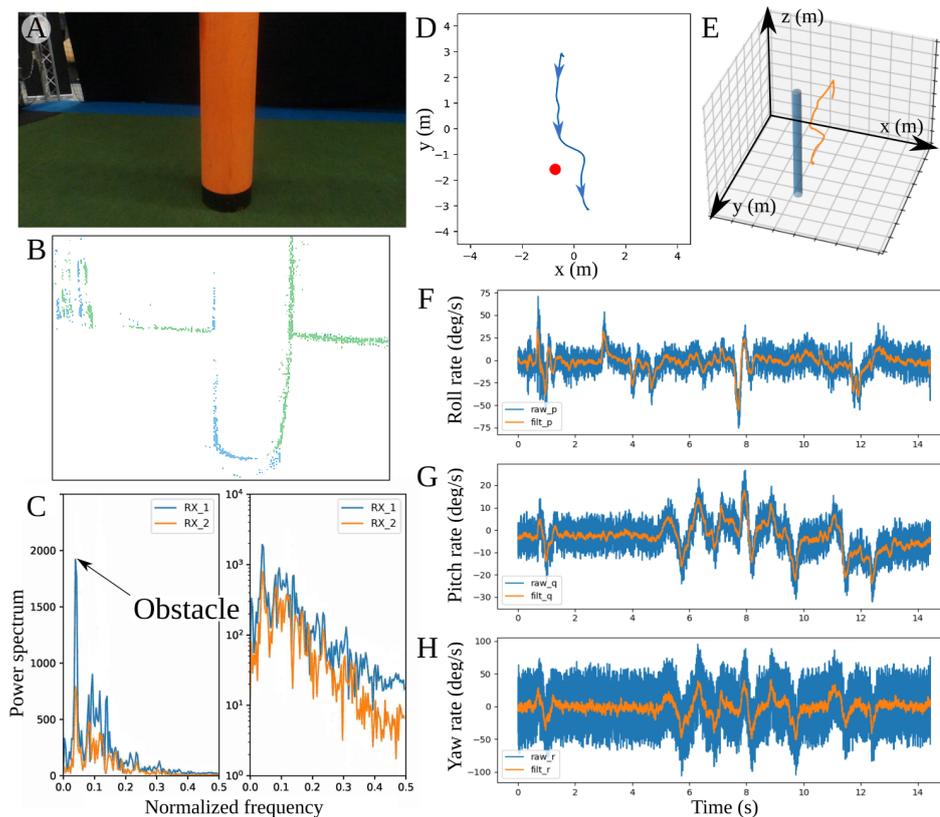

Figure 2.7: Example of a sample from the obstacle avoidance dataset, used in this research to show the performance of the FAITH method offline on real sensor data. **(A)** RGB camera. **(B)** DVS camera. **(C)** Radar sensor (linear and logarithmic scales). **(D-E)** 2D and 3D representations of the ground truth trajectory of the MAV (OptiTrack). **(F-H)** IMU plots (angular rates). Image from Dupeyroux et al. [6].

iments, the ground truth course angle is determined from the OptiTrack data, optic flow is generated and temporal slices are given as input to the MATLAB implementations of the FAITH and other FOE estimation methods. The results of this dataset experiment are presented in the paper (Chapter 3).

Onboard experiment

Because the simulated and dataset experiments show state-of-the-art performance of the FAITH method, an online experiment is performed. The dataset experiment already demonstrated the performance on real sensor data, therefore the online test is used to show the application of the FAITH method in a real-time obstacle avoidance task. As this is the main research goal, the set-up of this experiment is discussed in Section 2.2.4 on the obstacle avoidance strategy.

Concluding, the FAITH method for FOE estimation is shown to have state-of-the-art accuracy, with beyond state-of-the-art computation time on both simulated and real sensor data. Therefore, this method is used in the final processing pipeline. The next sections will give background on the obstacle detection and avoidance sub-system designs, which use the optic flow vectors and FOE estimation as input.

2.2.3. Obstacle Detection

Detecting obstacles using a monocular visual sensor is a complex task, in which many visual cues can be used. The literature review describes the visual cues which can be used to discern objects, of which motion based cues are most relevant for monocular visual obstacle detection (Appendix A, Section 4.2.1). As optic flow describes the apparent motion of an image of the scene, it provides valuable motion based cues about objects. Therefore, optic flow is used as the primary source of cues for the location of obstacles. As sparse normal optic flow is available and the computational resources are highly restricted, an algorithmic approach is taken to the obstacle detection task.

The literature study introduced time-to-contact (TTC) as a concept that is encountered both in nature and robotics. Neural networks in nature are capable of detecting looming predators, or even enable complex dynamic tasks such hitting a baseball. This property of optic flow is related to the motion parallax, which states that objects closer to the observer have a higher apparent velocity. The TTC is defined as the distance between observer and object, divided by their relative velocity. As both the relative velocity and distance are often unknown in visual TTC estimation, a different approach is taken. The TTC estimation in this research uses the geometrical properties of optic flow to estimate the TTC. This is a combination of the FOE, vector location and magnitude. The derivation is stated in Appendix I of the research paper in Chapter 3. This results in a TTC for each optic flow vector in the image. If the velocity of the observer is very low, or the obstacle very far away, the TTC tends to go towards infinity. Divergence is inversely related to the TTC and therefore has converging properties in these scenarios. Although this seems an advantage, divergence values are therefore also lower (i.e. zero for infinite obstacle distance or zero observer velocity) and less suitable to be clustered properly. Therefore, TTC is chosen as primary clustering variable. The spatial information of the object is captured in the vector locations. Assuming relatively small objects (e.g, a 50-cm pole at several meters from the MAV) and high contrast object edges, an additional assumption is made on the proximity of vectors. If vectors have a similar TTC and location, they can be clustered using an Euclidean distance measure. This research uses Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [5] as it is a non-parametric algorithm. It does not assume a predetermined amount of clusters and is able to determine clusters of arbitrary shapes. After the DBSCAN algorithm identifies the object clusters, the proposed obstacle detection method draws a bounding box around the cluster, indicating the object boundaries.

As this object detection method uses multiple properties of optic flow, which can be very sparse on low-textured objects, the approach has certain limitations. Firstly, the use of optic flow requires translational motion of the observer or objects. For example: if the observer does not move, the lack of optic flow results in no detection taking place. Secondly, if the distance between the object and the background is small, the TTC values of the object and background will be close to each other. This renders effective clustering impossible. Thirdly, object edges should be near each other or connected for the spatial proximity assumption to be fulfilled. Therefore, a clear object silhouette must be present for the DBSCAN algorithm to identify all object vectors as core points. Lastly, the object detection method uses the estimated FOE to determine the TTC of the points. As this estimation deteriorates when the potential FOE area is not fully bounded by optic flow (e.g., due to low-texture), this will also lead to erroneous TTC values and clustering subsequently (see Section 3-C of the paper in Chapter 3).

2.2.4. Obstacle Avoidance

The proposed obstacle detection method outputs the location of the object boundaries in the camera image. With this information, a simple obstacle avoidance strategy is proposed. The basic strategy checks whether the global FOE is inside of an object cluster. If the mean TTC of the cluster is below an user-defined threshold, the obstacle avoidance method gives an 1.5 second roll command in the direction of the cluster with the highest mean TTC. This ensures the MAV flies towards a region with texture, as information is known about this region. The roll command is given as input to the flight controller, which sets the roll rate of the MAV. This basic strategy is valid for static objects, as it does not consider the effect of object motion in determining the avoidance direction.

Similar to the obstacle detection method, this strategy has limitations. The sparsity of optic flow vectors and noise in the estimation limit the effectiveness of the method. The strategy uses the estimated FOE location to determine if a collision is imminent. Therefore, similar to the obstacle detection method, the obstacle avoidance strategy also suffers the consequences of erroneous FOE estimation. If the potential FOE area is not fully bounded by optic flow due to a low-textured environment, the subsequent obstacle detection and avoidance methods also deteriorate.

This concludes the introduction to the methodology and system design assumptions and decisions. As discussed in the introduction, the FAITH method to estimate the FOE has been shown to have state-of-the-art performance and the scientific paper discussing this is therefore the main body of content of this thesis. This paper focuses primarily on the FOE estimation method and uses the obstacle detection and avoidance method proposed in this research to demonstrate its onboard performance. The following chapter includes the paper, after which a conclusion is drawn and recommendations for future research are made.

3

Scientific Paper on FAITH Method

FAITH: Fast iterative half-plane focus of expansion estimation using event-based optic flow

Raoul Dinaux, Nikhil Wessendorp, Julien Dupeyroux and Guido C. H. E. de Croon*

Abstract— Course estimation is a key component for the development of autonomous navigation systems for robots. While state-of-the-art methods widely use visual-based algorithms, it is worth noting that they all fail to deal with the complexity of the real world by being computationally greedy and sometimes too slow. They often require obstacles to be highly textured to improve the overall performance, particularly when the obstacle is located within the focus of expansion (FOE) where the optic flow (OF) is almost null. This study proposes the FAst ITerative Half-plane (FAITH) method to determine the course of a micro air vehicle (MAV). This is achieved by means of an event-based camera, along with a fast RANSAC-based algorithm that uses event-based OF to determine the FOE. The performance is validated by means of a benchmark on a simulated environment and then tested on a dataset collected for indoor obstacle avoidance. Our results show that the computational efficiency of our solution outperforms state-of-the-art methods while keeping a high level of accuracy. This has been further demonstrated onboard an MAV equipped with an event-based camera, showing that our event-based FOE estimation can be achieved online onboard tiny drones, thus opening the path towards fully neuromorphic solutions for autonomous obstacle avoidance and navigation onboard MAVs.
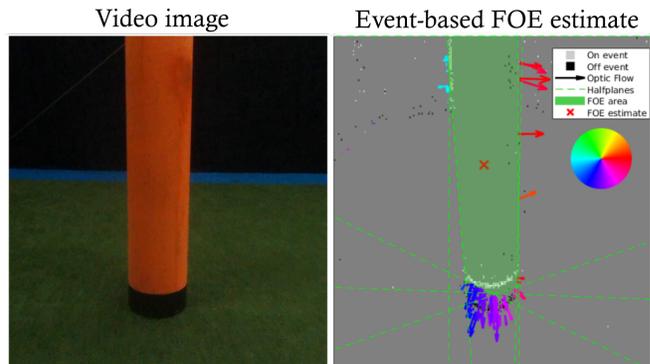
Fig. 1. An application of the FAITH method for fast and accurate FOE estimation in MAV flight towards a single pole. The left plot shows a video-image from the MAV, flying towards a single pole in the TU Delft flying arena. The right plot shows an event-image of the pole with an overlay of optic flow vectors and the FOE estimation, performed by our method.

I. INTRODUCTION

Autonomous navigation, including path planning, obstacle avoidance, and localization, for both ground and aerial robots is considered as one of the top ten technological challenges of our time [1]. Despite outstanding studies in this field, it must be noticed that we still fail at tackling real-world scenarios, where lighting conditions can change abruptly, light can be absent, and where obstacles can severely hamper the performance of the navigation system running onboard the robot. Another crucial aspect for making an autonomous navigation system suitable for real-world applications is to make sure that the robot can deal with high speeds. This is precisely one of the bottlenecks for drone applications, where computational resources and energy usage are important factors for the viability of the proposed method. Lastly, the navigation system must be endowed with deep auto-adaptation skills to make it worth deploying onboard robots in complex environments that remain hard to model, or of which the core nature is simply not understood yet.

The limitations of navigation systems are multi-factorial, but the most important reason for this may be the sensing component itself. While observing the animal kingdom, one can note that each species optimized its sensors to better evolve in its environment. For instance, birds and insects are

*All authors are with Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629HS Delft, The Netherlands. j.j.g.dupeyroux@tudelft.nl

sensitive to the polarization state of the skylight to estimate their course, dung beetles retrieve navigational cues from the Milky Way, and eagles have an extremely high visual acuity to better find their preys. In comparison, robots are often equipped with cameras for which both the temporal $(30 - 60\ fps$ on average) and the visual resolutions are limited. This gets even more crucial with small drones. Given their small dimensions and weight, micro air vehicles (MAVs) are safe to operate autonomously around humans in complex environments. Unfortunately, MAVs are endowed with highly restricted power capacity, and extremely limited computational resources. Their use is also hampered by the risk of GPS failure indoor, magnetometer disturbances because of surrounding ferrous materials (buildings, infrastructures), and IMU drift over time. Embedded cameras have greatly contributed to the reduction of navigation failure, but their use remains limited by the low computational resources available onboard. It is therefore crucial to determine fast and efficient methods to allow MAVs to autonomously navigate and avoid both static and moving obstacles.

The recent developments in neuromorphic systems represent a promising opportunity for autonomous obstacle avoidance and navigation onboard robots, in particular for MAVs. In this respect, event-based cameras were first released in 2008 by Lichtensteiner et al. [2]. Unlike conventional cameras which output images at a fixed frame-rate, event-based cameras produce a stream of asynchronous and independent events reporting changes in brightness at the pixel level [3]. Therefore, these cameras inherently capture the apparent motion. The intensity change threshold, which triggers the

pixel, is user-defined. The events are labeled by the pixel location, trigger time and a polarity (+1 for positive change of brightness, −1 for a negative change). Event-based cameras offer a high dynamic range (> 120 dB) along with a high temporal resolution (in the range of microseconds). These advantages make event-based cameras inherently insensitive to classical visual artifacts such as motion blur or the tunnel effect. As a result, these cameras provide accurate visual information at extremely high speed, making them suitable for aerial robotics, including MAVs, for various tasks such as obstacle avoidance [4], [5] and visual odometry [6].

In this study, we propose the FAITH (FAst ITerative Half-plane) method to estimate the course of the MAV by means of an event-based camera (i.e., the DVS240C [7]), along with a fast RANSAC-based algorithm for the determination of the focus of expansion (FOE) using optic flow (OF) as an input (Fig.1). Optic flow is described as the pattern of apparent motion of objects in a visual scene caused by the relative motion between the observer and a scene [8]. The FOE is therefore defined as the singular point from which the apparent OF expands, assuming the scene is static and the motion of the observer is purely translational. This point indicates the course of the observer, and therefore is a crucial element in visual-based navigation. Appendix I gives a theoretical background to OF and FOE estimation. Determining the FOE is challenging as only normal flow is available, and the computational limitation of the MAV does not allow for expensive online visual-processing.

The determination of the FOE onboard mobile systems equipped with cameras has received large attention from researchers over the past decades, showing a great variety of approaches to solve this very complex problem. In the following study, we focused on sparse OF-based FOE estimation, for which state-of-the-art solutions currently available can be divided into three categories: (i) counting vectors directions [9], [10], (ii) creating a probability map based on negative vector intersections and (iii) based on negative half-planes.

Methods relying on counting vectors show limited performance when exploited in online MAVs application. To reduce the computation cost of online FOE estimation, methods based on probability maps seem to be a promising alternative. Guzel et al. [11] proposed to compute a probability map based on the amount of OF vector intersections per location. They demonstrated the performance of their method through a navigation task with a ground robot equipped with a camera. A similar method was implemented by Buczko et al. [12], where RANSAC scheme randomly selects two OF vectors, create a candidate FOE location by calculating the intersection, and test this location against all OF vectors. After a predetermined amount of iterations, the candidate with the highest amount of inliers is selected as the FOE. Results obtained with a RGB camera showed a translation error as low as 0.81%. Yet, using vectors intersection remains a limited solution to FOE estimation since OF estimation on natural scenes is a complex task and the resulting estimates (normal flow) can differ from the true flow.
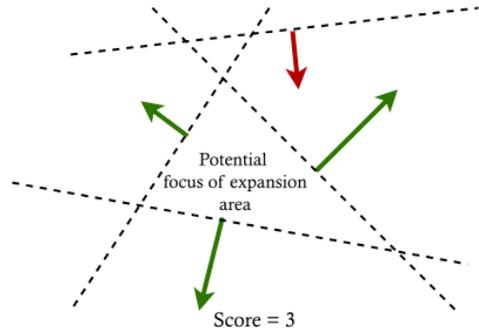

Fig. 2. Schematic example of an FOE estimation by the FAITH method. The arrows represent normal optic flow, the dotted lines their orthogonal half-planes. As the centre of the potential FOE area lies within three half-planes, the iteration score of this estimation is three.

To compensate for this, it has been proposed to build the probability map using the negative half-planes [13]. As the normal flow is computed, the assumption is made that the FOE must lie in the negative half-plane of as many normal OF vectors as possible. For each OF vector an orthogonal line is taken, which intersects the vector location. The negative half-plane of this orthogonal line is used to update the probability map. All locations which are not updated are subject to exponential decay over time. The location with the highest value on the probability map is selected as the FOE. This method has been used with event-based cameras to estimate time-to-contact (TTC) in the context of obstacle avoidance with MAVs [14].

Although the negative half-planes approach suggest an improvement in the course estimation, it is worth noting that the new computation introduced in the plane estimation and intersection considerably affect the overall performance.

Contributions – We propose the FAITH method for FOE estimation based on negative half-planes intersections, further optimized by means of a RANSAC process. Our contributions are:

(a) a novel course estimation algorithm (FAITH) that is highly computationally efficient, runs real-time onboard robots (including MAVs), and provides a robust estimate of the FOE even with poor-textured obstacles;

(b) an exhaustive assessment of the overall performance of the FAITH algorithm, first using the ESIM event-based camera simulator [15], and then using an extensive dataset collected in the TU Delft flying arena equipped with the OptiTrack motion tracking system;

(c) a real-world demonstration of the performance onboard an MAV designed for the purpose of this study.

II. MATERIALS AND METHODS

A. The proposed FAITH method

We apply an event-surface method to compute the local normal flow based on visual events streamed by an even-based camera [16], [17]. When an OF vector is available, we assume the FOE lies in the negative half-plane delimited by the straight line orthogonal to the OF vector (Fig. 2).

Algorithm 1 FAITH method for FOE estimation
---
for *iterations* **do**
    stop search = false
    Pick two random OF vectors.
    Calculate *current FOE area* (bounded by negative half-planes orthogonal to the selected vectors).
    while *stop search* == false **do**
        Pick new random vector.
        Calculate *new area* (bounded by *current FOE area* and the negative half-plane of the selected vector).
        if *new area* < *current FOE area* **then**
            current FOE area = *new area*
        else
            Calculate *score* as the total amount half-planes the center of *new area* lies in.
            stop search = true
        if *score* > *max score* **then**
            max score = *score*
            best area = *new area*
    FOE = center of *best area*

The aperture problem limits OF on edges to be normal to the edge, whereas only OF on corners result in true OF. Therefore, the assumption is made that the FOE must lie in the negative half-plane of a line orthogonal to the OF vector. We then build on the approach from [13] to compute a probability region for the FOE estimation. As MAVs are limited by their computational resources, the ego-motion estimation should require as little as computation as possible while still assuring accuracy. Because we are using an event-based camera, the algorithm implemented in [13], which updates all pixels of the probability map for each OF vector, will inevitably lead to large computational needs.

To compensate for the computational cost, we propose to apply a RANSAC scheme to create an FOE area by taking the intersection of the negative half-planes of two randomly chosen vectors (Algorithm 1). A new OF vector is then chosen and the intersection of the negative half-plane is updated. If the new vector reduces the size of the FOE area, it is added as a new boundary. This process is continued until the new chosen vector does not reduce the size of the FOE area. Then the center of this area is calculated (Fig. 2) and an iteration score is assigned by computing in how many negative half-planes the FOE estimate lies. The score and center position are saved and another iteration is performed. After a user-defined amount of iterations, the search is stopped and the iteration with the highest score is chosen as the best FOE candidate.

B. Computational complexity analysis

Given that our proposed method extends the one introduced in Clady et al. [13], we determined the computational complexity of both algorithms to assess their overall computational performances. For each vector, the method by Clady et al. updates all locations of a probability map. Therefore the computational complexity for this method can be written

as $\mathcal{O}(N * M_p)$, with N the number of OF vectors and M_p the number of pixels in the probability map.

In our method, the majority of the computational complexity lies in checking how many inliers the candidate locations have. This depends on the total of OF vectors and candidate locations, which is equal to the user-defined number of iterations run. Therefore it can be expressed as $\mathcal{O}(N * I)$, with N the number of OF vectors and I the number of iterations (i.e., potential FOE locations). To get an estimate of this I, the theoretical minimum of RANSAC iterations required to construct a proper model with a chosen probability is:

$$I = \frac{log(1 - p)}{log(1 - w^n)} \quad (1)$$

where I is the required number of iterations, p is the probability of selecting a proper model, w is the ratio between inliers and the total set, and n is the sum of inliers required for a proper model.

This formula can be seen as a theoretical upper bound as it assumes that the random selection of vectors can include already chosen vectors. For example: requiring a probability of 95% to find a proper model, assuming at least 10 vectors are required for creating a proper model and assuming that 75% of the total set consists of inliers, the total amount of iterations required is 52. Comparing the computational complexity of both methods (assuming a 240×180 pixel probability map) shows that the method implemented by Clady et al., $\mathcal{O}(43200 * N)$, is a few orders of magnitude more complex than the proposed method, $\mathcal{O}(52 * N)$. Therefore, it is concluded that in the general case the theoretical computational complexity of our method is lower, and user-defined.

III. PERFORMANCE BENCHMARK

To assess the performance of our method, we first test it in a virtual environment featuring an event-based camera simulator. Then, we demonstrate its robustness by testing it on a manually controlled obstacle avoidance dataset that we collected in our indoor flying arena equipped with the Opti-Track motion capture system (see Supplementary Materials). Lastly, the online performance is demonstrated by testing the method onboard an MAV equipped with a DVS240 camera in an autonomous obstacle detection and avoidance task. For the sake of experimental simplification, the motion of the MAV is bounded to translation in the horizontal plane and within the camera FOV. Appendix II discusses the impact this assumption. While assessing the performance of our proposed method, we compare it with the state-of-the-art FOE estimation methods. As described in Section I, three categories of FOE estimation methods using sparse normal flow are identified. Therefore, we also implement the three following algorithms to test them on both simulated and real-world dataset: (i) the vector counting method from Huang et al. [10], (ii) the probability map method based on vector intersections implemented by Buczko et al. [12], and (iii) the negative half-planes method introduced by Clady et al. [13],
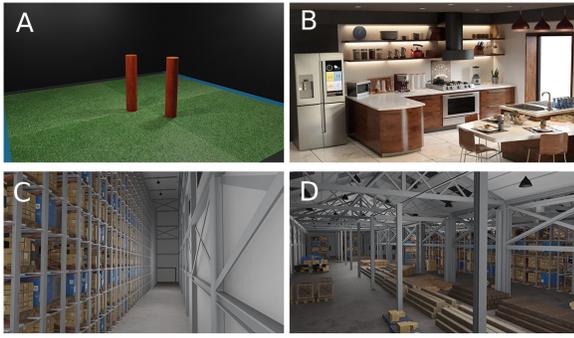
Fig. 3. Rendering of four scenes used in the simulated benchmark: **(A)** the TU Delft flying arena, **(B)** a kitchen, **(C)** a set of storage shelves, and **(D)** a wood warehouse.
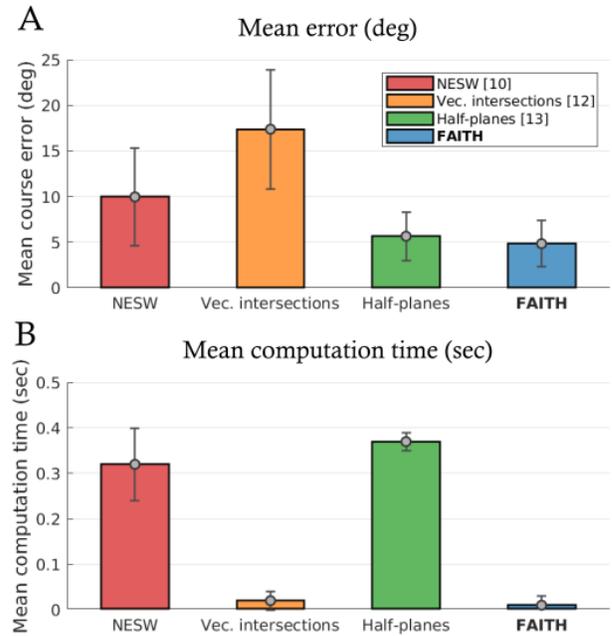

Fig. 4. Comparison of the overall performance of our FOE estimation method with three other state-of-the-art methods, after testing over 100 distinct trajectories in the four simulated environments (Fig. 3). **(A)** Average angular error (in degrees) in the FOE estimation. **(B)** Mean computation time (in seconds) required to process 10^3 OF vectors.

further referred to as 'NESW', 'Vec. Intersections', and 'Half-planes' respectively.

A. Benchmark on simulated data

First, the FAITH method is benchmarked in a simulated environment using the ESIM event-based camera simulator [15] provided with the DVS240 event-based camera specifications which we used in our indoor obstacle avoidance dataset. Four distinct 3D scenes are exported from the open-source software Blender to *.obj* files. These scenes have different textures and layouts to ensure the diversity of environments (Fig. 3). We then provide these scenes to the ESIM simulator along with 100 flight trajectories (camera coordinates over time in a *.csv* file). To test the robustness of the methods to different FOE locations, the trajectories are chosen such that the FOE covers all course angles in the FOV ($-30°$ to $30°$). Both straight trajectories (with different yaw angles) and sway trajectories (varying the FOE during simulation) are used. The ground truth FOE is known from the simulated trajectory and camera pose.

The results of these $N = 100$ simulations are shown in Figure 4 and Table I. The mean course angle estimation error is compared for the four methods. The FAITH method shows state-of-the-art accuracy, with a mean error of $4.84° \pm 2.53°$. The worst performance is achieved by the 'Vec. Intersections' method with an overall angular error of $17.39° \pm 6.54°$. Fig. 4-B shows the mean computation time per 1000 vectors. This proves the large reduction in computational effort for our method, confirming the theoretical insight detailed in Section II-B. It also clearly demonstrates the computational efficiency of the 'Vec. intersections' method, which also does not update all probability map pixels and uses a RANSAC scheme. In contrast, the mean course estimation error is significantly larger for the 'Vec. intersections' method, confirming that using the OF vectors instead of half-planes decreases the accuracy (normal vs. real flow).

B. Benchmark on an event-based obstacle avoidance dataset

The event-based domain requires new approaches and datasets due to the sparse asynchronous event representation. To address this challenge, a novel obstacle avoidance dataset

using a real event-based camera was recorded (see Supplementary Materials). It consists of ~ 1350 manual obstacle avoidance runs performed with an MAV equipped with an event-based camera (DVS240), a 24-GHz radar sensor, a Full-HD RGB camera, a 6-axes IMU, and OptiTrack data for position and attitude ground truth. The obstacles consist of one or two 50-cm wide poles, of which the ground truth location is known (Fig. 5). Each trial consists of approximately 10 seconds of recording.

The benchmark on the obstacle avoidance dataset is performed by comparing the four methods. The ground truth FOE is available as the OptiTrack system tracks both the pose and the position of the MAV during the trials. These trials contain a variety of trajectories, obstacles and backgrounds to ensure diversity of environment and motion.

The results of this benchmark, as seen in Figure 6 and Table I, confirm those obtained with the simulator indicating that the FAITH method outperforms the others. Comparing these results show that the FOE estimation accuracy of all methods on the live recorded data is lower than when using simulated data. This is a consequence of multiple factors, such as the higher amount of noise from the DVS240 camera, vibrations caused by the propellers of the MAV or the increased sparsity of the OF due to the texture and lightning conditions of the scene. In contrast, the relative performance of the methods does not change, our method is still the most accurate and computationally efficient of these methods.

TABLE I

OVERALL PERFORMANCE OBTAINED WITH THE FAITH METHOD, COMPARED TO OTHER STATE-OF-THE-ART FOE ESTIMATION METHODS, FOR BOTH THE SIMULATED BENCHMARK (ESIM) AND OUR OBSTACLE AVOIDANCE DATASET.

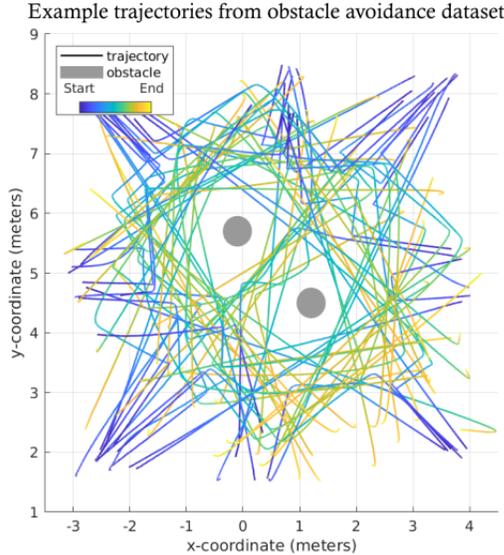| Method	ESIM benchmark ($N = 100$)		Obstacle avoidance dataset ($N = 1300$)	
	Angular error	Computation time	Angular error	Computation time
NESW [10]	$9.97° \pm 5.34°$	0.32 ± 0.08 s	$18.87° \pm 3.95°$	0.35 ± 0.12 s
Vec. Intersections [12]	$17.39° \pm 6.54°$	0.02 ± 0.02 s	$17.49° \pm 5.41°$	0.05 ± 0.02 s
Half-planes [13]	$5.66° \pm 2.67°$	0.37 ± 0.02 s	$10.60° \pm 3.91°$	0.43 ± 0.15 s
FAITH	$\mathbf{4.84° \pm 2.53°}$	$\mathbf{0.01 \pm 0.02}$ s	$\mathbf{10.06° \pm 2.88°}$	$\mathbf{0.05 \pm 0.02}$ s

Fig. 5. Representation of 78 sample trajectories from the obstacle avoidance dataset. This dataset is used to validate the performance of the FOE estimation method. The MAV is controlled manually and two poles in the center of the TU Delft flying arena are avoided.
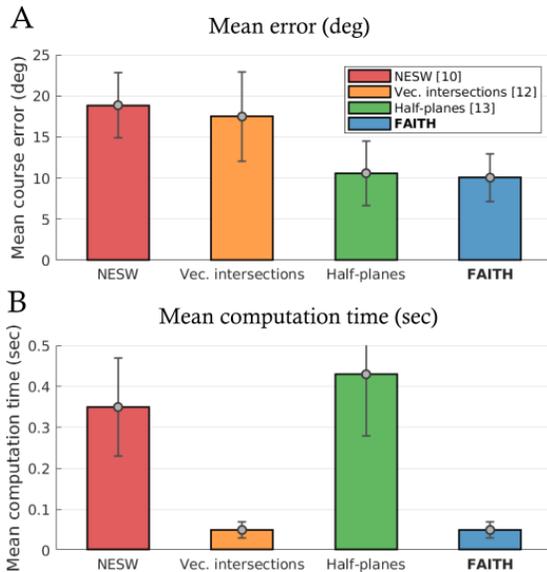

Fig. 6. Comparison of the overall performance of the FAITH method with three other state-of-the-art FOE estimation methods, after testing over 1300 samples of our obstacle avoidance dataset. (**A**) Average angular error (in degrees) in the FOE estimation. (**B**) Mean computation time (in seconds) required to process 10^3 OF vectors.

C. Experiment onboard MAV

To show onboard performance of the FAITH method, it is implemented within the ROS (Robot Operating System) framework using C++ and used in an autonomous obstacle avoidance task. The MAV is set to fly straight-forward at a constant velocity in the flying arena and encounters a pole approximately halfway. The obstacle avoidance algorithm then detects the pole and gives an avoidance command to the iNav Autopilot running onboard the MAV.

1) Obstacle avoidance strategy: A straight-forward obstacle avoidance algorithm is designed using OF as input and an avoidance course as output which is fed to the iNav autopilot. In order to detect an obstacle, OF is clustered based on the concatenated image coordinate (normalized between 0 and 1) and TTC (normalized by mean and variance). The FOE is estimated using the FAITH method. The TTC is calculated using this FOE estimation (Appendix I-C). To cluster the vectors, we apply a Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [18] with $\epsilon = 0.2$, $minPts = 20$ and an Euclidean distance measure. High TTC values are clipped to a user-defined maximum. The mean TTC of the clusters is calculated and the cluster with lowest TTC is assumed to be the highest priority obstacle. A bounding box is drawn around this obstacle cluster. If the FOE location is within the obstacle region and the mean obstacle TTC is below a user-defined threshold, the algorithm gives an 1.5 second roll command to the autopilot to avoid the obstacle. The sign of the roll command is determined by selecting the direction towards the cluster with the highest mean TTC.

2) Hardware architecture: The MAV is a quadrotor built upon the GEPRC FPV frame kit Mark4, featuring the Kakute F7 Tekko ESC Combo v1.5 flashed with the iNav autopilot. The embedded CPU consists in the Intel Up board (64-bits Intel Atom x5 Z8350 1.92GHz Processor) running the Linux 18.04 LTS operating system. An overview of the hardware architecture is provided in Fig. 8. The board is used for data acquisition and processing, autonomous navigation, and wireless communication with the host machine. All MAV-related embedded processing, i.e., DVS data acquisition, OF and FOE estimation, obstacle avoidance and navigation, are performed within the ROS (Robot Operating System) framework. As for the obstacle avoidance dataset, the MAV is equipped with the DVS240 event-based camera (240×180 pixels). The altitude of the drone is controlled separately with a downward-facing micro LiDAR (TFMini, QWiic).
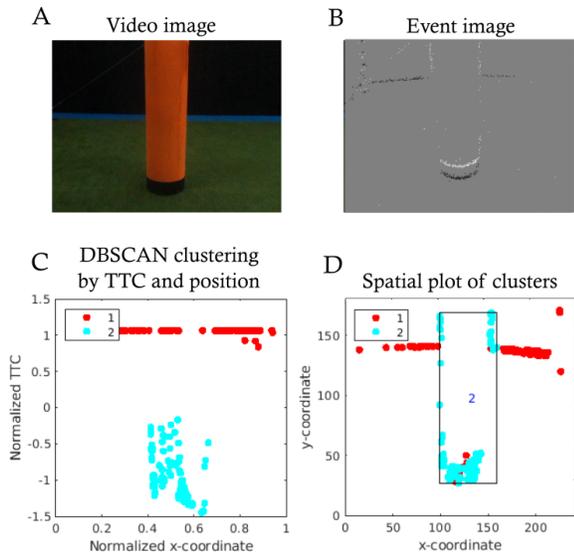
Fig. 7. Example of DBSCAN clustering for the onboard FOE estimation experiment, implementing the FAITH method in an obstacle avoidance task. **(A)** Frame-based image of the pole. **(B)** Event-based image of the pole. **(C)** Clustering optic flow based on TTC and position. **(D)** Clusters, mapped to a spatial plot. As Cluster 2 has the lowest mean TTC, it is identified as (highest priority) object and a bounding box is drawn.
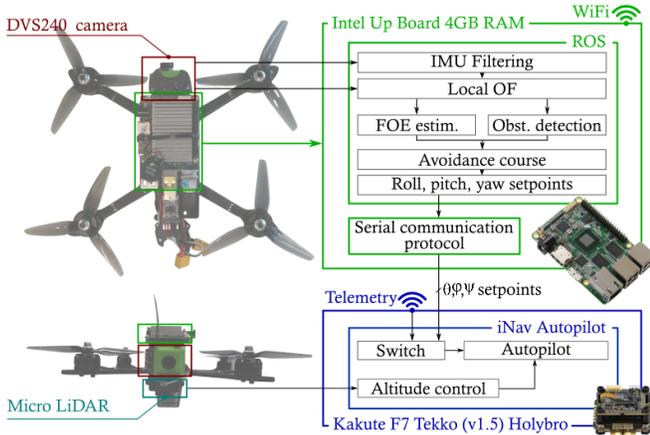

Fig. 8. Hardware architecture of the MAV. The visual processing and the obstacle avoidance algorithms are processed onboard Intel's Up board within the ROS environment. A switch allows the user to switch from manual control to autonomous mode. In both cases, the altitude is kept constant by means of the micro Lidar.

3) Experimental setup: During the experiments, the MAV ground truth position and attitude are determined by the OptiTrack motion capture system installed in the flying arena. A pole, of which the ground truth location is known, is positioned in the center of the flying arena. The MAV is set to autonomously fly along a straight trajectory, from 12 different starting positions and headings. A set of 60% of the trajectories are designed as collision courses with the pole, while the remaining 40% trajectories concern a near pass of the pole. This configuration is meant to qualitatively assess the robustness of the FOE estimation and obstacle avoidance methods in real-world conditions.
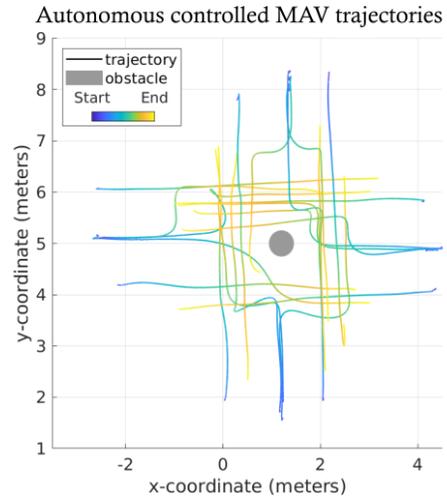

Fig. 9. 20 Successful autonomous obstacle avoidance trajectories using the FAITH method to estimate the FOE. This shows the successful implementation of our method in an obstacle avoidance task.

4) Results: The autonomous obstacle avoidance method, using FAITH to estimate the FOE, is shown to perform a successful obstacle avoidance manoeuvre in 80% of the runs (20 out of 25). The faulty runs are a result of the low-textured scene, which impedes the FOE estimation. When the potential FOE area (Fig. 2) is not fully bounded by OF, the FOE estimation becomes less accurate. This also influences the TTC estimation and subsequently deteriorates the clustering quality. As a result, occasionally when no fully bounding OF is generated in the scene, the object is not detected correctly. Fig. 9 shows the trajectories of 20 successful obstacle avoidance runs. This figure shows the ability of the MAV to autonomously determine its course using the FAITH method and avoid the object. This shows the successful onboard performance of our method in a real-time obstacle avoidance task.

IV. CONCLUSION AND FUTURE WORK

We introduced the novel FAITH method to determine the course of an MAV by means of an event-based camera, along with a fast RANSAC-based algorithm for the determination of the FOE. Using event-based normal OF as input, the method is able to efficiently estimate the course of the MAV. The accuracy and computational performance are validated by performing a benchmark using both a simulated event-based camera data and a novel live obstacle avoidance dataset containing real sensor data. On both simulated and real event-based camera data, the FAITH method shows a state-of-the-art accuracy, with a beyond state-of-the-art computational performance.

We further tested our method in an obstacle avoidance task onboard an MAV, successfully demonstrating real-time performance of our method. The limitations of OF-based strategies in low-textured environments show the bottleneck towards MAV autonomous applications, also suggested by results obtained with our dataset.

Supplementary materials

The ROS implementation of FAITH can be found here: https://github.com/tudelft/faith, and the supporting video https://youtu.be/X09mIqoqAFU. The Obstacle Detection and Avoidance dataset is available at: https://github.com/tudelft/ODA_Dataset.

References

[1] G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, R. Merrifield *et al.*, "The grand challenges of science robotics," *Science robotics*, vol. 3, no. 14, p. eaar7650, 2018.

[2] L. Patrick, C. Posch, and T. Delbruck, "A 128x 128 120 db 15μ s latency asynchronous temporal contrast vision sensor," *IEEE journal of solid-state circuits*, vol. 43, pp. 566–576, 2008.

[3] G. Gallego, T. Delbruck, G. M. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza, "Event-based vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.

[4] D. Falanga, K. Kleber, and D. Scaramuzza, "Dynamic obstacle avoidance for quadrotors with event cameras," *Science Robotics*, vol. 5, no. 40, 2020.

[5] A. Mitrokhin, C. Fermüller, C. Parameshwara, and Y. Aloimonos, "Event-based moving object detection and tracking," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–9.

[6] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 994–1001, 2018.

[7] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, "A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014.

[8] J. J. Gibson, P. Olum, and F. Rosenblatt, "Parallax and perspective during aircraft landings," *The American journal of psychology*, vol. 68, no. 3, pp. 372–385, 1955.

[9] K. Souhila and A. Karim, "Optical flow based robot obstacle avoidance," *International Journal of Advanced Robotic Systems*, vol. 4, no. 1, p. 2, 2007.

[10] R. Huang and S. Ericson, "An efficient way to estimate the focus of expansion," *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, pp. 691–695, 2018.

[11] M. Serdar Guzel and R. Bicker, "Optical flow based system design for mobile robots," in *2010 IEEE Conference on Robotics, Automation and Mechatronics*, 2010, pp. 545–550.

[12] M. Buczko and V. Willert, "Monocular outlier detection for visual odometry," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 739–745.

[13] X. Clady, C. Clercq, S.-H. Ieng, F. Houseini, M. Randazzo, L. Natale, C. Bartolozzi, and R. Benosman, "Asynchronous visual event-based time-to-contact," *Frontiers in Neuroscience*, vol. 8, 2014.

[14] F. Colonnier, L. D. Vedova, R. Teo, and G. Orchard, "Obstacle avoidance using event-based visual sensor and time-to-contact processing," 2018.

[15] H. Rebecq, D. Gehrig, and D. Scaramuzza, "Esim: an open event camera simulator," in *Conference on Robot Learning*, 2018, pp. 969–982.

[16] R. Benosman, C. Clercq, X. Lagorce, S. Ieng, and C. Bartolozzi, "Event-based visual flow," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 407–417, 2014.

[17] B. Hordijk, K. Scheper, and G. Croon, "Vertical landing for micro air vehicles using event-based optical flow," *Journal of Field Robotics*, vol. 35, pp. 69–90, 01 2018.

[18] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." AAAI Press, 1996, pp. 226–231.

[19] H. C. Longuet-Higgins and K. Prazdny, "The interpretation of a moving retinal image," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 208, pp. 385 – 397, 1980.

Appendix I
Optic Flow, Focus of Expansion and Time-To-Contact Theory

A. Optic Flow Theory

Optic flow (OF) consists of two components, due to translation and rotation. The OF generated by translation gives information about the scene and the ego-motion of the observer. In contrast, the OF generated by rotation does not provide any insights on translational ego-motion. Therefore, the OF in this research is derotated using an onboard IMU such that only OF based on translation is used. This research uses an event-surface method, firstly proposed by Benosman et al. [16], and later improved for online application by Hordijk et al. [17]. This method generates sparse normal OF. In order to describe the underlying geometry of OF, an arbitrary point from the 3D world is projected on a 2D surface. The projected point on the surface has the following coordinates (Fig. 10).

$$x = \frac{X}{Z}, \quad y = \frac{Y}{Z} \tag{2}$$

To determine the motion of this point, the equation above is differentiated with respect to time.

$$\dot{x} = \frac{\dot{X}}{Z} - \frac{X\dot{Z}}{Z^2}, \quad \dot{y} = \frac{\dot{Y}}{Z} - \frac{Y\dot{Z}}{Z^2} \tag{3}$$

Values for $\dot{X}$, $\dot{Y}$ and $\dot{Z}$ can be derived (for a derivation see Longuet-Higgins et al. [19]), resulting in the following OF equations.

$$u = -\frac{U}{Z} + x\frac{W}{Z} + Axy - Bx^2 - B + Cy = u_T + u_R$$
$$v = -\frac{V}{Z} + y\frac{W}{Z} - Cx + A + Ay^2 - Bxy = v_T + v_R \tag{4}$$

Note that these equations consist of a translational (u_T, v_T) and rotational (u_R, v_R) component. The rotational component is a result of camera rotations and does not contain information about the ego-motion of the observer. This effect is compensated in this research by using the known ego-rotation from an onboard Inertial Measurement Unit.

B. Focus of Expansion Theory

When an observer translates through a static scene, the OF diverges from a singular point called the FOE. At this location on the image, the OF is zero and all OF is directed outwards. This position is an indication of the course of the
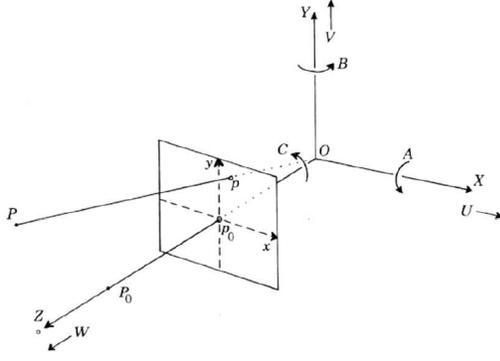
Fig. 10. Optic flow reference system. From Longuet-Higgins et al. [19].

observer. If the OF is zero and the rotational component is filtered out, the following derivation is made using Eq. 4.

$$u_T = 0 = -\frac{U}{Z} + x_{FOE}\frac{W}{Z}$$
$$v_T = 0 = -\frac{V}{Z} + y_{FOE}\frac{W}{Z}$$
(5)

Rewriting these equations gives the following result.

$$x_{FOE} = \frac{U}{W}, \quad y_{FOE} = \frac{V}{W}$$
(6)

To show OF diverges from the FOE, (5) and (6) are used to re-express u_T and v_T.

$$u_T = -\frac{U}{Z} + \frac{xW}{Z} = \left(-\frac{U}{W} + x\right)\frac{W}{Z} = (x - x_{FOE})\frac{W}{Z}$$
$$v_T = -\frac{V}{Z} + \frac{yW}{Z} = \left(-\frac{V}{W} + y\right)\frac{W}{Z} = (y - y_{FOE})\frac{W}{Z}$$
(7)

Rewriting this equation shows the geometrical relation which results in the OF diverging from the FOE.

$$\frac{u_T}{v_T} = \frac{x - x_{FOE}}{y - y_{FOE}}$$
(8)

This geometrical relation is used as basis for the methods discussed in the benchmark (Section III).

C. Time-To-Contact Theory

The Time-To-Contact (TTC) is a property of each point in an image, describing its relative velocity to the camera principle axis. Eq. 7 can be rewritten to the following equation for divergence.

$$\frac{W}{Z} = \frac{u_T}{x - x_{\text{FoE}}} = \frac{v_T}{y - y_{\text{FoE}}}$$
(9)

The divergence is inversely related to the TTC.

$$\tau = \frac{Z}{W}$$
(10)

In the onboard test of the FAITH method for estimating the FOE, an obstacle detection method is used which clusters the OF based on the vector position and TTC. Divergence is

inversely related to the TTC and has converging properties. Although this seems an advantage over TTC, divergence values are much lower (i.e. zero for infinite obstacle distance or zero observer velocity) and unsuitable for proper clustering. Therefore, TTC is chosen as primary clustering variable in this research.

APPENDIX II
FOE OUTSIDE THE FIELD OF VIEW

The performed benchmark on simulated and real event-based camera data considers only FOE locations inside the camera field of view. In the obstacle avoidance strategy, the MAV flies towards clusters which are within the field of view, as this gives certainty about the scene the MAV is flying towards. If the MAV flies a course which is outside the field of view, our method will provide an unbounded FOE region, and thus also no exact FOE location. Although this is a limitation of our method, it does provide the general direction the MAV is moving towards. The side in which the FOE region is unbounded is also side on which the FOE lies, thus the FOE location is bounded to a half-plane. Of the compared methods in the benchmark, only the vector intersections method (e.g. implemented by Buczko et al. [12]) is able to estimate FOE locations outside the field of view. Fig. 11 shows the performance of the vector intersections method for 40 simulated trials, with an FOE angle ranging from $30°$ to $90°$. The course estimation error and CV grows rapidly as the course is further outside the FOE. This results in CV values of over 300%, which show a very low estimation certainty. Therefore, it is concluded that this method has a limited advantage over our method regarding estimating the FOE outside the field of view.
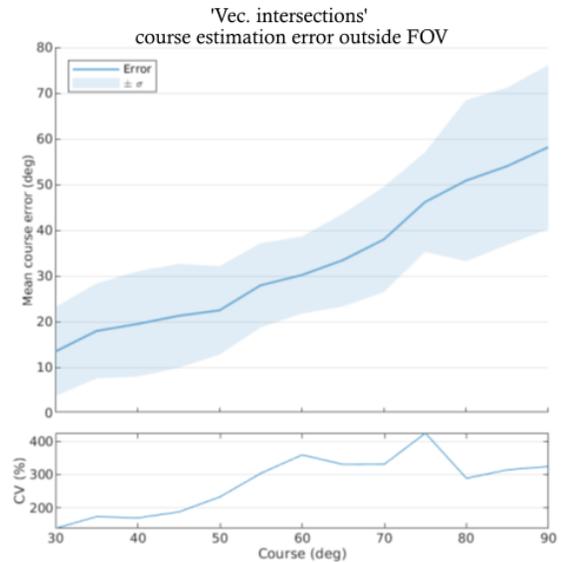

Fig. 11. Performance of the 'Vec. intersections' method implemented by Buczko et al. [12] for a course of $30°$ to $90°$, outside the FOV. The lower plot shows the coefficient of variation as percentage, $CV = \frac{\mu}{\sigma}$

4

Conclusion

As MAVs have limited computational resources and energy storage, navigation tasks are required to be performed onboard, efficient and fast. This research proposes a monocular event-based obstacle detection and avoidance method, applicable onboard an MAV. To achieve this, firstly a literature study is performed which provides a theoretical background using previous research. As a result from this study, an algorithmic approach to obstacle avoidance is chosen, which leads to designing sub-systems and step-by-step processing. A processing pipeline is designed, based on the use of event-based normal optic flow. This pipeline consists of three sections: course estimation, obstacle detection and obstacle avoidance. A novel course estimation method 'FAITH' is proposed which uses optic flow half-planes along with a fast RANSAC scheme. The object detection method is based on DBSCAN clustering of optic flow vectors, using the time-to-contact and vector location as clustering variables. The performance of these methods is demonstrated by three experiments: in a simulated environment, offline on real sensor data and online onboard an MAV. Firstly, the performance of the FAITH method is compared to previous research in a simulated environment. By using an event-based camera simulator and four virtual scenes, an experiment is performed in which the mean course estimation error per method is determined. Secondly, the performance of the methods is determined in an offline experiment on real sensor data. A manually controlled obstacle avoidance dataset using a real event-based camera is recorded in which the ground truth MAV attitude and course is known. The FAITH method is shown to have state-of-the-art accuracy and beyond state-of-the-art computation time on both simulated and real sensor data. Lastly, the real-time performance of the FAITH method and the obstacle detection and avoidance method is demonstrated in an obstacle avoidance task. The methods are implemented in C++ using the Robot Operating System (ROS) framework onboard the MAV. This platform is used in an obstacle avoidance task in which a successful avoidance rate of 80% is achieved. The remaining 20% of unsuccessful trials are due to the low-textured scene, which deteriorates the focus of expansion, time-to-contact and object cluster estimations subsequently. These three experiments show that the designed obstacle detection and avoidance approach is computationally efficient and accurate. This approach can therefore be used to perform monocular event-based obstacle detection and avoidance efficient and fast, onboard an MAV.

5

Recommendations

The main research goal of this thesis is achieved successfully with the assumptions and decisions described in this report. Due to the restricted timeline of the research project, the scope of the research is limited to the critical elements for the research goal. Therefore, certain topics are open for future research. In this chapter, several recommendations for future research are made. These are divided into the three main challenges of this research: course estimation, obstacle detection and obstacle avoidance.

Course estimation

A limitation of the novel FAITH method to estimate the FOE is its inability to perform an estimation outside the FOV. As described in the research paper (Chapter 3), the method uses optic flow half-planes due to the limited information normal optic flow carries. Therefore, the FOE estimation becomes unbounded if the course is outside the FOV of the camera. The vector intersections method discussed in the paper is able to provide an estimation outside the FOE, although this deteriorates fast as the course is further outside the FOV. A fusion between the FAITH and vector intersections method is potentially valuable as the vector intersections method can provide an FOE estimate if the FAITH method gives an unbounded FOE estimate. This would increase the computational demand though, as the two algorithms would execute in series.

The FAITH method also does not take any temporal relation into account between time-frames. Each time-frame, the FAITH method estimates a new FOE estimate. Due to the inertia of the MAV, its course will not change instantaneously. Therefore, a correlation exists between the FOE estimates of subsequent time-frames. By applying for example a Kalman filter to the FOE estimate, this temporal relation can be captured. This is expected to increase the accuracy and robustness of the course estimation.

Obstacle detection

As mentioned in Section 2.2.3, the obstacle detection method is tested on a low-textured pole. As the environment was also low-textured in the experiments, the FOE estimation was occasionally inaccurate, resulting in no obstacle being detected. Although this is a fundamental challenge, the experimental setup in the flying arena of the TU Delft does not represent real-world scenarios well. Therefore, a recommendation is to test the obstacle detection method in realistic scenarios using for example humans or trees as obstacles. This would provide valuable information on the performance of the method in real-life scenarios.

The obstacle detection method is based on the use of optic flow vectors, just as the course estimation method. Although this approach has been shown to be effective, it does not utilize the event-stream itself. As the event-based camera output provides a unique representation of the edges in the image, it could also support the object detection method. As nearby events also have a spatial correlation, this can be introduced in the object clustering as an extra clustering variable. It is expected that this will increase the object clustering quality.

Obstacle avoidance

The implemented obstacle avoidance strategy for this research is based on a threshold of the time-to-contact (TTC) of the nearest obstacle. As this threshold is reached, a 1.5 second roll command is given to the autopilot in the direction of the cluster with the highest TTC. This is a basic avoidance strategy, and does not include any path or motion planning. Three recommendations for future research can be made in this domain.

Firstly, this strategy does not take into account any spatial or temporal relation between the different time-frames, as it only considers the objects, TTC and FOE at the current time-frame. The information about the location of the obstacle is mapped to a two dimensional space, which limits the ability to perform advanced avoidance manoeuvres. To exploit the three dimensional spatial relation between the obstacle and the MAV, for example, the TTC information can be used in a vector field histogram [2] type of approach. As the obstacle (x,y) location and its TTC are known, the Cartesian histogram space can be created containing the obstacle and MAV information. From this space a suitable avoidance direction and velocity can be chosen. Another valuable approach would be to use a potential field [7] in which the goal point is an attractive potential and the obstacle a repulsive potential. The original approach does assume knowledge on absolute distances between the MAV and the obstacle, but this can be solved by using the TTC as a repulsive force.

The second recommendation is related to the object TTC and cluster size. Although the TTC on itself does not provide an absolute location of the obstacle, its development over time does provide an indication of the urgency of the avoidance reaction. Another visual cue that can be potentially valuable is the object size on the camera image. In the implemented object detection method, the object cluster is indicated with a bounding box. The change of size of this bounding box is also an indication of the urgency of the avoidance manoeuvre and therefore can be used to improve the obstacle avoidance method.

Lastly, the third recommendation extends the current avoidance strategy to include dynamic obstacles. As mentioned above, the implemented avoidance strategy uses the object TTC and FOE location to determine an avoidance manoeuvre. This strategy is not tested on multiple poles and is not valid for dynamic obstacles. To improve this strategy for dynamic obstacles, an extension on this method is shown in Figure 5.1. It is not experimentally verified, as this is outside the scope of the thesis. This strategy uses the local cluster FOE to determine whether an object is a collision candidate. After determining the object clusters in the optic flow, the FOE for each cluster is separately estimated by the FAITH method. As seen in Figure 5.1, the strategy uses the object cluster labels and optic flow as input, from which the local FOE is estimated. This strategy is also valid for dynamic obstacles as this takes the local direction of optic flow into account. This strategy is based on work Schaub et al. [10], discussed in Section 4.2.4. of the literature review. The strategy assumes that if an object is on a collision path, the optic flow from the object diverges. If the FOE of these vectors is inside the object boundary, it is on a collision path. A proposed avoidance strategy is to maximize the distance between the cluster FOE and the cluster boundary. This advanced strategy would lead to an increased computational effort, as the FOE of each cluster has to be estimated separately. The concept is similar to the implemented collision detection approach, but is also valid for dynamic obstacles. Therefore, this approach is expected to be valuable for future research.

Figure 5.1: Processing flow for the obstacle avoidance system, using an advanced obstacle avoidance strategy. Green elements are sensors onboard the MAV, blue elements are processing steps, the purple elements are the advanced obstacle avoidance strategy and the yellow element is the MAV control loop. The arrows with labels show the output from functions and their connection to other functions.

A

Appendix - Literature Study

Obstacle Detection and Avoidance onboard an MAV using a Monocular Event-based Camera

MSc. Thesis - Literature Review

R.M. Dinaux

Supervised by: dr. J. Dupeyroux and Prof. G. de Croon

TUDelft

Obstacle Detection and Avoidance onboard an MAV using a Monocular Event-based Camera

MSc. Thesis - Literature Review

by

R.M. Dinaux

to obtain the degree of Master of Science
at the Delft University of Technology,

Student number: 4298470
Thesis committee: Dr. J. Dupeyroux,   TU Delft, supervisor
                    Prof. dr. ir. G. de Croon, TU Delft, supervisor

TUDelft

Abstract

Micro Air Vehicles (MAVs) are able to support humans in dangerous operations, such as search and rescue operations at night on unknown terrain. These scenes require a great amount of autonomy from the MAV, as they are often radio and GPS-denied. This research takes a bio-inspired approach to designing an MAV obstacle detection and avoidance system usable in these conditions. A bio-inspired event-based camera is used to sense the environment, and all processing is executed fully on-board to ensure autonomy. To provide a fundamental theoretical basis for the design of the obstacle avoidance system based on event-based optic flow, several topics are reviewed. Biological inspiration is drawn from insect and nocturnal vision, resulting in concepts usable in the obstacle avoidance system design. Regarding visually guided behaviour of insects, the centering response in corridors, peering behaviour and saccadic motion are identified as usable concepts. The fundamental principles of nocturnal vision show that spatial and temporal summation, and a slow response time with high gain improve vision in low-light conditions. These neural processing concepts are used as bio-inspiration in the obstacle avoidance system design. As this design is based on event-based optic flow, several methods are reviewed to provide an overview of the current field of research. Subsequently, state-of-the-art obstacle avoidance research is reviewed to provide methods which suit different obstacle avoidance scenarios. These methods are implemented in future experimental research. This literature study provides a fundamental theoretical basis for the bio-inspired approach to MAV obstacle avoidance in low-light using an event-based camera.

List of Abbreviations

ANN	Artificial Neural Network
APS	Active Pixel Sensor
CIE	Commission Internationale de l'Eclairage
CMOS	Complementary Metal Oxide Semiconductor
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DAVIS	Dynamic and Active-pixel Vision Sensor
DCMD	Decensing Contralateral Movement Detector
DS	Directionally Selective
DVS	Dynamic Vision Sensor
ECN	Evenly-cascaded Neural Network
EMD	Elementary Motion Detection
FOC	Focus of Convergence
FOE	Focus of Expansion
FOV	Field of View
FPGA	Field-Programmable Gate Array
FPS	Frames Per Second
GPU	Graphics Processing Unit
IMU	Inertial Measurement Unit
LGMD	Lobula Giant Movement Detector
LGN	Lateral Geniculate Nucleus
LIDAR	Light Detection And Ranging
LMC	Large Monopolar Cells
LPTC	Lobula-Plate Tangential Cell
MAV	Micro Air Vehicle
NN	Neural Network
OF	Optic Flow
PD	Proportional Derivative
RANSAC	Random Sample Consensus
RGB-D	Red Green Blue – Depth
SNN	Spiking Neural Network
SNR	Signal to Noise Ratio
SOC	System on a Chip
SWAP	Size Weight And Power
TTC	Time-To-Contact
VSLAM	Visual Simultaneous Localization And Mapping

Contents

1

Introduction

Micro Air Vehicles (MAVs) are able to fly in challenging environments, due to their agile movement and small size. When considering a disaster aftermath at night or a fire in a building with an unknown map, it is often dangerous to send humans into the scene. A fully autonomous MAV is able to enter and navigate in the struck area, providing a safer and faster alternative to sending in humans. This research proposes an obstacle avoidance system which can be used to increase the autonomy of such MAV. Much of the previous research on autonomous MAVs use a ground-station to perform sense and avoid computations, but as indoor or disaster scenes often are radio and GPS-denied, this is not a feasible approach. The Size, Weight and Power (SWAP) of the MAV is limited as the operational conditions require it to fly as long as possible and with fully on-board processing. A bio-inspired approach, based on insects, is used to design the obstacle avoidance system. They have shown to solve complex visual tasks with very little neurons and photoreceptors [97], and thus use very efficient methods. Following this approach, an event-based camera [76] is used in this research. It allows to asynchronously detect changes in pixel brightness, such that it captures motion in the scene. This camera design is based on the same principles as the human visual system (see Section 4.1.3). Based on this visual sensor, a novel bio-inspired obstacle detection and avoidance system is proposed. This system is based on the use of optic flow [49], as it is a well-known phenomena. Its use is encountered both in nature (e.g. in the human brain [67]), and the computer vision domain (e.g. in the famous Lucas-Kanade algorithm [81]). Estimating event-based optic flow is often challenging in dark scenes due to visual noise [46], resulting in a low Signal-to-Noise Ratio (SNR). Therefore, as this obstacle avoidance system has to navigate in dark indoor scenes, several bio-inspired adaptations to the event-based camera processing pipeline are proposed to improve performance in low light. These adaptations are based on biological principles that allow night vision of nocturnal animals. To verify the performance of the obstacle avoidance system and its night vision methods, benchmark experiments are conducted in the experimental research phase. To validate these results, the experiments are performed under different controlled conditions. This literature review provides a fundamental theoretical basis for these experiments.

The review is divided into three chapters: defining research questions (Chapter 2), biological inspiration (Chapter 3) and state-of-the-art obstacle detection and avoidance methods (Chapter 4). As this literature review serves as fundamental basis for the experimental research, the research questions consist of two parts: 'literature study' and 'Experimental research' questions. After defining these questions, first the motivation for taking a bio-inspired approach is given in Section 3.1, along with fundamental knowledge on biological visual systems. Afterwards, the biological principles of obstacle avoidance and vision in low-light are given in Section 3.2. The most relevant biological inspiration for this research is synthesised in Section 3.3. Chapter 4 starts with an introduction to event-based cameras and their non-conventional processing in Section 4.1. Afterwards, key concepts for event-based obstacle avoidance and state-of-the-art methods are discussed in Section 4.2. To select the most suitable methods as basis for the novel obstacle avoidance system, Section 4.3 proposes obstacle avoidance scenarios and their corresponding approaches from state-of-the-art research. Chapter 5 concludes the literature review with a concise summary of the most relevant findings.

2

Research Questions

With the aforementioned context, a concise set of research questions is given in this section. The research questions consist of two parts: literature study and experimental research. The first part uses previous research to answer important questions that define the basis of the experimental research. The experiments implement and test the knowledge from the literature study, leading to a novel obstacle avoidance system and low-light vision methods. This literature review answers the research questions under 'Literature Study', where future experiments answer the research questions under 'Experimental Research'. First, the central research question and objective are given.

The central research question of this thesis is:

> What bio-inspired approach can be used to perform monocular event-based obstacle detection and avoidance in low-light conditions?

The main research objective and contribution is in twofold:

> To contribute to the development of obstacle avoidance systems for MAVs using an event-based camera in low-light; by both providing a novel obstacle avoidance system using only on-board processing, and providing methods to improve performance of the event-based camera in low-light.

The central research question is split into several sub-questions. Each lower-level question answers a part of a higher level question. First, the questions which are answered by literature study are given. Afterwards, the questions which are answered by experimental research are given.

Literature Study

1. Why a bio-inspired approach to obstacle avoidance by insects?
    (a) What does the theory state about the general principles of insect vision?
    (b) What comparison can be drawn between human, insect and computer vision?

2. What biological insights regarding obstacle avoidance in low-light can be gained when considering a nocturnal insect?
    (a) What biological principles are fundamental for obstacle avoidance?
        i. What biological processing principles enable obstacle avoidance?
            A. What does the theory state about biological visual processing?
            B. What is the main biological motion detection theory?
        ii. What type of obstacle avoidance behaviour is encountered at insects?
    (b) What biological principles are fundamental for nocturnal vision?
        i. What does the theory state about 'nocturnal' animals and environments?

  ii. What does the theory state about specific principles used by insects in low-light?

  iii. What does the theory state about noise in low-light, with respect to the visual system?

3. What obstacle avoidance system using a monocular event camera can be used as baseline method?

 (a) What does the theory state about event-based vision?

  i. What is the principle of operation of the event-based camera?

  ii. What are the benefits of using an event-based camera in contrast to frame based cameras?

  iii. What type of data processing is used for an event-based camera?

  iv. What biological principle is the event-based cameras based on?

 (b) What are state-of-the-art monocular event-based obstacle avoidance methods in the mobile robotics domain?

  i. What does the theory state about mapping methods?

  ii. What does the theory state about methods using optic flow?

   A. What are the general principles of optic flow?

   B. What does the theory state about event-based optic flow estimation?

   C. What are state-of-the-art event-based optic flow estimation methods?

   D. What state-of-the-art obstacle avoidance methods use event-based optic flow?

Experimental Research

1. What bio-inspired systems can be designed, considering the results of the literature study?

 (a) What bio-inspired methods can be used in the event-based camera system that potentially improve performance in low-light?

 (b) What bio-inspired event-based obstacle avoidance system can be designed?

2. What impact do these low-light methods have on the performance of the event-based camera system?

 (a) What metrics can be used to evaluate qualitative and quantitative accuracy?

 (b) What experimental setup can be used to measure these metrics?

 (c) What can be learned from testing the low-light methods, considering the evaluation metrics?

3. What is the performance of the novel obstacle avoidance system?

 (a) What metrics can be used to evaluate qualitative and quantitative accuracy, and computational efficiency and complexity?

 (b) What experimental setup can be used to measure these metrics?

 (c) What can be learned from executing the experiments, considering the evaluation metrics?

The research questions from this chapter are used as a general guide in the literature review. In the synthesis, after each chapter, the relevant research questions are explicitly answered. The following chapter reviews biological research in human and insect visual systems, obstacle avoidance and low-light vision.

3

Biological Inspiration

In order to prevent collisions, catch prey and more generally wander around in nature, humans and animals have developed powerful senses. Evolution has caused the visual system to be able to adapt to a variety of scenes, including under day and night conditions. These biological principles can be an useful inspiration in developing robotic systems operating in difficult conditions, for example navigating through low-light environments. Human vision, utilizing a camera eye, is known for its ability to adapt to a broad range of lighting conditions. Insect visual system, often utilizing a compound eye, are developed for specific lighting conditions, but often with higher visual acuity. Therefore both camera eye (human) and compound eye (insect) vision are used as inspiration in this chapter. Firstly, the advantage of the biologically inspired approach will be discussed in Section 3.1. This section will include the general principles of camera eye and compound eye vision, and draw a conclusion on a comparison of human, insect and computer visual systems. After discussing the fundamental principles of these visual systems, the more complex biological principles of obstacle avoidance and nocturnal vision will be discussed in Section 3.2. With this this knowledge, several methods are chosen as biological inspiration for the obstacle avoidance system design.

3.1. Motivation for the bio-inspired approach

In order to highlight the benefits of a biologically inspired approach to visual obstacle avoidance, both the human and animal visual system will be discussed. Insects are able to perform impressive navigational tasks and aerial manoeuvres, whilst having a relatively very basic neural system. Therefore insects are an interesting field of research, to learn and be inspired by. Many insects have a compound eye (which is anatomically different than the well-known camera eye), and therefore it will be discussed separately. In Section 3.1.1, a brief background on the biological principles of human vision will be given. The same will be done for insect vision in Section 3.1.2. Afterwards, a comparison will be drawn between human, insect and computer visual systems in Section 3.1.3. The main goal of these sections is to emphasise the gap between the efficiency of biological visual systems and the computer vision domain. The next section will start with a brief introduction to the fundamental principles of the camera eye and the human visual system.

3.1.1. General principles of human camera eye vision

The camera eye has come a long way of millions of years of evolution, before ending up at the very complex structure it has today. Starting as a small light sensitive spot on the skin approximately 540 million years ago [68], the evolutionary advantages of perceiving the world around using vision made the eye slowly develop and progress to more complex features. To be able to compare the human, insect and computer visual system, first a high level overview of the human visual system is given.

Neural processing of visual information in humans

The visual system of the human eye is a highly optimized chain of events. After being reflected from an object in the scene, a photon travels firstly through the cornea and lens (see Fig. 3.1). Here the path of the photons is adjusted twice, after which it falls at the focal point (in focused vision) on the retina. The retina consists of photoreceptor cells who absorb the photon. These photoreceptors set a chain reaction in motion which ultimately results in an action potential being send towards the brain. A significant part of visual processing

already takes place in the neural connections in the retina. Many of the neurons in the retina connect laterally to other neurons in the same layer, enabling more advanced receptive fields such as motion. These retinal pre-processing concepts will be discussed in Section 3.2.4. After this pre-processing in the retina, the action potential is send from the eyes through optic nerves to the optic chiasm, where the information from both eyes is combined (see Fig. 3.1). Next, the signals are send through the lateral geniculate nucleus (LGN). Here the temporal and special correlation between the signals of the eyes is determined. This creates a three-dimensional representation of the visual image. The output of the LGN is also fed back to the eyes as control the for the vergence and focus of the eyes. After processing in the LGN, the signals are send towards the visual cortex for further higher level interpretation [132]. In the following section, the chain of events from the photon to sending an action potential towards the brain, is highlighted.
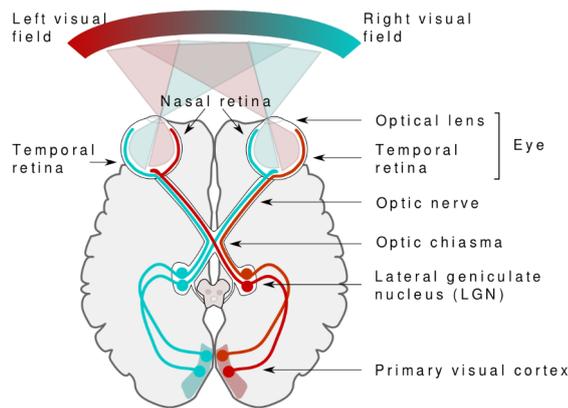

Figure 3.1: Schematic overview of the anatomy of the human visual system. Image from Nieto [93]

Photon to action potential

To understand how biological systems process light into information, the full chain of events has to be taken into consideration. Photons that reflect back from surfaces are captured in the eye at the retina. To be able to convert these photons to a usable signal for the brain, a series of events take place. Firstly, the photon is absorbed by photoreceptors in the retina. In the retina, there are many cells responsible for converting the photon to an action potential. First, the cells responsible for absorbing the photon are discussed, the rods and cones:

Rods   These cells perform best in low light conditions and are located mostly on the outer edges of the retina. The location of these rods make them govern peripheral vision (vision outside of point of fixation). There is only one type of rods in the retina, sensitive to one type of wavelength. Therefore, rods are not able to distinct colour. In the human eye there are much more rods than cones: approximately 120 million rods to 6 to 7 million cones [32]. Rod cell stimuli are added over approximately 100ms, which makes them sensitive to low quantities of light but also makes them slower in response to temporal changes. Rods are sensitive to wavelengths of around 498 nm (green-blue) and insensitive to wavelengths of around 640 nm (red) [135]. This selective sensitivity causes the eye's sensitivity to shift towards blueish green at low light levels (called the Purkinje shift [8]).

Cones   The cone shaped cells perform best in bright light conditions and are responsible for colour vision. These cells are primarily present in the foveola (the visual focus) on the retina, where 100% of the photoreceptor cells are cones. There are three types of cones present in the retina, all sensitive to different wavelengths: short, medium and long wavelengths (for respectively blue, green, red colour vision).

Next to these photoreceptors, which are directly responsible for converting photons, a third type of cells is an important link in the visual system. Retinal ganglion cells are responsible for converting the signals from photoreceptors to action potentials which are processed by the brain.
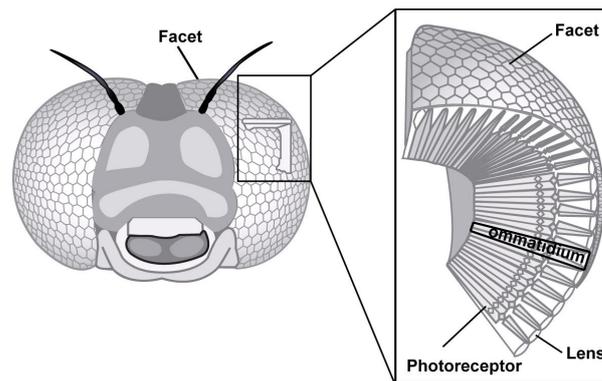
Figure 3.2: Compound eye overview with dissection. Image from EPFL [37]

In order to process the photons to signals usable by the brain, rods and cones use a chemical chain of events. Rhodopsin (for rods) or iodopsin (for cones) absorb a photon, generating a chain of amplifying unstable events. This eventually results in bipolar cells inducing action potential in the retinal ganglion cells, which fire this electric signal towards the brain [89].

This section has given a brief physiological background on human vision. In the next section, an introduction to the fundamental principles of insect vision will be given.

3.1.2. General principles of insect compound eye vision

From an evolutionary perspective, the compound eye has taken a different path than the human eye, but nonetheless has developed into a highly sensitive sensor. A short introduction to the visual system of insects using compound eyes is given, starting with their visual sensor: the compound eye.

Compound eye

The compound eye is the visual organ of many arthropods, such as insects. It consists of thousands of ommatidia which consist of small independent photoreception segments. Each segment contains a cornea, 'facet' lens and photoreceptor cells in the 'rhabdom'. The rhabdom is a membranous structure that contains rhodopsin cells which absorb photons (see Fig. 3.2 for an overview of the compound eye).

The inner structure of these compound eyes differ between insects, and are dependent on the living environment and diurnal ('living by day') or nocturnal ('living by night') lifestyle. These compound eyes are typically split into three categories: *apposition*, *optical superposition* and *neural superposition* compound eyes (see Fig. 3.3).

The least complex type of compound eye is the *apposition* eye (top row in Fig. 3.3). This is thought to be the evolutionary predecessor of the more complex neural and optical superposition eyes [94]. The light entering the eye only reaches the photoreceptor by a single corneal facet lens located directly above. This type of eye is mostly seen in diurnal insects, living in bright light.

In *optical superposition* eyes (middle row in Fig. 3.3), the light passes the lens through a bullet shaped cone and is send across a clear zone in the eye. After passing this clear zone it reaches a single photoreceptor in the retina. Up to several thousands of lenses forward light towards a single photoreceptor, resulting in significant improvement in visual sensitivity. Therefore, many nocturnal animals have this form of refracting superposition eyes. The effective width of the pupil (or aperture) is much larger (see Section 3.2.4, 'A' in Equation 3.2), resulting in a high visual sensitivity [138]. Consider for example a comparison between the diurnal bee and the nocturnal dung beetle. The diurnal bee has apposition eyes and a visual sensitivity of 0.1 μm^2 steradians. The nocturnal dungle beetle has superposition eyes and this results in a visual sensitivity of 68 μm^2 steradians, therefore much more sensitive than the diurnal bee [41].

Neural superposition eyes (bottom row in Fig. 3.3) have the same separated lens and rhabdom structure as apposition eyes but groups of different ommatidia are connected through neural connections (neural pooling). A study by Kirschfeld et al. [64] showed that one central rhabdomere shares the field of view with his six adjacent ommatidia. The receptor responses of these seven ommatidia, which image the same section of the scene, are send to the same synaptic cartridge in the lamina. In order to achieve this, the angular separation
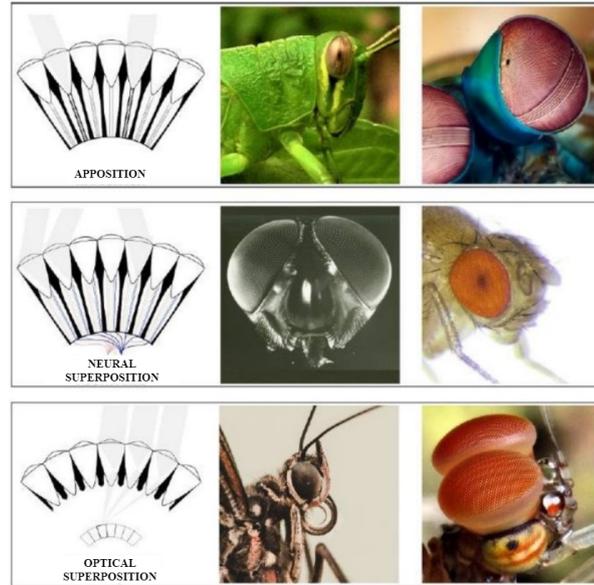
Figure 3.3: Three types of compound eyes: apposition, neural superposition and optical superposition. Examples (left-to-right, top-to-bottom): migratory locust, mantis shrimp, house fly, fruit fly, moth, mayfly.

of rhabdomeres ($\Delta\phi$ in Fig. 3.4) must be equal to the interommatidial angle (D/R) across the whole eye. Land et al. [69] showed that the amount of photons ($\bar{N}$ in Eq. 3.1) required in order to perceive a certain contrast (C in Eq. 3.1), are related by the following formula:

$$\bar{N} > \frac{1}{C^2} \tag{3.1}$$

This pooling in neural superposition eyes allows for the photon signals of 7 receptors to be combined (7N in Eq. 3.1). This improves the contrast detectability by $\sqrt{7}$ without compromising resolution. To achieve this increment in resolution for a normal apposition compound eye, the rhabdom diameter (D in the Figure 3.4) has to be scaled by $\sqrt{7}$, which would compromise the resolution (see Eq. 3.2).

Figure 3.4: Compound eye dimensions. (a): relation between apposition compound eye (upper) and camera type eye (lower). (b): The acceptance angle of an ommatidium ($\Delta\rho$) split up in its two parts consisting of the point-spread function (left), and the rhabdom acceptance angle (right). Image adapted from Land et al. [70]

Non-uniform ommatidia density

The compound eye does not consist of a grid of evenly placed ommatidia, but rather a non-uniform distribution. This change in $\Delta\phi$ (see Fig. 3.4), and thus ommatidia density, causes certain areas of vision to be of higher acuity than others. This change in density exists in both the horizontal as vertical directions. Acuity in certain visual areas is traded for other areas, a trade off due to the limited amount of space for ommatidia. Optical superposition eyes show this effect less often, as they rely on a spherical shape of the eye.

There are three types of density structures often seen in insects, related to: forward flight, prey and mating behaviour and flat scenes such as water surfaces [70]. Flying herbivores such as bees and butterflies have a density structure that supports forward flight: a small $\Delta\phi$ in the horizontal and vertical front of the eye and increasing towards the back (see Fig. 3.5). The second density structure often seen in insects is related to prey capture and mating. This acute zone is located forward and upward. Often only males have this acute zone, using it to detect females against the sky or for hunting preys. The third density structure, found in insects, is the horizontal acute zone. This is often found in flat environments such as water or deserts. The high density of ommatidia in the horizontal plane helps imaging the horizon during scavenging.
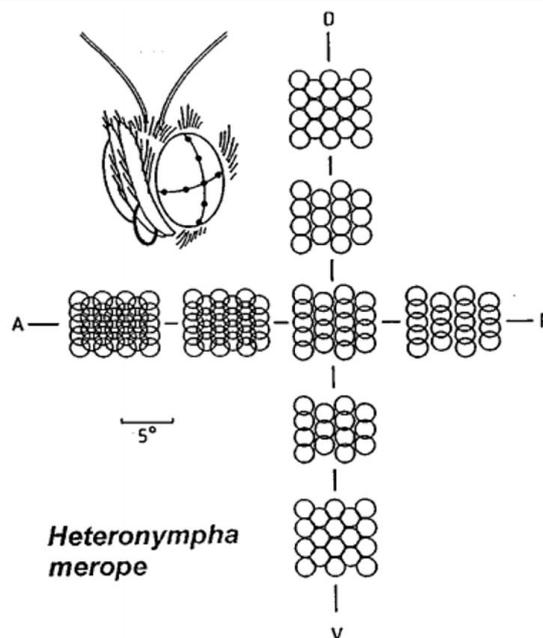

Figure 3.5: Ommatadia pattern of different regions of the eye of an Australian woodland butterfly (Heteronympha merope). A, D, P, V: Anterior, Dorsal, Posterior, Ventral. Image from Land et al. [70]

Ocelli (simple eye)
The visual system of an insect often cannot rely fully on its compound eye. It requires an extra visual sensor as support, which is the ocelli eye. The term 'ocellus' comes from the Latin 'oculus', meaning eye or little eye in zoology. These little eyes are found in most insects, co-existing next to the compound eye. Ocelli are light-sensitive organs, mostly located at dorsal (top-most) or lateral (side) positions. The amount (often three or one) and form of ocelli differ per species, where flying arthropods are often seen with larger, more sensitive ocelli [122]. Ocelli consist of a single lens and a rhabdom, where light sensitive proteins convert the photons into usable electric potential. The structure of the ocellus does not allow for visual images to be perceived, as the lens and retina are too close together. For example: the visual focal point of the ocellus of a migratory locust is five times as far as the retina, not allowing for a sharp image to be perceived [102]. Although this restriction only allows for light intensity to be measured by ocelli, their neural organisation does allow for high visual sensitivity and high speed of signal transmission [87]. It is suggested that ocelli often are used to detect instability in flight, due to their spectral sensitivity, and the temporal and spatial filtering characteristics of their neurons [141].

Neural processing of visual information in insects
Similar to the human visual system, the neural processing of visual signals in insects start in lobe-like structures in the brain. These optical processing lobes are divided into three sections: the lamina, medulla and the lobula (see Fig. 3.6). Each section performs a specific visual processing task, but all are also interconnected and often provide feedback signals to each other [15]. Axons from the eyes synapse onto the lamina, where five parallel processing streams are found (L1 – L5). The large monopolar cells (LMCs) in processing streams L1 and L2 are most prominent and largely sufficient for motion dependent behaviour. These neurons then

forward the signals towards the medulla for further retinotopic mapping. The medulla and lobula-complex sections contain greater number of neurons and also perform more complex visual processing. The medulla is thought to be involved in a variety of visual processing tasks, such as colour, motion and shape detection [109]. The lobula-complex consist of the most dense nerve structure and is thought to be mainly responsible for motion perception. The lobula plate contains large directionally sensitive tangential cells which spatially integrate the output of local motion detectors. Section 3.2.1 elaborates further on these directionally sensitive cells.
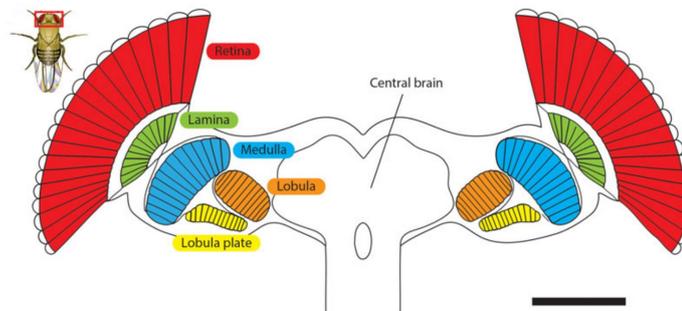

Figure 3.6: Overview of optical processing lobes of the fruit fly (Drosophila Melanogaster). Scale bar = 100 μm. Image adapted from Takemura [131].

These sections give a fundamental theoretical basis on human and insect vision. In many optics, computer vision systems are still much less efficient than biological systems. Computational efficiency, energy usage, processing speed are some of many examples where biological visual systems still outperform computer vision. Therefore, in the next section a comparison is drawn between human, insect and computer vision to make this potential gain in efficiency more tangible.

3.1.3. Comparison between human, insect and computer vision

Biological visual systems have had the advantage of many millions years of evolutionary progress to end up at the highly optimized systems they are today, highlighted in previous sections. Modern day computer vision systems are able to solve many impressive challenges, but still are orders of magnitude behind the capabilities of biological systems. Computer vision systems often exploit the benefit of having great computational power available via modern CPUs or GPUs. In contrast, biological visual systems perform the same (and often more) impressive tasks, with much less computational power, energy usage and operations. Therefore, many valuable lessons can be learned from biological principles, launching the field of research forward. This section briefly compares these visual systems on visual sensors, processing power and energy consumption to show the theoretical possible gain using a bio-inspired design.

To simplify the comparison, it will be reduced to comparing the visual systems of the human (see Section 3.1.1), the fruit fly (see Section 3.1.2) and the commonly used Parrot Bebop 2 [101]. The fruit fly is an interesting example with a relatively simple and well-mapped neural structure, but still capable of performing complex tasks such as object tracking, navigation and other visually guided behaviour. The Parrot drone is an interesting example, as it shows that there still is very large gap between robotics and biological systems. Although the Parrot drone also uses its high definition camera for entertainment purposes, it still shows the orders of magnitude difference between the computer and biological visual systems. Table 3.1 gives a summary of the comparison made below.

Visual sensor

First, a comparison between the visual sensors of the human, fruit fly and drone is made. The Parrot Bebop has a 14 Megapixel CMOS camera, which is equal to 14 million pixels. It is able to capture 1920 x 1080 pixels at 30 frames per second. Although this specific drone camera is also used for entertainment purposes, many of the visual systems in current day visually guided robotics use high definition cameras comparable to this example.

A human eye has approximately 127 million photoreceptors (rods and cones) [32], which are similar to individual pixels in a camera. The high density of photoreceptors in the fovea, enables humans to discern small details and motion in scenes.

The fruit fly retina only consists of approximately 800 photoreceptors [97], which is many orders of magnitude less than the human eye and the Parrot Bebop 2. Although this low amount of photoreceptors limits the fly its visual acuity, it is still able to fly around without hitting obstacles, navigate towards targets and perform avoidance manoeuvres (see Section 3.2.1).

Information processing and energy consumption

When considering the processing of information in these three visual systems, a comparison between the neural system of the human and fly and the computational processing of the Parrot Bebop 2 is made. Although this comparison cannot be made directly, as neural networks process information with a significantly different method, its comparison is used to show the computational efficiency. The energy consumption by these neural networks and processing units is used to emphasise the gap between biological and artificial computation. From an evolutionary perspective, energy consumption puts a constant pressure on the cost/benefit ratio of sensory and neural systems [95]. Therefore these systems have developed towards minimal energy consumption, while still processing all essential information.

In order to enable visual processing, the Parrot Bebop 2 [101] has a System on a Chip (SoC) circuit, with a dual-core Cortex-A9 CPU (which can run at 2 GHz) and a quad-core Mali-400MP4 GPU (capable of processing 2 Gpix/s). Although dependent on computational load, on average this CPU uses 3.7 W and a comparable mobile GPU uses 4.1 W [24].

Considering the human brain as a whole, it contains approximately 86 billion neurons. The human primary visual cortex (mainly responsible for visual processing) contains 140 million neurons [75]. Although in rest the human brain consumes a stunning 20% of the total energy intake [55], the visual cortex only compromises 0.16% of the neurons in the human brain. To estimate the amount of energy used by this part of the brain, an assumption of linearity between energy use and amount neurons is made. The human brain uses on average 2159 kJ per day [55], which is approximately 25 Watts. Using the ratio of neurons in the primary visual cortex and the total brain and multiplying it with the average energy consumption of the brain, the energy consumption of the primary visual cortex is gained. Therefore it is estimated that the primary visual cortex uses approximately 40 mW.

When considering the amount of neurons in the visual system of the fruit fly, only approximately 150.000 neurons are found [97]. Assuming the same linearity of energy consumption per neuron, proposed in the work of Houzel et al. [55], the fruit fly brain is estimated to use 44 μW. This shows again, the fruit fly brain is able to perform complex visual tasks with very limited resources.

From this comparison (see Table 3.1) between a commonly used drone system, the human eye and the fruit fly eye, it is concluded that there is still a large performance and efficiency gap between biological and robotic vision systems. The fruit fly is able to navigate and avoid obstacles with a very low amount of photoreceptors, neurons and energy consumption, where a drone setup still relies on heavy processing power. This brute force approach shows that there is still a lot of efficiency to gain in visually guided systems. Therefore, taking a bio-inspired approach to visual obstacle avoidance is a valid methodology.

	Visual Sensor	Visual Information Processing	Visual Information Processing Energy Usage
Parrot Bebop 2	14 million pixels	Dual-core Cortex-A9 CPU (2 GHz) & Quad-core Mali-400MP4 GPU (2 Gpix/s)	CPU: 3.7 W GPU: 4.1 W
Human	127 million photoreceptors	140 million neurons (primary visual cortex)	40 mW
Fruit fly (Drosophila melanogaster)	800 photoreceptors	150.000 neurons (visual system)	44 μW

Table 3.1: Summary of characteristics of visual systems of a typical drone, human and fruit fly.

These previous sections discussed the fundamental principles of human and insect visual systems and shows the relevance of a bio-inspired design approach. The next section will discuss more complex visual biological processing such as obstacle avoidance and nocturnal vision. This body of theory provides insights which are used in the design of the visual obstacle avoidance system on the MAV.

3.2. Biological obstacle avoidance and nocturnal animals

3.2.1. Fundamental biological principles for obstacle avoidance

ith this background on general principles of human and insect vision, this section will discuss more complex biological visual processing. In order to perceive the dynamic world, the visual system uses a complex combination of neural structures. Perceiving visual motion is fundamental to gaining environmental awareness, and is the basis for obstacle detection and avoidance. Optic flow, which is the apparent motion of the optical image, is directly related to this principle [67]. Therefore, the next sections will show the biological principles behind motion detection and some applications. The main biological motion detection theory is discussed: Elementary Motion Detection (EMD). After highlighting this theory, some examples of visually guided behaviour are given.

Biological motion detection theory

The complexity of neural connections in the eye and foremost in the brain, make it inherently difficult to exactly pinpoint brain areas responsible for certain tasks, such as motion perception. This neural structure is significantly less complex in insects such as the fruit fly. Therefore much research is conducted on flies, for example regarding motion detection theory. Dispite the complexity, there are many models describing motion detection (for a review, see Borst et al. [17]), called EMD models. When considering an image on the retina, individual photoreceptors are not able to perceive the direction of motion as they only perceive an one dimensional brightness change. Only when the signals of at least two photoreceptors are used, it is possible to discern direction of motion. This directional perception is achieved in synapses downstream, and thus is a result of neural computation. There has been extensive research on models for this computation, of which the most established one (the Hassenstein–Reichardt model [104]) is discussed below. There are extensions of this model (e.g. by Eichner et al. [36]), but this simple version is sufficient to show the purpose of the EMD models and some applications.

Directionally-sensitive motion detection

When considering directionally-sensitive motion detection, one of the simplest models proposed, consists of delays and correlation calculation. Consider two photoreceptors, A and B, in close vicinity of each other. When motion passes the two photoreceptors, they both are stimulated but with a short time interval in between. If the signal from A is delayed, and the correlation with signal B is calculated, the correlation will be strong in case of a motion from A to B. If the motion moves from B to A, the correlation will be low. This amount of correlation indicates the direction of motion. However, time delay operators are not commonly found in nature. This computation is performed, not using this simple delay and correlate scheme, but by temporal filters which will be explained below.

The directionally-sensitive motion detector can be modelled by the circuit overview in Fig. 3.7, which shows strong similarities to the simple concept explained above. Two photoreceptors A and B in each others vicinity are shown, which are excited by a moving object. R is a temporal filter, which often represents the first stages of the visual processing pathway. This includes the dynamics of the photoreceptors and the first neural processing in the retina. G and H are also temporal filters, representing higher levels of visual processing in for example the lamina, medulla and lobula complex. The outputs of A-R-H and B-R-H are multiplied with the processed signals of the adjacent photoreceptor. The 'average' box averages the signal with other photoreceptor responses, processing other image patches in the vicinity. This circuit results in the same qualitative response as the simple circuit described earlier. For example: consider an edge moving from A to B and the temporal filters G are sluggish. The multiplication of the A-R-G signal with the B-R-H signal will result in a strong positive value as response. If the object moves from B to A, this circuit will result in a strong negative response. The optomotor response (discussed in Section 3.1.2) is an example visually guided behaviour, using this directionally sensitive principle.
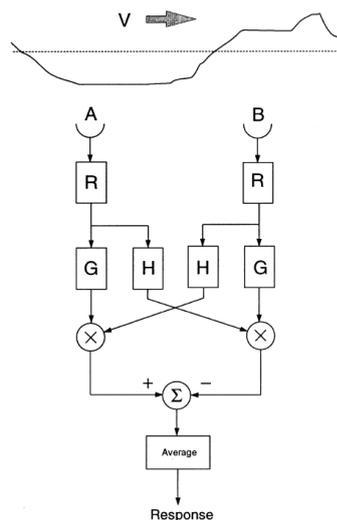
Figure 3.7: The Hassenstein–Reichardt Elementary Motion Detection model. A and B: photoreceptors, R: temporal filter representing lower visual processing, G and H: temporal filters representing higher level visual processing. Image adapted from Srinivasan et al. [125]

Directionally-sensitive cells

There are three types of cells which have been identified for vertebrates to be responsible for directionally-sensitive motion detection. This concerns the following types of ganglion cells: ON/OFF, ON and OFF ganglion cells [18]. ON/OFF ganglion cells which respond to leading and trailing edges of a stimulus. These cells are considered local motion detectors. ON ganglion cells respond only to a leading edge of a stimulus. These cells respond best to global motion and are tuned to lower temporal frequencies [142]. OFF ganglion cells are tuned to respond to trailing edges of stimulus, primarily sensitive to upward motion [62]. Including both direction and brightness information in the response of these cells would result in too large amount of information being transferred (due to the bandwidth of the optic nerve) under certain viewing conditions. Experiments of Im et al. [58] have shown that, where the directionally sensitive retinal ganglion cells in laboratory conditions respond to both direction and brightness, these cells suppress response to brightness in natural viewing, transmitting a relatively pure motion signal to the brain.

Directionally in-sensitive motion detection

Some insect behaviour has been discovered not to depend on directionally sensitive motion detection, but to be directionally in-sensitive. This has two advantages: it requires a less complex motion detection system and it allows to measure image velocity accurately. The image velocity is used in behaviour such as the landing response, peering behaviour, centring behaviour and the flight speed control (all discussed below). Also, as insects fly forward in a straight line, the direction of image motion is known but the image velocity becomes relevant to determine range. This velocity detection can be achieved with a non-symmetrical motion detection scheme (see Fig. 3.8). It also contains two photoreceptors and a temporal (low-pass) filter.

This asymmetrical motion detector is independent of the grating period (temporal frequency), which can be seen in the response in lower plot of Fig. 3.8. When using a single asymmetrical motion detector, angular velocity estimation is still ambiguous. This can be solved by using at least two motion detectors, tuned to different angular velocities (by changing the time-constant of the low-pass filter). This will result in two responses, both giving two possible angular velocities, but with one overlapping. This overlapping angular velocity is then assumed to be correct. This asymetrical system is only one of many models possible and it is thought that these symmetrical, asymmetrical and but also partly symmetrical models co-exist in biological visual systems [144].
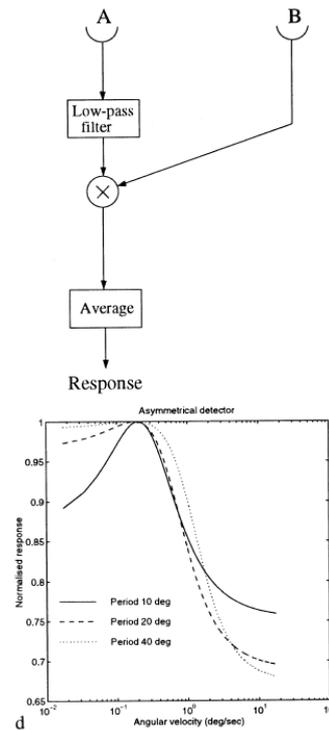
Figure 3.8: Direcitonally in-sensitive Elementary Motion Detector with response plot to a grating stimulus. A and B: photoreceptors, Low-pass filter: Representing lower visual processing. Image adapted from Srinivasan et al. [125].

Visually guided behaviour

The theoretical models described in the previous section are encountered in nature in many forms of insect behaviour. This visual processing is often coupled to motor responses, enabling insects to navigate without requiring very complex and high level understanding of the environment. Some applications of this sensory-motor coupling are described below. Following this section, obstacle avoidance principles will be discussed.

Optomotor response   The optomotor response is described as natural orienting behaviour based on the whole optical field. Reichhard et al. [105] showed that flies tend to compensate visual movement that is not a result of their own intended motion. For example: when the optical image of the fly moves to the right, the fly also tends to move to the right to compensate that image movement. The optomotor response of flies to moving striped patterns is mainly governed by the temporal frequency of the moving pattern, rather than the angular velocity. Therefore it requires a directionally sensitive motion detector.

To display this behaviour, Srinivasan et al. [125] measured the torque response of a fly mounted to a measuring device. It allows to see the different response curves to certain grating densities at different angular velocities (see Fig. 3.9). Normalizing these responses for grating densities, show that the optomotor response is angular velocity independent and spatial frequency dependent (rightmost plot). It also shows that the response is less for low and high temporal frequencies. At low temporal frequencies, the response is low because the directional information from neighboring photoreceptors becomes weaker. The response drops to zero at high frequencies because the optics become less and less effective at transmitting contrast (also due to reaching the critical fusion frequency). For optomotor response, where the direction of motion is most important, this response behaviour is sufficient.

Landing response   Flying invertebrates are able to land softly on all sorts of surfaces, with an elegant controlled motion. The landing strategy of flies has been a subject of research in the past as it is heavily based on neural optic flow processing. This principle uses mechanisms of directionally sensitive motion detection, just as the optomotor control described above. Previous research shows that landing response is based on an expanding image of the surface. The magnitude of the landing response is shown to be based on spatial-frequency, contrast of the pattern and duration of expansion [16]. Research also showed that the time of onset
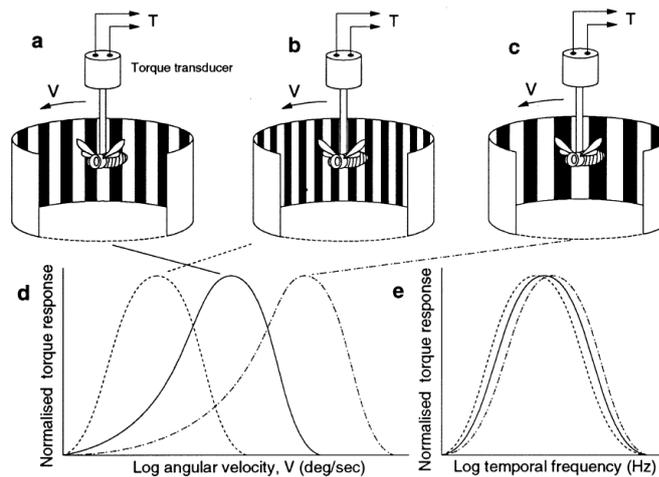
Figure 3.9: Results of optomotor response experiments by Shrinivasan et al. [125], showing torque response of a fly in a scene with different angular velocities.

of deceleration was approximately the same, regardless of the speed of approach [134]. This showed that the landing fly uses the relative rate of expansion of the image of the surface ($dr/dt)/r$, with r = instantaneous target size) to determine the time-to-contact. This time-to-contact is used to determine the onset of deceleration. From this onset, the insect holds the rate of expansion of the image constant, resulting in the smooth and controlled landing often seen at insects [7].

Centring response   As bees fly through a hole, they tend to fly as close to the centre of the hole as possible. This is a result of the centring response behaviour, controlled by the visual system. As insects lack stereo vision, balancing the distance to the edges of the hole is controlled by other visual cues. Experiments on bees flying in between two walls [126] have shown that they estimate the distance of surfaces in terms of the apparent motion of their visual images. They compare the angular speed of the images from both walls and keep those balanced. It is also shown that bees measure the angular speed largely independently of spatial period, intensity profile and contrast of the grating. In the context of the grated wall experiments, this implies that the density of grading does not affect the centring response, only the speed of the wall. Experiments have shown that bumblebees use maximum pooling of the frontal visual field in order to control their lateral position in a corridor [71].

Visual regulation of flight speed   Experiments on fruit flies [33] and bees also have shown that their flight speed is controlled by monitoring and holding the speed of which the image moves constant. The flight speed is controlled by monitoring the angular velocity, independently of the spatial structure. This behaviour is similar to the centring response. Experiments also show that optic flow is used for visual odometry: bees integrate the amount of optic flow perceived to reach a goal, to estimate distance travelled [124]. Experiments have shown that bumblebees monitor the optic flow from 23 degrees to 155 degrees to regulate the flight speed [6]. This ensures that also small objects on the collision path are detected and avoided.

Peering behaviour   Insects have, in contrast to humans, fixed focus vision. Humans can use binocular disparity (from stereopsis) to determine the distance of objects but insects cannot use these techniques. Nature has evolved certain peering strategies to overcome these restrictions. For example: the locust moves its head from side to side, before jumping towards a target. This behaviour is shown to be a means of measuring object distance [136]. It is based on a simple principle: when the observer moves its eyes, the retinal images of objects closer to the observer move further and faster than images further away from the observer (motion parallax, see Section 4.2.1). Experiments on praying mantis have shown that the speed of the retinal image governs the distance estimate [66]. The amplitude and velocity of the peering motion can be varied, but these quantities are used to determine the ratio of peering velocity to image motion velocity.

These forms of insect behaviour utilize the biological principles described by the Elementary Motion Detection models, mainly used for navigational purposes. The next section will describe biological obstacle avoidance methods.

Biological obstacle avoidance methods

To effectively avoid obstacles, an observer requires extra environmental awareness in addition to the behaviour described in the previous section. As animals move throughout the world, they constantly need to perceive and interpret information about moving objects in their field of view in order to prevent collisions. For many animals, visual information about incoming predators is valuable for planning an escape manoeuvre. This section gives a short introduction to visual depth cues, Section 4.2.1 elaborates further on this. Incoming objects (or predators) have a characteristic motion: their size in the retinal image enlarges as they approach the observer. This results in a characteristic optic flow (see Section 4.2.2 on optic flow), which can be utilized both by humans and insects. As described in the previous section, humans have an extra tool to perceive depth: binocular disparity. They are able to use the increasing binocular disparity of the retinal pictures perceived by both eyes to estimate a depth field. This measurement of change in binocular disparity is mainly used for slow moving objects, where the optic flow is utilized for fast moving objects [40]. Most insects (except the praying mantis [111]) are not able to use stereopsis to perceive an incoming object, as the field of view of each eye often does not overlap and the distance between eyes is relatively small. Therefore insects are mainly dependent on optic flow for object avoidance. Some applications of this interpretation of optic flow are Time-To-Contact, the Lobula Giant Movement Detector and saccadic movements of insects. These examples biological obstacle avoidance principles, are briefly discussed below.

Time-To-Contact   In order to effectively avoid collision, the observer requires to know what time it has left to react. This time-to-contact is a measure for urgency of the avoidance reaction, and therefore implicitly also the velocity of the reaction. The time-to-contact can be estimated if the distance and relative velocity between the observer and the object is known ($\frac{Z}{\dot{Z}}$ =Time-To-Contact (TTC), with Z = relative distance of the object, $\dot{Z}$ = relative velocity between the object and observer). Both these quantities are not directly available to the observer and a different approach is required. The expanding image of the object on the observers retina is able to give a ratio of two types of information: the image size and the rate of expansion of the image. This ratio is able to give an accurate estimate of the time-to-contact, without the requirement for the absolute speed or distance. The usage of this ratio is encountered in nature, for example in plummeting gannets. They streamline their wings at a fixed time-to-contact from the water, avoiding injury to their wings [73]. This usage of the time-to-contact is also shown to be used by humans: breaking in a car [72], playing table tennis [14] or hitting an accelerating ball [74].

Lobula Giant Movement Detector   The Lobula Giant Movement Detector (LGMD) is a wide-field visual neuron located in the Lobula of the Locust (see Fig. 3.10), and its neural structure is expected to exist in other insects as well. It is assumed that this neuron triggers an escape reaction, for example when a locust is attacked by a predator [118]. This neuron increases its firing rate according to the velocity and proximity of an incoming object. It is tuned to respond to objects on a direct collision course, but has little to no response to receding objects [59]. The peak of the firing rate is located at or close to the projected time of collision [43]. The descending contralateral movement detector (DCMD) is its postsynaptic partner, and connects with motor neurons associated with jumping and flight steering manoeuvres [22].

Saccadic movements to induce optic flow   The flight trajectories of flies and bees have been shown to consist of straight flight sequences, followed by a turn (called a saccade). These intersaccadic movements, varying from 20 ms to 200 ms, are purely translational. It is suggested that these translational movements are used to obtain optic flow using Lobula-Plate Tangential Cell (LPTC)-neurons [61]. The peak angular velocity and succession into either direction are variable and depend on the visual surroundings. Saccade rate and amplitude also are correlated with the time-to-contact to the frontal objects such as walls. By increasing the duration of an intersaccade, the dependency on the texture in the environment can be decreased, assuming the visual integration time is increased accordingly. The intersaccade duration is possibly linked to the collision avoidance necessity. If there are no obstacles in the neighbourhood, there is no need for collision avoidance and long intersaccades are possible. However, if there are obstacles in the vicinity, collision avoidance could be necessary and short intersaccades followed by an evasive turn might be required. Saccade and
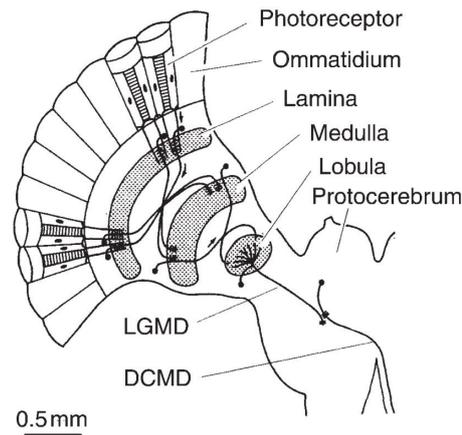
Figure 3.10: Location of the Lobula Giant Movement Detector (LGMD) and Descending Contralateral Motion Detector (DCMD) in the locust visual pathway. Image adapted from Rind et al. [108]

velocity control rely to a large extent on the intersaccadic optic flow generated in eye regions looking in front of the fly [61].

The previous sections gave insight into biological motion detection and obstacle avoidance principles, encountered in humans and insects. These principles are used as inspiration for the obstacle avoidance system onboard the MAV. As described in previous sections, the visual system of humans and insects consists of tightly tuned parts, which allow for sight under various conditions. In order to apply the MAV obstacle avoidance system in low-light conditions, the visual system has to be tuned to these conditions. Therefore, the next section will introduce the fundamental principles of biological night vision.

3.2.2. Defining low-light vision
Many years of evolution have caused the visual system of animals to be tightly tuned to allow for extraordinary high performance under various conditions. As described in Section 3.1.3, current computer vision systems still rely on heavy computational power while biological systems elegantly perform complex processing tasks. When considering low-light conditions, this becomes even more important. Much of the current research in MAV design is performed in laboratory light conditions (often using bright white lights). This is convenient for testing, but does not allow for these systems to be applied outside of the laboratory environment. To create an MAV obstacle avoidance system that is capable of performance under various conditions, including low light, inspiration is taken from animals living in these conditions: nocturnal animals. The next sections will introduce the definition of these conditions, and the principles enabling vision in low-light intensity.

Nocturnal animals and low-light conditions
The life pattern of many species such as humans, animals, plants and microbes depend on the day and night cycle. As humans are mostly restricted to a diurnal (living by daytime) lifestyle, the vast majority of research is also conducted for daytime conditions. As nocturnal (living by night) conditions push animal sensory capabilities to the limit, this is still a very interesting field of study. Many valuable lessons and concepts can be learned from nocturnal animals, including sensory and processing techniques.

Nocturnal behaviour   Animals that inhibit nocturnal behaviour are characterized by being active during the night and sleeping during the day. The leading hypothesis for this behaviour is thought to the 'bottleneck theory' [53]. During the Mesozoic Era (± 250 - 65 million years ago) the ancestors of mammals avoided diurnal (living by day) saurian predators by living at night and hiding (and sleeping) by day. They also did not compete for the same resources as diurnal animals. During this Era these mammals evolved a variety of sensory adaptations to support their new night life. This included for example: improved hearing [28] and higher visual sensitivity. It is thought that at the same time, many of these animals lost the capability for photopic and tetrachromatic colour vision [53].

In order to distinct between different light conditions, an international definition of luminance levels is set. An international standard specified by Commission Internationale de l'Eclairage (CIE) contains three vision levels: photopic, mesoscopic and scotopic vision [50].

- 'Photopic vision' = From English photo, from Greek phos, meaning "light" or "produced by light". Occurs at luminance levels of 10 to 10^8 cd/m^2 (or Lux).

- 'Mesoscopic vision' = from Greek mesos, meaning "middle". Occurs at luminance levels of 10^{-3} to 10 cd/m^2 (or Lux).

- 'Scotopic vision' = From Greek skotos, meaning "darkness". Occurs at luminance levels of 10^{-6} to 10^{-3} cd/m^2 (or Lux).

This thesis mainly focuses on the mesoscopic and scotopic conditions (see Fig. 3.11) for MAV navigation, as these conditions are highly demanding for the visual system and pushes its sensory capabilities. The next sections will introduce visual noise and principles that allow for vision under these low-light conditions.
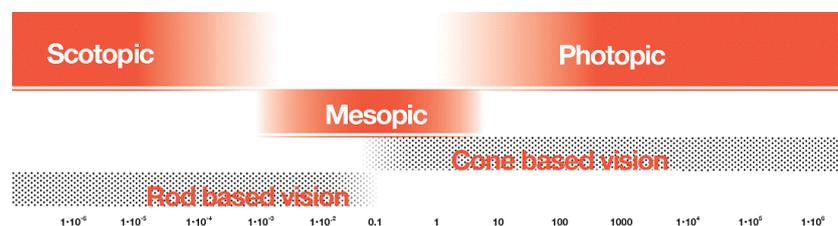

Figure 3.11: Luminance spectrum from scotopic to photopic vision (measured in Lux). Rods and cones in the human visual system are sensitive to both low and high light intensity. Image from 'Azo materials' [84].

3.2.3. Noise in visual systems

At night or indoors, often there is only a small amount of photons available for photoreceptors (or digital sensors) to capture. A sufficient Signal to Noise Ratio (SNR) is required to obtain valuable information from the scene. Therefore, identifying the main causes of visual noise is valuable for designing low-light systems. The causes of visual noise for biological and computer vision systems overlap in certain areas. The quantum behaviour of photons and thermal agitation are a cause of noise in both biological and computer vision systems. In biological systems, spontaneous biochemical reactions also cause noise. These different forms of visual noise are described below.

Photon shot noise   Light travels in discrete photon packages, which are subject to quantum effects. This implies that light behaves stochastically. As photons hit the photoreceptor, a discrete amount of electrons is generated by the photosensitive proteins. When measuring either the amount of photons or electrons, the measured quantity is an integer number drawn from a probability distribution determined by the emission source [52]. This causes the intensity of the light to vary, even at a constant source brightness. This effect is mainly noticed when there is little signal that overrules the noise (thus at a low SNR). If the effect of photoreceptor bleaching in eyes can be ignored, photon absorption is described with a Poisson distribution [77]. Fig. 3.12 shows an example of photon absorption at four different light levels. In the first two light intensities the black center circle cannot be discerned. At the third light intensity, its shape is still uncertain and only with the fourth light intensity, it discerned for certain.

Thermal noise   Thermal noise (also called Johnson–Nyquist noise [96]) is a result of the thermal agitation of electrical charge carriers, electrons. The very small movement of the electrons cause an electrical flow which is perceived by the visual system as a signal. Increasing the temperature also increases the amount of thermal noise. It is present in all electrical systems, therefore also in the signal transduction pathway of biological visual systems and in computer vision sensors. It has been shown that a lower biological body temperature results in less of these noise events, which improves visual sensitivity [2].

Biochemical retinal noise   Another form of visual noise originates from the biochemical processes leading to signal amplification. The biochemical pathways, responsible for picking up the photon and signalling the brain, are occasionally activated resulting in so called 'dark light'. This is due to spontaneous conversion of the rhodopsin protein or spontaneous activation of transducin in the signal transduction pathway. This triggers the ganglion cells to fire a burst of spikes, resulting in a false positive measurement [9]. Other causes of noise in visual pathways originate from synaptic transmission (in bipolar synapses [42]) and the spike generation processes [133]. These processes also are subject to quantum effects, therefore adding perceived visual noise. These two noise types set the ultimate lower limit for stimulus detection. These noise levels differ per animal, where some insects experience approximately 1 event every 10 hours, nocturnal toads experience for example about 360 events every hour.
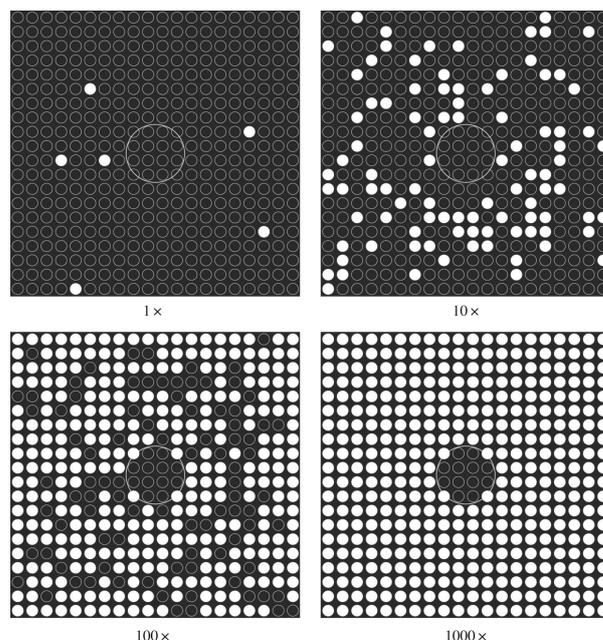

Figure 3.12: Example of the combination of spatial resolution and light level. 400 Photoreceptors are set in a matrix, shown as small circles. The black disc at centre cannot be discerned in the first two light levels ($1\times$, $10\times$) as it is disguised by noise. Image adapted from Warrant [138].

3.2.4. Animal vision in low-light

Where the humans visual system is highly limited in low-light conditions (e.g. moonless night), a great amount of animals are able to walk, crawl, fly and swim around without effort. Even in the deepest seas, animals are still able to manoeuvre their way around, for example using very faint light from bioluminescence [140]. The visual systems of these animals are greatly adopted to low-light conditions by evolutionary development. Nocturnal animals rely vitally on these evolutionary benefits and are therefore an interesting field of study. The following section will highlight some of the principles seen in nature which enable vision in low-light intensities.

Before discussing the principles that allow vision in low-light, a quantification of visual sensitivity is discussed. As visual sensitivity is hard to quantify, Michael Land proposed a tool [65] which enables sensitivity comparison between species and can quantify the vast majority of camera and compound eyes correctly [41]. The following equation quantifies optical sensitivity:

$$S = \left(\frac{\pi}{4}\right)^2 A^2 \left(\frac{d}{f}\right)^2 \left(\frac{kl}{2.3 + kl}\right) \tag{3.2}$$

In this formula, 'S' is the sensitivity of the eye in units of μm^2 steradians (steradian = unit of solid angle), 'A' is the diameter of the eye its pupil (or aperture), 'd' is the diameter of the photoreceptor (rhabdom for insects), 'f' is the focal length, 'k' is the absorption coefficient of the photoreceptor and 'l' is the photoreceptor length. When these length units are in micrometres, the unit of 'k' is μm^{-1}.

When considering nocturnal animals, a great variety of visual characteristics are seen which allow for excellent night vision (for a review see the work of Warrant et al. [137]). These principles can be divided in sensory and neural processing characteristics. Sensory characteristics such as wider pupils, wider receptive fields, shorter focal lengths and a tapetum lucidum are related to the physical structure of the eye. Spatial pooling, time integration and increased gain of rod responses are related to the neural processing of the information received by the eye. These principles will be discussed briefly below.

Wider pupils and larger eyes   Nocturnal animals often have wider pupils to allow more light to fall onto the retina. This increases photon capture and therefore visual sensitivity. Examples of vertebrates who use this principle are owls and tarsiers (with pupil diameters of approximately 2 cm). This is reflected Eq. 3.2: increasing the diameter of the eye pupil 'A' will increase the visual sensitivity. The most distinguished example of this feature is part of the cephalopods: the giant deep-sea squid (with a pupil diameter of approximately 36 cm) [69]. Some deep-sea fish go to even greater lengths to catch the dim light at the bottom of the sea. For example: the Grimatroctes microlepis has a significantly larger pupil than its lens (an aphakic gap), resulting in more light being able to enter the eye past the lens. This light will be defocussed but still significantly improves sight [139]. Some nocturnal animals have relatively large eyes to maximize visual sensitivity. Within the group of nocturnal animals, the visual predators (e.g. Tarsius or Loris) tend to have larger eyes than diurnal primates [63] [48]. This feature also results in more photons reaching the retina, which allows for higher visual sensitivity.

Wider receptive fields   If the receptive field of the photoreceptors is widened (therefore viewing a large solid angle), the scene that the visual channel views is also broadened. This also results in more photons being captured. This is solid angle is reflected in Formula 3.2 in $\pi d^2/(4f^2)$ steradians. Increasing the viewed solid angle by a photoreceptor therefore also increases visual sensitivity.

Shorter focal lengths   The focal length determines the distance between the retina and the lens in order to achieve a focused image on the retina. Nocturnal animals on land often have a highly powerful curved cornea, which allows the lens to be closer to the retina. This implies that the focal length 'f' is smaller and thus visual sensitivity is increased (see Eq. 3.2). A smaller focal length relative to the pupil diameter 'A' results in a large viewed solid angle. For nocturnal vertebrates this often results in a tubular eye shape, allowing a short focal length [119]. These tubular eye shapes are seen for example in owls, which have great scotopic visual sensitivity [83]. Short focal lengths are also encountered in nocturnal spiders with extremely sensitive eyes e.g. the net casting spider 'Deinopis subrufa'. These spiders utilize a combination of a very large lens, short focal distance and large diameter of its photoreceptors to achieve vision that is 2000 times as sensitive as the human eye [13].

Tapetum lucidum   Often, when a bright light is shined on the eyes of nocturnal animals, it seems that their eyes 'light up', a phenomenon known as 'eye-shine'. This is due to the tapetum lucidum, a reflective layer in the retina of many nocturnal animals [98]. This reflective layer is often positioned behind the photoreceptors, reflecting unabsorbed light a second time back through the photoreceptors. This effectively doubles the length of the photoreceptor, increasing visual sensitivity in dim light. The tapeta have been developed evolutionary within different invertebrates and vertebrates species but show surprisingly similar light reflection mechanisms. The tapetum reflects wavelengths most relevant to each species behaviour [120].

Pooling of rod responses (spatial summation)   Nocturnal animals make use of spatial summation of rod responses to increase visual sensitivity. The number of ganglion cells (which forward action potentials to the brain) is much lower than the amount of rods that can be triggered. This implies that many rods converge to a single ganglion cell, determining the local spatial resolution [29]. A trade-off is made between visual sensitivity and visual resolution (with a dense area of ganglion cells). This spatial summation is found in many nocturnal animals, such as in bees [51], cockroaches [107] and moth [129].

Time integration of rod responses (temporal summation)   Just as stimuli are summed spatially, a temporal summation also is able to increase visual sensitivity (see Fig. 3.13). It can be compared to the shutter time of a camera: light is absorbed over a longer period before processing, increasing visual sensitivity. This time integration comes at the cost of the response time. Many nocturnal animals use this long time integration.

For example: the common toad (B. bufo) has a neural integration time of approximately 1.5 seconds [35]. This allows it to see under scotopic conditions but highly limits its response time. Therefore high integration times are more often seen at animals who do not require fast and agile movement.

Slow response time with increased gain   As described above, nocturnal animals use spatial and temporal summation to increase their visual sensitivity. These concepts are also often used in combination with a higher transduction gain. If photoreceptors send stronger responses (higher voltage potentials) to stimuli, this results in a higher contrast gain. The downside of this high gain is that noise (discussed in Section 3.2.3) will also be amplified. Thus on itself, this increased gain does not increase visual sensitivity. As the visual noise from each photoreceptor or ommatidia is uncorrelated, the noise is averaged out spatially. This is where the spatial summation is beneficial again, to suppress visual noise, amplify the signal and therefore increase visual sensitivity [129].
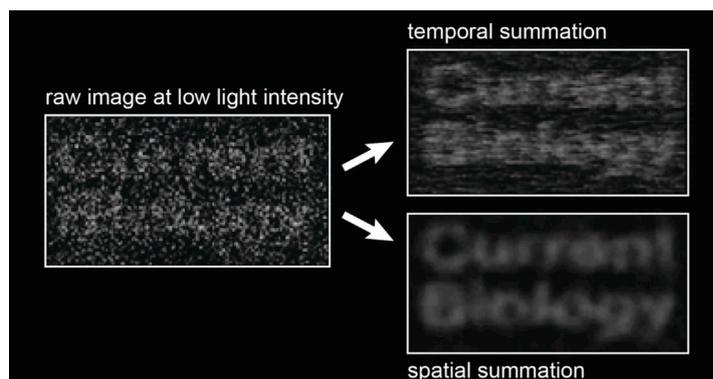

Figure 3.13: Example of temporal and spatial summation of an image at low light intensity.
Image adapted from Stöckl et al. [129]

These biological principles described above all contribute to increasing the Signal to Noise Ratio (SNR) of the visual system. Some principles focus on maximizing the received amount of photons (such as wider pupils or receptive fields), where others focus on minimizing the effect of visual noise (such as neural pooling). All visual systems, including human, animal or computer vision, are subject to different forms of noise. As low-light conditions imply that there is little signal available, the SNR subsequently will be low. Therefore, in the next section different forms of visual noise will be identified.

Previous sections gave insight into the biological principles behind obstacle avoidance and vision in low-light conditions, from a biological perspective. The following section will synthesise this literature review, into concepts usable for the obstacle avoidance system design.

3.3. Synthesis

To highlight the most relevant findings from the literature review regarding biological principles of obstacle avoidance and low-light vision, this section will synthesise the biological research into important insights. The relevant research questions and answers are also given.

Why a bio-inspired approach to obstacle avoidance by insects? (Section 3.1)

Section 3.1 showed the motivation for using a bio-inspired design, based on insect vision. First, a general introduction to the human and insect visual system is given. From this introduction is concluded that the insect visual system is able to provide more valuable insights, relevant to this research, than the human visual system. Therefore, the following sections focused on insect obstacle avoidance and low-light vision.

The non-uniform distribution of ommatidia on the compound eye, discussed in Section 3.1.2, shows an interesting insight. This principle enables insects to have more visual acuity on certain areas of their visual field. The output from the visual sensor on the MAV has to be down-sampled (due to the high output rate, see Section 4.1). This non-uniform down-sampling encountered at insects can be applied to maintain the visual acuity in important areas of the field of view (e.g. supporting forward flight), while reducing the visual acuity in other less important areas. This reduces the computational load while maintaining relevant visual acuity.

As summarized in Table 3.1, the visual system of insects (here: the fruit fly) is orders of magnitude less complex on all categories than human or the commercial drone. Insect show a low amount of photoreceptors, neurons and power usage while still being able to fly and navigate effortless. This shows that a bio-inspired approach is valuable and it provides a general sense of the possible efficiency gain.

What biological principles are fundamental for obstacle avoidance? (Section 3.2.1)

To gain inspiration for the obstacle avoidance system design, Section 3.2.1 reviewed the Elementary Motion Detection (EMD) theory and showed examples of visually guided behaviour. The EMD theory is shown to be valid for humans and insects. This is still a high level abstraction (representing neural structures as filters) of the actual processing, but serves as a fundamental theory for the examples of visually guided behaviour. Motion detection is found to be processed by directionally-sensitive cells, which respond to certain stimuli such as leading and/or trailing edges on the retina image. The pixels of the event-based camera used as visual sensor in this research, shows great similarity with these cells.

The visually guided behaviour shows interesting concepts which are useful for the obstacle avoidance system design. The following concepts provide valuable insights:

- Time-to-contact (TTC)
  This principle is useful as it only depends on the flow of the image and in theory can provide an accurate time to collision. TTC can be estimated using a monocular camera, which is a highly limiting factor for the MAV design. Therefore, this concept is used in the system design.

- Centering response
  By balancing the optic flow in both sides of the image, insects are able to fly through the center of a corridor. This simple strategy can only be applied in corridors. Therefore it can used as inspiration for a solution in corridor environments during the system design.

- Peering behaviour
  This is an interesting concept as it perceives depth from motion, using the motion parallax. As this research uses a monocular camera, stereopsis cannot be used to perceive depth. Therefore, utilizing the motion parallax can be a valuable approach.

- Saccadic motion
  Flying in straight sequences, followed by a turn, can be an effective strategy. As the event-based camera (used in this research) only triggers on motion, this strategy might provide a controlled method to induce motion in the scene.

What biological principles are fundamental nocturnal vision? (Section 3.2.4)

In low light conditions, biological visual systems are subject to three types of noise described in Section 3.2.3: photon shot, thermal and biochemical retinal noise. The first two are relevant to this research as they also are encountered when using the event-based camera in low-light conditions. To increase the signal-to-noise ratio, the signal can be strengthened or noise can be suppressed. Photon shot noise can be described with a Poisson distribution and can be partially suppressed by using filters. Thermal noise can be suppressed by lowering the temperature of the event-based camera. Lowering the temperature of the system is a strategy also encountered in nature (see Section 3.2.3).

To gain inspiration for improving the event-based camera system in low-light, Section 3.2.4 reviewed the visual system of nocturnal animals. The biological characteristics that enable vision in low-light are divided into two categories: sensory and neural processing characteristics. All sensory characteristics are not feasible to implement, as hardware modification to the event-based camera would be necessary. Neural processing characteristics can be used as inspiration for the processing pipeline of the obstacle avoidance system. The following characteristics are potentially valuable:

- Spatial summation
  By pooling rod responses, nocturnal animals gain increased visual sensitivity in low-light. This biological principle effectively reduces the visual acuity, while increasing the visual sensitivity. In the context of the obstacle avoidance system, this would result in reducing the resolution of the event-based camera by pooling triggered pixels (or optic flow vectors). This can be used in combination with the proposed non-uniform down-sampling, described above.

- Temporal summation
  Time integration of rod responses is also an often encountered characteristic at nocturnal animals. It is comparable with using a time surface image, as it also accumulates events over time. In the context of the obstacle avoidance system, this strategy might cause issues due to the fast and agile motion of the MAV.

- Slow response time with increased gain
  Combining spatial and temporal summation is often encountered in nature. Amplifying visual signal on itself does not increase visual sensitivity. In combination with reducing the noise by spatially averaging the signal (as the individual pixel noise is uncorrelated), this can improve low-light vision. This is concept of spatio-temporal summation, in combination with increased gain, can be used for the obstacle avoidance system design.

This concludes the chapter on biological inspiration from nocturnal insects. The central research questions on biological obstacle avoidance and nocturnal vision are answered. This fundamental theoretical knowledge is as inspiration in the system design, resulting in a bio-inspired obstacle avoidance system. The following chapter will review previous research on MAV obstacle detection and avoidance methods.

4

Obstacle Detection and Avoidance methods

To design a novel obstacle avoidance method, inspired by the concepts from Chapter 3, previous research on event-based cameras and state-of-the-art obstacle avoidance methods is reviewed. As the research goal is to fly fully autonomous in unmapped, GPS denied environments, the MAV is limited by its Size, Weight and Power (SWAP). As all processing for sensing and navigation has to be performed on board the MAV, there are strict restrictions on what is feasible. This chapter will discuss the chosen visual sensor (the event-based camera) and its processing in Section 4.1. Following this, previous research on obstacle avoidance algorithms is discussed in Section 4.2. Section 4.3 will provide a synthesis of the most useful concepts and methods from this review.

4.1. Event-based vision

Flying MAVs in unknown environments such as disaster scenes, dark indoor spaces and other hazardous circumstances is challenging. Often there is no prior information about objects in the environment, there is no map available, lighting conditions are uncontrolled and the MAV cannot return easily to recharge the battery. This requires the MAV to perform with low latency, a high visual bandwidth and with low power consumption. Conventional cameras often perform poorly in these challenging conditions due to their all-purpose design.

A revolutionary new bio-inspired camera design was released in 2008 by Lichtensteiner et al. [76] which addressed the majority of these challenges. They released the first commercially available event-based camera, which is able to perceive changes in brightness (and thus capture apparent motion). Unlike conventional cameras, the event-based camera is triggered asynchronously on an individual pixel level by brightness change. The intensity change threshold which triggers the pixel is user-defined. This results in the event-based camera not having a set frame-rate, but rather asynchronously outputting a stream of events. The events are labelled by the pixel location, trigger time (in microseconds) and sign of the intensity change.

Section 4.1.1 will describe the physical principles which characterize the event-based camera. Subsequently, Section 4.1.2 will discuss the benefits of using an event-based camera with respect to a conventional camera. The bio-inspired design is highlighted in Section 4.1.3. As the output of the event-based camera is significantly different from a conventional camera, Section 4.1.4 will highlight the main challenges and methods for processing the event data stream.

4.1.1. Design of the event-based camera

The event-based camera uses a novel method to capture the motion in a scene. It does not take the brightness values at all pixels at a set framerate (such as a normal camera would), but its output rate is rather determined by motion of edges in the scene. An event-based camera pixel triggers on brightness changes in the scene, asynchronously and independent of other pixels. This asynchronous output of events is inspired by the spiking of action potential seen in the biological visual pathways (see Section 4.1.3). Light intensity is

not perceived linearly by humans, but is rather characterised on a logarithmic scale. Each pixel continuously monitors the log intensity, and compares it to the last measured value at that pixel. If the change in log intensity exceeds a certain threshold (which is user defined), the pixel triggers and sends a x, y location, the time t and the polarity of the change (see Fig. 4.1). The DAVIS camera, used for this research, also includes an Active Pixel Sensor (APS). This frame-based sensor shares the same photodiode as the DVS in each pixel, and allows to capture intensity readouts. This is used to create an intensity image at a constant frame rate.
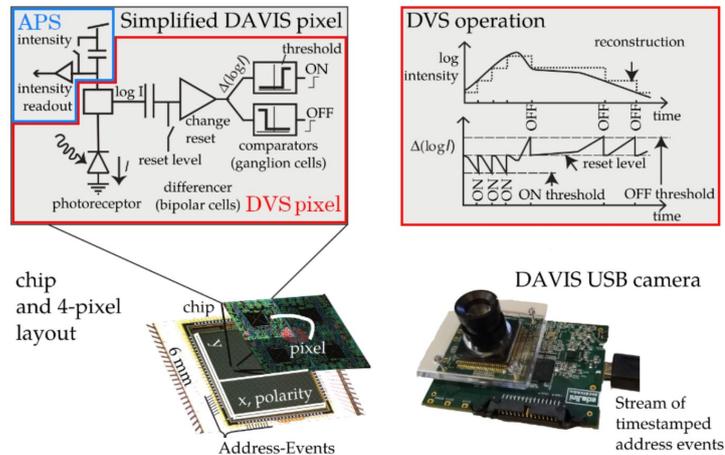

Figure 4.1: Left image: simplified circuit diagram, red = DAVIS circuit, blue = APS circuit.
Right image: example of triggering of a single DVS pixel. Image adapted from Gallego et al. [46]

Let t_{k-1} be the last time when an event triggered at pixel location x. Let $L_{k-1} = L(x, t_{k-1})$ be the log intensity level at that pixel location at t_{k-1}. A new event is triggered at the pixel location at t_k if the difference in log intensity between L_{k-1} and L_k is larger than an user defined threshold $C > 0$. Thus a new event is triggered if:

$$
\begin{aligned}
||L(x, t_k) - L(x, t_{k-1})|| > C &\quad \text{(positive event)} \\
||L(x, t_k) - L(x, t_{k-1})|| < -C &\quad \text{(negative event)}
\end{aligned}
$$

(4.1)

To emphasize the novelty of this new approach to visual sensing, the next section will compare the characteristics of the event-based camera to a conventional camera.

4.1.2. Characteristics of the event-based camera

As the event-based camera opens up a new paradigm in visual sensing, a comparison between conventional frame-based cameras and event-based cameras is made. Conventional frame based cameras often face motion blur under fast moving scenes, and under or over sampling, resulting in too much or too little produced data. The event-based camera is scene driven, and thus modifies its sampling rate to the motion present in the scene. A comparison of the main characteristics of the event-based camera with respect to a conventional camera is made below.

- **Very fast sampling (ranging from 2 MHz to 1200 MHz of events).** Events are timestamped with microsecond resolution and are transmitted with sub-millisecond latency, which make these sensors react quickly to visual stimuli.

- **Asynchronous sampling, resulting in sparse data.** As the camera only triggers on motion, a sparse representation of the scene is obtained. Where conventional cameras capture all intensities of all pixels in the frame in a set frame rate (e.g. 24 frames per second), event-based cameras only trigger on object edges that traverse the pixels.

- **Low power consumption.** The event-based cameras use only 10 mW (including processing: 100mW) [46], where conventional cameras often need a few Watts for operation.

- **High dynamic range.** Event-based cameras have a very high dynamic range (> 120 dB), allowing usage under varying lighting conditions. The light that hits a pixel is the result of the product of scene illumination and surface reflection. If the log intensity of the scene changes, that is generally due to a change

in surface reflectance (e.g. caused by object motion). The illumination in a scene is often constant. Resulting in: log((scene intensity) × (object reflectance)) = log(scene intensity) + log(object reflectance), as the log of a product is the sum of the logs. Therefore event-based camera has a built-in invariance to scene illumination.

- **Bandwidth (compared to 60 FPS camera).** Event-based cameras also outperform conventional cameras under varying light conditions. To show the physical limit of the event-based camera, its response was measured by Lichtsteiner et al. [76] under different motion frequencies varying from low to high (see Fig. 4.2). Above a cutoff frequency, the physical photoreceptor dynamics filter out the change in intensity, resulting in a dip in performance. When using bright light, the DVS pixel bandwidth is approximately 3 kHz which is comparable to a shutter speed of 300 microseconds. When reducing the light intensity by a factor 1000, the bandwidth is still about 300 Hz which is still 10x higher than the Nyquist frequency of a 60 FPS camera (30 Hz). Events are generated reliably and reproducibly down to less than 0.1 lux of scene illumination [76].
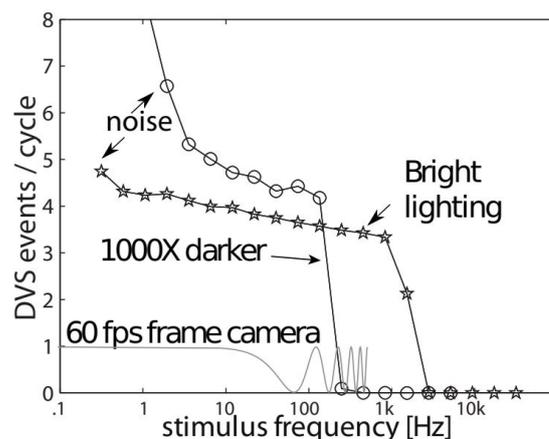

Figure 4.2: Measured responses of DVS to sinusoidal LED stimulation for two DC levels of illumination. Figure adapted from Lichtsteiner et al. [76].

4.1.3. Bio-inspired design of the event-based camera

The event-based camera has a biologically inspired design, based on the spiking of neurons in visual pathways. The visual system can be divided into two information processing streams: 'where' and 'what' pathways. The dorsal area is related to spatial vision ('where') and the ventral area is related to object recognition ('what'). The DVS design is related to 'where' pathway, as it encodes spatial information. In comparison, grayscale events can be corresponded to the 'what' pathway. The DVS fulfils the role of directionally selective (DS) cells in first order motion perception, described by models from Hassenstein and Reichardt (see Section 3.2.1). The DVS outputs a location and ON or OFF signal, indicating respectively intensity increase and decrease at that pixel. This principle is analogous to the ON/OFF directionally selective cells in the visual system, which fire on leading or trailing edges of a stimulus.

These characteristics enable an unique interpretation of the scene. Although the event-based camera has impressive specifications, the novel design also requires a new form of digital processing. Conventional computer vision methods cannot be used directly on the event stream and the sensor also produces noise and non-idealities. Therefore the next section will discuss the challenges and solutions for data processing of the event-based camera.

4.1.4. Event stream data processing

The asynchronous and binary event-based camera output is an unique representation but also a challenging one. Conventional computer vision methods rely on frame based video input, and all processing pipelines are designed accordingly. Therefore old processing pipelines are re-designed or novel methods are created to handle the event stream. The three main challenges on event data processing are described below.

- **Different space-time output than conventional cameras (asynchronous)** The decoupling (or non-existence) of the frame-rate also requires the processing pipeline to handle asynchronous data (see Fig. 4.3). This is often resolved by transforming the event stream back to conventional computer vision representations (such as time surfaces or 2D frames). This often causes loss of the unique space-time representation, and therefore loss of scene information. Others approach the asynchronous event stream event by event with neuromorphic computing, keeping the characteristic event stream intact (see Section 4.2 for examples of neuromorphic applications).

- **Different photometric sensing (binary increase or decrease signal instead of greyscale info)** As the event-based camera is triggered on intensity change, it only outputs the sign of the change (+ or −). Therefore brightness change is also dependent on current and past motion of objects. This results in a fundamentally different image than from conventional cameras, which output absolute intensity level per pixel. This also requires a non-conventional approach to event processing.

- **Inherent shot noise in photons and electrical circuits** Digital visual sensors are also subject to photon and electrical shot noise (see Section 3.2.3). As the event-based camera captures change in intensity, it also captures the fluctuation in intensity due to the stochastically arriving photons. These effects has been characterized for conventional cameras, but not yet for event-based cameras.
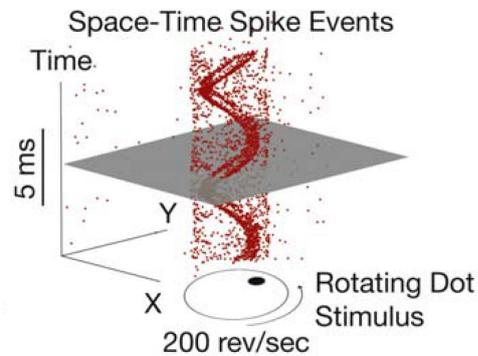

Figure 4.3: Visualization of a rotating dot, captured by an event-based camera.
This shows the asynchronous output of the camera. Image adapted from Liu et al. [79]

Event representations and processing methods

The event-based camera outputs an asynchronous stream of events, which has to be interpreted to obtain valuable information about the scene. There are several event representations and processing methods used in literature. The main representations and processing methods used in previous research are listed below.

- Individual events
  $e_k = (xy_k, t_k, p_k)$ events are used in event by event processing schemes such as Spiking Neural Networks. The disadvantage of using individual events is that they do not contain information about the underlying motion and are subject to noise. Often, deterministic filters (e.g. for noise reduction or feature extraction) or probabilistic filters (e.g. Kalman filters) are used as they can handle asynchronous data.

- Event packages
  Neighbouring events can be grouped based on spatial-temporal information. When processing events in groups, it is possible to improve the SNR. The number of events per group strongly influences the performance of the algorithm. Two main strategies are often used: a constant number of events, or a constant observation time. Using a constant number of events that are assigned to a group, fits better with the asynchronous output of the camera. Using a constant observation time to assign events to a group, causes a variation in the amount of events per frame, which could cause issues further down the processing pipeline.

- Event image or 2D histogram
  Events in each others spatial-temporal neighbourhood can be mapped to a simple 2D frame. This

results in the loss of the asynchronous stream and data sparsity. This method is often used as it converts the event stream to a format that can be interpreted by conventional image-based algorithms such as Convolutional Neural Networks.

- Time surfaces
  This method maps all motion to a 2D frame and decayes older motion. The intensity of the image is the amount of motion history that pixel has had, with brighter pixels corresponding to more historic motion. This is called 'Motion History Images' in the computer vision domain. The effectiveness of these time surfaces is less in very textured environments, as pixels spike frequently and overwrite itself on motion.

- Voxel grid
  This representation can be seen as a 3D histogram, effectively dividing space-time up into boxes. Each box represents a certain space and time interval. This is a form of down sampling, and a discretization of space-time.

- 3D point set
  When time is considered the third spatial dimension, an event stream can be represented as a 3D point cloud. This is used for point-based geometrical processing methods such as plane fitting.

- Point set on image plane
  Events can also be represented by an evolving image of points on a 2D image. This is mainly used by early vision algorithms that track edges and shapes.

- Motion compensated event images
  When moving an edge over event-based camera pixels, the edge triggers pixels along its motion. This motion can be compensated by warping events to a reference time and aligning them such that a focussed image is produced. This was first proposed by Gallego et al. in 2018 [45] and the focus framework later improved in 2019 [47]. This mapping creates a more familiar representation of visual information, by showing the map of edges in the event stream. Section 4.2.2 also elaborates on this approach.

With this knowledge from previous sections on operational principles and data processing for the event-based camera, the next section will discuss state-of-the-art research in event-based obstacle avoidance. The following section will highlight the key concepts for obstacle avoidance in the robotics domain and event-based obstacle detection and avoidance methods.

4.2. Event-based obstacle avoidance methods

MAVs are able to fly agile through various environments, performing vertical take-offs and landings, hovering and other movements in 6 degrees of freedom. The environments in which they are flown are often unknown, dynamic scenes who require environmental awareness to prevent collisions. To gain awareness of the environment, sensors such as cameras, ultrasonic sensors or light detection and ranging (LIDAR) systems can be used. The Size, Weight and Power (SWAP) restrictions of the MAV also limits the sensors and accompanying processing which can be used to gain this environmental awareness. Active sensors (such as LIDAR) often weigh more and require more power than passive sensors (such as a camera sensor). Due to these considerations and following the bio-inspired approach, it is decided to use a monocular visual sensor on the MAV. Therefore, all obstacle detection methods described in this section will focus on the use of a monocular camera. First, visual cues which allow the observer to sense depth are discussed, including the fundamental principles of optic flow in Section 4.2.1 and 4.2.2. The focus of expansion (FOE) is discussed afterwards in Section 4.2.3. With this background on optic flow and FOE estimation, a review of previous research in obstacle avoidance using monocular cameras is performed. This review is split into two types of methods, based on optic flow in Section 4.2.4 and learning based in Section 4.2.5.

4.2.1. Visual depth cues

An observer is able to use many cues to estimate its position relative to objects in a scene. See Fig. 4.4 for a categorization of depth cues. In natural scenes, a distinction between two types of depth cues is be made: *observer* and *object-centered* cues [106]. As the name suggest, *observer-centered* cues are related to the visual system of the observer. Accommodation, convergence, myosis and binocular disparity of the observer their

eyes give depth cues to the brain. These concepts all rely on the use of stereo-vision. As the visual system on the MAV is strictly limited by weight, computational power and energy usage, only a monocular camera is chosen to be used. Therefore, no observer-centered cues are used. *Object-centered* depth cues can be perceived with a monocular camera. These cues are categorized in static and motion-based cues. Static cues in an image include linear perspective, interposition, the height in the image plane, light and shadow, relative size, textural gradients and aerial perspective. Motion-based cues are the motion parallax and dynamic occlusion.

The motion parallax is an interesting cue to highlight as it depends on the motion of the observer in a static scene. When the observer moves (in a transitional motion), objects closer to the observer move at a higher velocity than objects further away. This effect is also used by some insects to perceive depth (see 'Peering behaviour' in Section 3.2.1). The optic flow generated by these objects can be used to detect them and estimate their relative position and depth. Optic flow is a key concept to many obstacle avoidance systems, and therefore is reviewed first. The fundamental principles of this apparent motion are discussed, and a review of event-based optic flow estimation methods is performed. After discussing optic flow, obstacle avoidance methods using a monocular camera are discussed.
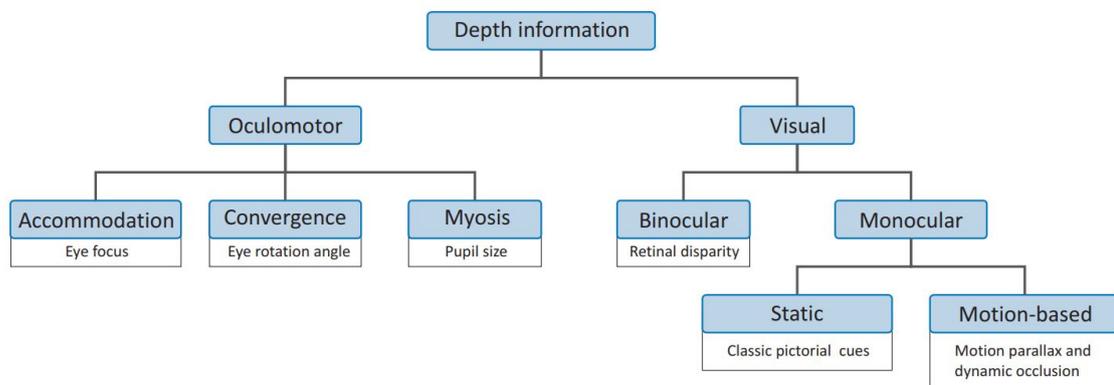

Figure 4.4: Categorization of observer and object-centered depth cues. This research focuses on monocular motion-based depth information. Image adapted from Reichelt et al. [106]

4.2.2. General principles of optic flow

Optic flow can be described as the distribution of apparent velocities of movement in an image. Gibson [**?**] introduced the concept of ecological optics, where optic flow is used to discern actions that can be taken in an environment (affordance perception). Optic flow is used by humans for a variety of tasks such as self-motion perception [20] and others (see Section 3.2.1).

When an observer moves towards an object, all visual stimuli on the retina move radially away from this object. This visual source point is called the focus of expansion (FOE), where the optic flow is zero. The image will move faster, as it reaches an angle of 90 degrees in the field of view, after which the image gradually slows down in to the focus of convergence (FOC). The FOE is discussed separately in Section 4.2.3.

In order to obtain an optic flow estimate for an arbitrary point from the 3D world, it is projected on a 2D surface (the camera or retina). The projected point on the surface has coordinates (see Fig. 4.5):

$$x = \frac{X}{Z} \ \text{ and } \ y = \frac{Y}{Z} \tag{4.2}$$

As we want to determine the motion of this point, the equation above is differentiated with respect to time:

$$\dot{x} = \frac{\dot{X}}{Z} - \frac{X\dot{Z}}{Z^2}$$
$$\dot{y} = \frac{\dot{Y}}{Z} - \frac{Y\dot{Z}}{Z^2} \tag{4.3}$$

Values for $\dot{X}$, $\dot{Y}$ and $\dot{Z}$ are found (for a derivation see Longuet-Higgins et al. [80]), resulting in the following optic flow equations:

$$u = -\frac{U}{Z} + x\frac{W}{Z} + Axy - Bx^2 - B + Cy = u_T + u_R$$
$$v = -\frac{V}{Z} + y\frac{W}{Z} - Cx + A + Ay^2 - Bxy = v_T + v_R$$

(4.4)

Note that it consists of a translational and rotational component. The rotational component is a result of camera rotations and does not contain information about the environment. Its effect can be compensated by using an Inertial Measurement Unit. These equations above also describe the relative velocity to the camera principle axis:

$$\frac{W}{Z} = \frac{u_T}{x - x_{\text{FoE}}} = \frac{v_T}{y - y_{\text{FoE}}}$$

(4.5)

Which is inversely related to the **'Time-To-Contact' (TTC)**:

$$\tau = \frac{Z}{W}$$

(4.6)

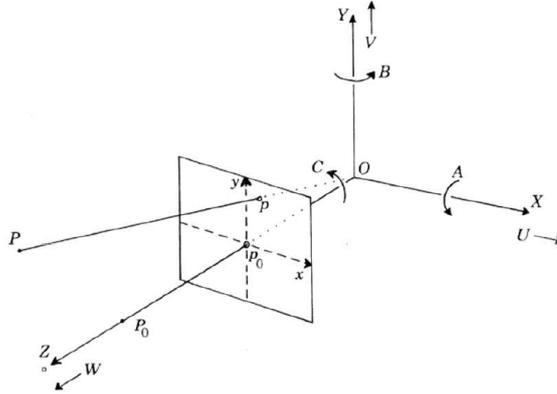Figure 4.5: Optic flow reference system. Image from Longuet-Higgins et al. [80].

Estimating event-based optic flow

As event-based cameras produce an asynchronous stream of events, a non-regular data processing pipeline has to be created compared to when using the output of a regular frame-based camera. In order to calculate optic flow from the event stream, previous research shows multiple approaches. At first, researchers applied conventional computer vision algorithms on event images. In many methods, this does not fully utilize the unique format of event-based camera output. Since event-based cameras are a relatively new field of study, recently many impact-full findings and improvements have been done. However, there is still a lack of standardized metrics and datasets (such as commonly found in the Computer Vision domain). Therefore, the actual performance of methods relative to others is often unknown. Although this new paradigm requires unconventional techniques, calculating optic flow from events can be efficient. Events represent edges, which are the sections of the scene where optic flow estimation is less ambiguous. This allows for a more efficient calculation of optic flow. Another advantage is the very high sampling rate. As this rate is much higher than when using conventional cameras, it is possible to measure high speed optic flow.

Early approaches   In contrast to the synchronous output of frame-based cameras, the event-based camera sensor does not provide this absolute brightness data. Therefore, a different relation has to be found between motion and the x, y, t space to calculate optic flow. Early research focused on event-based optic flow used conventional computer vision methods [11] [34]. As the temporal derivative of brightness can be calculated using events, it is possible to calculate the optic flow (when making assumptions for the spatial derivative
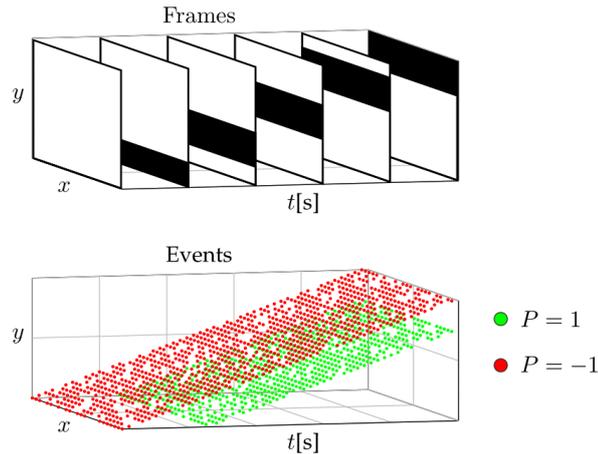
Figure 4.6: Top image: Frame based images of a horizontal bar moving upward. Bottom image: Event-based output, with two event surfaces caused by the leading and trailing edge of the horizontal bar. Image from Paredes-Valles et al. [100].

∇L). But as moving edges can generate a very sparse amount of events, the derivatives are hard to calculate and lead to unreliable results [19]. Therefore, optic flow methods using the distribution of events in the x, y, t space give better results.

Event surface approach   Approaching optic flow from a local time and space distribution of events was first proposed by Benosman et al. [12]. When considering edge slopes in the $x - t$ and $y - t$ cross sections of the event stream, the edge motion over time can be derived. Planes are fitted in this x, y, t space and the optic flow is estimated by reading the slopes of these planes (see Fig. 4.6). This results in the normal flow (which is perpendicular to the edge), but not directly in the actual direction of the edge motion. In general, the aperture problem [92] limits the accuracy of optic flow methods. The size of the spatial-temporal neighborhood, on which the planes are fitted, also influences the accuracy of the flow estimates.

A study by Rueckauer et al. [112] in 2016 compares classical Lukas-Kanade methods with local plane fitting methods, showing that plane fitting methods generally are more accurate and computationally less demanding. Though this was only demonstrated for optic flow generated with a rotating event-based camera (non-translational movement), it still shows a significant performance gap. However, they also showed that these methods require hand crafted outlier rejectors as they do not properly model the output of the event-based camera.

Several improvements on the plane fitting approach are made afterwards. In 2018, Aung et al. [5] optimized the method for usage on a Field-Programmable Gate Array (FPGA) allowing for 100M plane fits per second. Hordijk et al. [103] made efficiency improvements by reducing the number of parameters in the local plane, and capping the amount of identified optic flow vectors.

Spatio-temporal filters   The local plane structure of the event cloud is well suited for extraction of information by filtering. Brosch et al. [19] introduced spatio-temporal filters, that yield directional motion selectivity (based on Gabor and temporal filters). By hand-crafting filters to be sensitive to flow velocities and directions, optic flow can be determined. By normalizing the responses, the effect of the aperture problem is limited. Barranco et al. [10] also use a frequency based approach, in order to determine optic flow at textured edges. By using a bank of filters tuned to certain spatio-temporal frequencies, the local spatial and temporal angular frequencies is determined. Although this approach shows better results on textured edges than e.g. Benosman et al. [12] their method, it does not show real-time performance.

Artificial neural networks (ANN)   ANNs have proven to be highly valuable for conventional frame-based computer vision. An advantage of ANNs is that they avoid explicitly modeling the entire problem. The new event-based paradigm does not yet fully benefit from this progress, as the event representation is significantly different and there still is a lack of large quantities of labeled training data. When using neural networks for estimating event-based optic flow, often the event stream is collapsed into time-slices in order to be interpreted by the network. This does not fully utilize the characteristic asynchronous representation, but these

methods still achieve significant performance.

EV-FlowNet is a self-supervised learning method proposed by Zhu et al. [147] which firstly introduced the image-based representation of the event stream. This image representation can be interpreted by a standard NN architecture. The event images are fed into the NN as sole input, while the corresponding greyscale images are used as supervisory signal. This loss function is based on a proxy ground truth, originating from conventional image-based optic flow techniques. This allows the network to be trained without any manually labeled data.

Ye et al. [143] proposed an Evenly-Cascaded convolutional Network (ECN), which is able learn to estimate optic flow, depth and ego motion from sparse event data (trained on the MVSEC dataset). The ECN has only 150k parameters and also uses time slices as input. It is shown to be robust to noise (often encountered in low-light conditions), by not using the latest but averaged timestamps in the event-image. This averages out noise in the time slices and therefore increases performance.

The method used in EV-FlowNet is later extended by Zhu et al. [148] to support unsupervised learning and a novel discretized event volume representation. This representation captures the full spatio-temporal distribution of events by discretizing the time domain and accumulating events in a linearly weighted fashion. It outperforms the original EV-FlowNet on almost all MVSEC datasets and also the ECN on certain datasets.

Spiking Neural Nets (SNN) Where ANNs do not fully utilize the unique asynchronous event stream representation, SNNs are in favor as they are able to handle the event stream as direct input. The SAVME method from Orchard et al. [99] uses a spiking neural net to mimic the Lucas-Kanade algorithm. It uses individual events as spike input for the SNN and uses bio-inspired direction and velocity selective neurons to calculate the optic flow. This method is not yet applicable in real-time computation.

The method proposed by Paredes-Vallés et al. [100] also determines optic flow with a hierarchical SNN, using a novel biologically plausible Spike-Timing-Dependent Plasticity (STDP) protocol. It is also able to handle the rapidly varying input distribution of the event camera output. Just as the SAVME method, it learns the neural selectivity from the event stream and thus can also be interpreted as a spatio-temporal filter.

Correlation-based methods A fundamental challenge in estimating event-based optic flow is the lack of association between events and established features. Grouping events and searching for the most similar event group would require a time window specification, and is thus not suitable for the asynchronous event stream.

Zhu et al. [146] propose a correlation-based method for feature tracking, where the optic flow is calculated by maximization of the expectation over all data associations. By making the event associations probabilistic, no hard commitments have to be done for events to features. It outperforms a Kanade–Lucas–Tomasi (KLT) tracker applied to a 240 FPS camera, but is not yet applicable in real-time calculation.

Works by Gallego et al. [44], [45], [47] seek for the point trajectories on the image plane that best fit the event data, also known as contrast maximization or focus. The framework proposed is able to calculate optic flow, perform 3D reconstruction and motion estimation. By warping the events, a good fit is found with previous events, and therefore recovers the relative motion between camera and scene expressed in parameters. These methods maximize the fit of groups of events, using their spatio-temporal and polarity information. Stoffregen et al. [128] review different reward functions for contrast maximization. They showed that using a combination of sparsity- and magnitude-reward functions, supports dealing with the aperture problem and increases performance under noisy conditions.

Liu et al. [78] propose an adaptive block-matching optic flow (ABMOF) method, which uses a conventional computer vision algorithm 'block matching' (known from MPEG video encoders) on time slices of the event stream (see Section 4.1.4). These time slices are created by taking either a constant time or constant amount of events. Blocks from these time slices are matched against each other, which allows for optic flow derivation from their position difference. The advantage of this method is that it is highly parallelizable on hardware.

This section gave a general overview of optic flow and event-based optic flow estimation methods. In this context, the focus of expansion is an important flow characteristic. Therefore its theory and estimation methods are discussed in the following section.

4.2.3. Focus of Expansion

The focus of expansion is a singular point from which the apparent optic flow expands, assuming the scene is static and the motion of the observer is purely translational. This singular point indicates the course of the observer, and therefore is a crucial element in visual-based navigation. Determining the FOE is challenging as often normal flow is available (due to the aperture problem), and the computational limitation of the MAV does not allow for computationally expensive online visual-processing. If the optic flow is zero and the rotational component is filtered out, the following derivation is made.

$$u_T = 0 = -\frac{U}{Z} + x_{FOE}\frac{W}{Z}$$
$$v_T = 0 = -\frac{V}{Z} + y_{FOE}\frac{W}{Z} \tag{4.7}$$

Rewriting these equations gives the following result.

$$x_{FOE} = \frac{U}{W}$$
$$y_{FOE} = \frac{V}{W} \tag{4.8}$$

To show optic flow diverges from the FOE, (4.7) and (4.8) are used to re-express u_T and v_T.

$$u_T = -\frac{U}{Z} + \frac{xW}{Z} = (-\frac{U}{W} + x)\frac{W}{Z} = (x - x_{FOE})\frac{W}{Z}$$
$$v_T = -\frac{V}{Z} + \frac{yW}{Z} = (-\frac{V}{W} + y)\frac{W}{Z} = (y - y_{FOE})\frac{W}{Z} \tag{4.9}$$

Rewriting this equation shows the geometrical relation which results in the optic flow diverging from the FOE.

$$\frac{u_T}{v_T} = \frac{x - x_{FOE}}{y - y_{FOE}} \tag{4.10}$$

Estimating the Focus of Expansion

The determination of the FOE on-board mobile systems equipped with cameras has received a large attention from researchers over the past decades, showing a great variety of approaches to actually solve this very complex problem. This literature study is focused on sparse OF-based FOE estimation, for which state-of-the-art solutions currently available can be divided into three categories: (i) counting vectors directions, (ii) creating a probability map based on negative vector intersections and (iii) based on negative half-planes.

i) Counting vectors directions   An early approach by Souhila et al. [123] estimates the FOE by counting the amount of vectors that diverge from a location, for all horizontal and vertical locations. The location in which most vectors diverge is the FOE. Huang et al. [57] take a similar approach, in which all vectors are projected onto lower-dimensions and all horizontal and vertical locations are evaluated.

ii) Probability map based on negative vector intersections   Guzel et al. [121] proposed an implementation of a method which calculates the intersections of all rays in opposite direction of the optic flow vectors. A probability map is generated which holds the amount of intersections per location. The location which has the most intersections has the highest probability of being the FOE. Stabinger et al. [127] used this method in an obstacle detection application. A similar method was implemented by Buczko et al. [21] which uses a RANSAC scheme to randomly select two vectors, create a candidate FOE location by calculating the intersection, and testing this location against all vectors. After a predetermined amount of iterations, the candidate with the highest amount of inliers is selected as the FOE. This and research by Suhr et al. [130] use this RANSAC method to determine outliers in a visual-odometry pipeline.

iii) Probability maps based on negative half-planes intersections   The third type of approach, creating a probability map using negative half-planes was implemented by Clady et al. [26]. As normal optic flow is calculated, the assumption is made that the FOE must lie in the negative half-plane of as many normal vectors as possible. For each optic flow vector an orthogonal line is taken, which intersects the vector location. The negative half-plane of this orthogonal line is used to update the probability map. All locations which are not updated are subject to exponential decay over time. The location with the highest value on the probability map is selected as the FOE. This research by Clady et al. and research by Colonnier et al. [31] use this method to estimate the Time-To-Contact in an obstacle avoidance task.

These sections gave a general overview of optic flow, the focus of expansion and event-based estimation methods. These key concepts are often used in visual obstacle avoidance methods. The following section will review obstacle avoidance methods, primarily based on optic flow.

4.2.4. Obstacle avoidance methods using optic flow

Optic flow is a key concept in order to gain environmental awareness. As it describes the apparent motion of an image of the scene, it holds important information about objects and motion of the observer. Therefore, previous research already proposed a variety of methods for the use of optic flow in obstacle avoidance. In this research, optic flow is estimated from the output of an event-based camera. As this event-based camera is a relatively new field of research, limited previous work has been proposed. Therefore, this section will review both frame and event-based methods to obstacle avoidance. These methods are reviewed on the following criteria: obstacle detection method, obstacle avoidance strategy, novelty of method, weaknesses of method, computational needs and biological plausibility. These criteria are used to highlight the advantages and disadvantages of implementing the method for this research.

Optic flow balancing

A simple strategy, based on the centering behavior of insects (see Section 3.2.1), is optic flow balancing. When flying through a corridor, it is possible to stay laterally centered by balancing the amount of optic flow in the left and right side of the image. The literature review showed that only **frame-based** monocular cameras have been used for this approach, no optic flow balancing approaches using an event-based camera are proposed yet. Early research in 2004 showed an implementation by Argyros et al. [4] on a mobile robot. This implementation used a very basic control law, but proves the concept of mimicking the centering behaviour. The main disadvantage of optic flow balancing is that it is not able to avoid obstacles directly in or near the focus of expansion, as the optic flow is zero at this point.

In 2010, Zingg et al. [149] implemented this optic flow balancing strategy on an MAV for indoor corridor following. An omnidirectional camera is used to calculate the optic flow along the two walls. From this optic flow, the distance to the walls is estimated. A PD controller on the error from the corridor center is used to keep the MAV centered. It was also concluded that very accurate IMU data is required to calculate reliable flow information.

Research by Agrawal et al. [1] in 2017 showed that using a control strategy that uses the inverse of the optic flow difference between the two Field of View (FoV) halves, it is able to avoid frontal collision with walls. The disadvantage of this strategy is that it generates a lot of control jittering, as the optic flow difference is zero when flying straight through a corridor.

In 2019, Cho et al. [25] implemented the standard optic flow balancing method, but also extended it towards the vertical axis by splitting up the FoV in four planes. Figure 4.7 shows the left horizontal optic flow region, with increased activity as a wall is nearby. A frontal obstacle avoidance strategy is also implemented by calculating the expansion of optic flow in both horizontal planes and subsequently determining the heading rate. Their experiments are performed with low velocities and therefore do not include any pitch or yaw compensation. Also, the performance highly depends on the quality of dense optic flow estimation, which is low for non-textured objects.
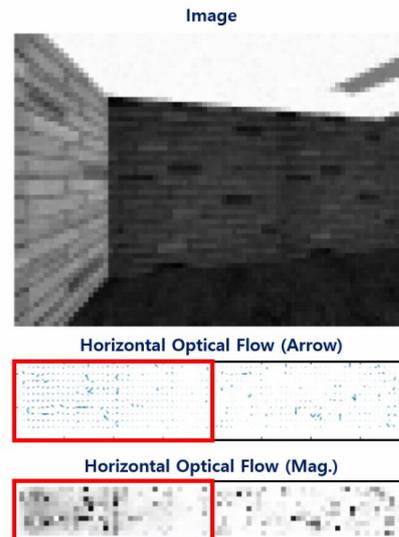
Figure 4.7: Example of an optic flow balancing method, applied in simulation. The left horizontal optic flow region shows increased magnitude and is subsequently used in calculating the control input. Image adapted from Cho et al. [25]

Concluding in general, optic flow balancing is a simple method, inspired by the centering response of insects. It enables straight flight in well-textured corridors but is not able to handle obstacles that lie in the focus of expansion. It has a low computational demand as the strategy can be derived directly from the estimated optic flow in regions of interest in the FoV.

Time-To-Contact (TTC)

As shown in Section 4.2.2, the calculation of TTC is straightforward. If the Focus of Expansion (FoE) and the optic flow vector is known, the TTC is a direct derivative of the optic flow. TTC is a biological plausible concept (see Section 3.2.1), and requires low computational effort as it is a direct derivative of the already calculated optic flow. It is also able to handle multiple objects, as for each tracked image patch, the TTC can be calculated. A disadvantage is that it cannot handle objects directly in the FoE (as the optic flow is zero in the FoE). It is also highly dependent on the estimation accuracy of the FoE and flow vectors.

Considering **frame-based** monocular cameras, early works by Kai-Tai et al. [60] in 2001 showed the possibility of estimating TTC in real time. They estimated dense optic flow and mapped the TTC for all pixels in an angular histogram. Information from the histogram is then used to choose a suitable avoidance direction. In 2009, research from Byrne et al. [23] performed expansion segmentation in simulation on images by minimizing an expectation-maximization framework. They used a novel TTC uncertainty model and were able to segmentate collision regions in simulation. It was not tested on-board a mobile robot, and the segmentation optimization required high computational power.

TTC estimation using **event-based** monocular cameras was proposed first by Clady et al. [27] in 2014. They implemented event-based TTC on a mobile ground robot. To estimate the optic flow, an event surface method was used. Subsequently, a probability map was used to estimate the FoE. As they do not compensate for rotational motion of the mobile robot, it was only able to perform under stable translational motion. The implementation on the mobile robot did not contain any control algorithm, it was only proposed to show the TTC estimation. Milde et al. [85] also implemented TTC on a mobile robot using event surfaces to estimate optic flow in 2015. They also included a (open-loop) control scheme, based on TTC estimates. It was not validated by controlling the mobile robot under closed-loop conditions or using real-time data processing.

In 2018, Colonnier et al. [30] improved the work of Clady et al. and proposed an implementation on a quadrotor. An FPGA was used to filter events, and subsequently send them to a ground-station for further processing. Under assumption of a constant velocity, they implemented a Kalman filter on the TTC estimation. Dividing the Field of View (FoV) into three regions of interest, they were able to create a controller which allowed for a simple obstacle avoidance strategy. If a threshold TTC in one of the regions of interest was reached, the controller steered the quadrotor in the opposite direction until the TTC has decreased again.

Concluding in general, using TTC for obstacle detection is a simple, biologically feasible strategy, able to handle multiple objects with low computational demand. Disadvantages are the heavy reliance on the optic flow and FoE estimation accuracy and the inability to handle objects in (or close to) the FoE.

Object size expansion & epipolar geometry methods

As described in Section 4.2.1, the change of object size is a depth cue which can be used for obstacle detection. First, the object has to be detected and subsequently, the rate of expansion has to be determined. There are several **frame-based** approaches which track features, identify whether they belong to an object, and subsequently estimate the rate of expansion of the identified group. No event-based methods are proposed yet on this topic.

In 2013, Mori et al. [90] used a SURF feature tracker and estimated the relative scaling of tracked features over several frames in a certain region of interest. Its performance is highly dependent the accuracy of the SURF feature tracker and on the amount of texture on the object. Al-Kaff et al. [3] used the same feature size expansion method in 2016. They also added a convex hull around the tracked features and included the size expansion of that hull in their estimation. This approach has the same disadvantage as the work of Mori, described above. In 2017, Falanga et al. [39] proposed a method that was able to fly through a narrow inclined gap using on-board sensing and advanced trajectory planning. It uses a priori information about the geometry of the gap, and uses a feature tracker to estimate the relative position of the gap. Although this method shows the agility of the MAV and fast on-board processing, it is not generalizable due to the specific task it was proposed for.

Works by Schaub et al. [116] [117] proposed a method that is able to handle dynamic objects using clusters of tracked features, similar to the size expansion methods described above. As psychologist J. Gibson described [49], if an object expands symmetrically on the scene image, the obstacle is on a collision path. If the epipolar center of optic flow vectors, related to the dynamic object, lie within the object boundary, the object is on a collision path. If the epipole of the flow vectors is outside of the dynamic object, it passes the observer (see Fig. 4.8). This approach was implemented on an autonomous vehicle, but has a high computational demand (due to the clustering optimization) and is strongly dependent on the feature tracker accuracy.
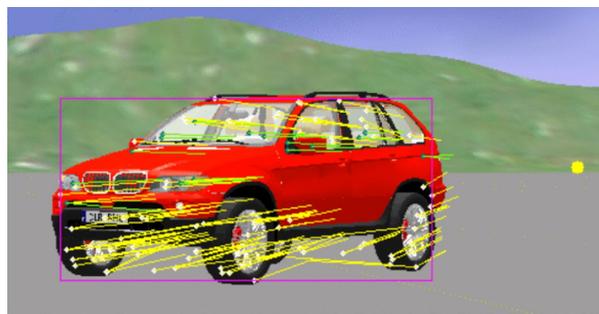

Figure 4.8: Example of epipolar geometry method, applied in simulation. The yellow dot is the epipolar center of the optic flow vectors in the obstacle cluster. The epipolar center clearly lies outside the object boundary and thus the object is not on a collision course. Image from Schaub et al. [117]

Concluding in general, size expansion methods using tracked features are able to handle frontal objects but cannot handle corridors and are heavily dependent on the feature tracker accuracy. Using epipolar geometry, it is possible to discern if objects are on a collision course or passing the observer. This is also highly dependent on the clustering accuracy of optic flow vectors on objects.

Scene mapping methods

Creating a map of the environment enables map-based motion planning. There are many optimized planning methods which are outside the scope of this literature review. A well-known mapping method is 'Visual Simultaneous Localization And Mapping' (VSLAM). It localizes the MAV locally, estimates its state and builds a three-dimensional map of the environment using visual information. This is often achieved by using monocular, stereo or RGB-D cameras [91]. Although these methods allow for an advanced model of the environment, the computational complexity limits its usage on fully autonomous MAVs. Often, the visual data is processed by an off-board ground station and only the control command is send to the MAV using a

wireless connection (such as by Zhang et al. [145] or Esrafilian et al. [38]), which strongly limits its autonomy. Heng et al. [54] showed that by tightly synchronizing the IMU and four on-board stereo vision cameras, while heavily down-sampling the 3D virtual scan data, SLAM and obstacle avoidance can be performed using fully on-board processing but is still a computationally highly demanding task.

This concludes the review of optic flow based obstacle avoidance methods. As obstacle avoidance is a complex task which is hard to model, many researchers use a learning based approach. The following section reviews previous research on learning based obstacle avoidance methods.

4.2.5. Obstacle avoidance methods using learning

Visual obstacle avoidance is a complex task with many possible cues. In the computer vision domain, often a learning based approach is used to avoid modelling the entire problem. As already described in Section 4.2.2, optic flow estimation is often performed using neural networks and a learning based approach. This section will cover methods that use the camera video stream as input, and output a control command. This end-to-end approach avoids modelling the complex mechanisms behind obstacle avoidance, but also has its limitations.

Imitation and supervised CNN learning

In 2012, Ross et al. [110] proposed a monocular **frame-based** approach based on imitation learning. Frame based video and control input from a piloted drone, flying through a dense forest, was used as training set. Based on four types of features, the video stream is analyzed and given as input to the trained network. The main disadvantage of this approach is the large amount of manually generated training data which is required.

Manglik et al. [82] trained a convolutional neural network (CNN) to estimate TTC from a frame based camera. They used a LIDAR scanner to generate ground truth values and video from a monocular camera to train the network. Although this is a novel approach, the network is hard to generalize to unknown scenes and a LIDAR scanner is required to generate ground truth values.

The first implementation of a CNN for monocular **event-based** obstacle avoidance, was proposed by Moeys et al. [88] in 2018, in combination with their PRED18 dataset. In a 'prey following' task, they divided the FoV into three vertical regions and used a CNN to recognize the relative heading of a target 'prey'. They trained the CNN by using laser range meters to estimate distance to obstacles. Their approach was only able to output three control commands: steer left, right or center. In 2019, Sanket et al. [115] proposed the EVDodgeNet, an MAV implementation which is able to avoid small objects thrown at the MAV. It uses a CNN on event-images to perform obstacle segmentation and to estimate optic flow. Based on the size of the object (known a priori), it is able to estimate a three dimensional avoidance path. The disadvantage of this approach is its requirement for a priori information, and its heavy computational requirements.

LGMD neural structure

The Lobula Giant Movement Detector neurons (see 'Lobula Giant Movement Detector' in Section 3.2.1) in the brain of locust, are able to detect looming predators. Several implementations have took the approach of mimicking this neural structure for obstacle avoidance. A **frame-based** camera approach was done in 2016 by Hu et al. [56]. They implemented the system on a micro-robot, which was able to avoid collision with textured poles. Its neural structure allowed for very low computational demand but also allowed for only three output signals: stop, drive forward, or move right or left (chosen randomly). This strongly restricts the generalizability of their approach.

When using an **event-based** camera for this method, the asynchronous spiking ouput of the event-based camera fits well with the spiking structure of spiking neural nets (SNN). In 2017, works by Salt et al. [113] [114] implemented the event-based camera and an LGMD neuron on a quadrotor. It was able to avoid looming stimuli by moving towards a region in the field of view with less spiking LGMD neurons. Milde et al. [86] implemented an LGMD network on a mobile robot with a ROLLS neuromorphic chip. Their approach allowed for two motor outputs of the network: turn left (when there is an obstacle in the right field of view), and vice versa. Although in general, the LGMD method is one of the most biologically feasible approaches, the implementations still lack generalizability as it is still a very undeveloped field of research.

In general, learning approaches are advantageous as they do not require to model the full problem and neural nets are biologically inspired. These methods also have certain disadvantages, such as being hard generalize to unknown scenes. Also, low-light applications are often an issue for neural net approaches, as little training data is available for these conditions. More broadly, this is an issue for event-based camera applications with a learning approach, in contrast to conventional frame-based approaches.

4.3. Synthesis

This chapter focuses on previous research on obstacle avoidance using event-based cameras. To highlight the most relevant findings in this field of research, this section will answer the research questions on event-based cameras and obstacle avoidance.

What does the theory state about event-based vision? (Section 4.1)

The event-based camera is chosen as a basis for this research as it has characteristics that suit the research objective well. Its pixels detect brightness change, which can be interpreted as apparent motion in the scene. It is evident that this is highly relevant for obstacle avoidance. The event-based camera also has a very low power consumption (< 100 mW), which is advantageous for MAV applications. It is also based on a biological inspired principle (directionally sensitive cells), which follows the bio-inspired approach of this research. As the event-based camera outputs an asynchronous stream of events, the processing pipeline also needs to be changed significantly. Using event packages (e.g. to estimate optic flow) preserves the correlation between individual events, and has been shown to be an effective method of processing events.

Section 4.2.2 discussed different methods of estimating event-based optic flow. In general, there is no clear best performing method, as there is still a lack of benchmark metrics and datasets in this field of research.

In order to use conventional neural network approaches to estimate optic flow, the event stream has to be converted back to frames. This disregards valuable spatio-temporal information. ANN approaches require a lot of training data and do not generalize well to low-light environments. Spiking neural net optic flow research is still in its infancy, and has not been shown to perform as well as other approaches. Correlation (or 'focus') methods have a high potential as they fully utilize the spatio-temporal structure of the event cloud. However, these methods require an optimization scheme to find a correlation between events. This has a high computational demand and unknown performance for estimating optic flow. In contrast, using an event surface approach utilizes the time and space distribution of events. It has also been shown to estimate optic flow with higher accuracy than the conventional Lukas-Kanade algorithm. Therefore, this is a promising approach to estimating event-based optic flow.

What are state-of-the-art monocular event-based obstacle avoidance methods in the mobile robotics domain? (Section 4.2)

This section discussed different optic flow, scene mapping and learning based obstacle avoidance methods, using a frame- or event-based camera. *Optic flow based* methods utilize the apparent motion of the scene image. Optic flow balancing, Time-To-Contact (TTC) and object size expansion methods have been shown to have a relatively low computational complexity and are based on biological principles discussed in Section 3.2.1. *Scene mapping* methods (such as VSLAM) allow for map-generation, but are also highly computationally demanding and not biologically feasible. As fully on-board processing is required for this research, and a bio-inspired approach is taken, scene mapping methods are not considered. *Learning based* methods, such as imitation learning or supervised CNNs, are advantageous as they do not require to model the full problem and neural nets are biologically inspired. However, these methods are also hard to generalize to low-light environments, as little training data is available for event-based camera applications. Learning approaches also often require heavy computations, which cannot be performed on-board the MAV. Based on these considerations, optic flow based methods are most suitable to fit the research objective: an obstacle avoidance system usable in low-light and with on-board processing.

To review optic flow based obstacle avoidance methods from a functional perspective, different scenarios are proposed in this section. Depending on the scene in which the MAV has to manoeuvre without hitting obstacles, the discussed methods have certain advantages and disadvantages. With these scenarios, the most suitable approaches to obstacle avoidance are given, alongside with a recent state-of-the-art paper. First, scenarios with static objects are shown in Table 4.1. As can be seen, the Time-To-Contact and object size

expansion methods can be utilized for both the frontal wall and pole. However, if the frontal surface of the pole is too small, the Time-To-Contact method will not be able to detect the pole (as the TTC in the FoE is zero). Therefore, for the pole scenario, the object size expansion methods are preferred. The optic flow balancing approach can be applied to both the corridor and one-sided wall scenarios. The state-of-the-art research for each method is also given in Table 4.1. No event-based approaches are known for the object size expansion and optic flow balancing methods and thus frame-based approaches are listed.
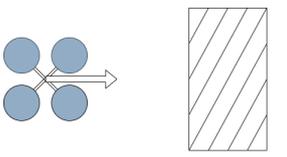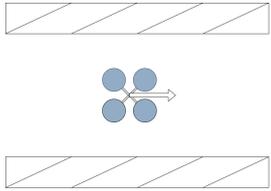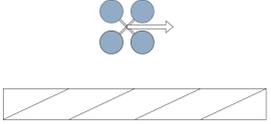
Frontal Wall	Frontal Pole	Corridor	One-sided Wall
Method: Time-to-contact	**Method:** Object Size Expansion	**Method:** Optical Flow Balancing	**Method:** Optical Flow Balancing
State-of-the-art: Colonnier et al. (2018) [30] (Event-based camera)	**State-of-the-art:** Mori et al. (2013) [90] (Frame-based camera)	**State-of-the-art:** Cho et al. (2019) [25] (Frame-based camera)	**State-of-the-art:** See 'Corridor' scenario

Table 4.1: Scenarios for static obstacle avoidance, with most suitable methods and state-of-the-art research papers.

Table 4.2 shows the scenarios with dynamic objects, such as an approaching pole. The frontal approaching pole is similar to the static pole, although if the MAV would stop its motion, the pole would still advance towards the MAV. The same methods as for the static pole can be applied here. The Translating Perpendicular Pole and Approaching Pole Under Angle both require a more complex method. It is unknown if the obstacle will pass in front or behind the MAV, or is on a collision course. The literature review has shown that using epipolar geometry of the optic flow vectors provides a solution to these scenarios. These methods are able to detect if the pole is on a collision course or will not hit the observer.

Frontal Approaching Pole	Translating Perpendicular Pole	Approaching Pole Under Angle
Method: Object Size Expansion	**Method:** Obstacle segmentation + Epipolar lines	**Method:** Obstacle segmentation + Epipolar lines
State-of-the-art: See 'Frontal Pole' scenario	**State-of-the-art:** Schaub et al. (2016) [117] (Frame-based camera)	**State-of-the-art:** See 'Translating Perpendicular Pole' scenario

Table 4.2: Scenarios for dynamic obstacle avoidance, with most suitable methods and state-of-the-art research papers.

This concludes the chapter on event-based cameras, optic flow, FOE estimation and obstacle avoidance methods. As the obstacle avoidance methods apply to specific scenarios, a combination of these methods is required to make the system robust to any scenario. This implementation is performed in the experimental research. For this literature research, the central research questions are listed and answered in this synthesis, such that this knowledge is applicable in the obstacle avoidance system design.

5

Conclusion

This literature review provides a fundamental basis for the research into a bio-inspired approach to obstacle avoidance in low-light using event-based cameras on an MAV. The contribution of this literature review is in threefold: biological inspiration is drawn from fundamental principles of insect obstacle avoidance and nocturnal vision, and state-of-the-art optic flow, focus of expansion and obstacle avoidance methods are reviewed. To support the bio-inspired approach, the biological principles behind human and insect vision are discussed and compared. This shows that insects are able to process visual information using very efficient methods and which provide valuable insights for MAV applications. The first topic of biological inspiration is insect obstacle avoidance. Several types of visually guided behaviour are discussed, of which time-to-contact estimation, the centering response in corridors, peering behaviour and saccadic motion are the most relevant concepts. These concepts are used as inspiration in the obstacle avoidance system design. The second topic of biological inspiration is the set of fundamental principles enabling nocturnal vision. Spatial and temporal summation and a slow response time with increased gain are neural processing characteristics, which can be used as inspiration to improve the performance of the event-based camera system in low-light. In these low-light conditions, two relevant types of visual noise are identified: photon shot noise and thermal noise. To suppress these types of noise, spatially filtering the output and cooling the event-based camera is proposed. After discussing these two topics of bio-inspiration, state-of-the-art research in event-based optic flow, focus of expansion and obstacle avoidance is reviewed. As the event-based optic flow domain still lacks standardized metrics and datasets, no clear comparison can be made between methods. The event-surface optic flow approach has been shown to utilize the space-time distribution of events well, has high accuracy compared to conventional techniques and therefore is a promising approach. To categorize state-of-the-art obstacle avoidance methods, several static and dynamic obstacle scenarios are proposed. According to these scenarios, the most suitable obstacle avoidance methods and state-of-the-art research is listed. With this literature review, all literature research questions are answered and a fundamental theoretical basis is provided for reaching the research objective.

Bibliography

[1] Pooja Agrawal, Ashwini Ratnoo, and Debasish Ghose. Inverse optical flow based guidance for uav navigation through urban canyons. *Aerospace Science and Technology*, 68:163–178, 2017. ISSN 1270-9638. doi: https://doi.org/10.1016/j.ast.2017.05.012.

[2] A-C Aho, Kristian Donner, Christel Hydén, Lis Olesen Larsen, and Tom Reuter. Low retinal noise in animals with low body temperature allows high visual sensitivity. *Nature*, 334(6180):348–350, 1988. ISSN 1476-4687.

[3] Abdulla Al-Kaff, Qinggang Meng, David Martín, Arturo de la Escalera, and José María Armingol. Monocular vision-based obstacle detection/avoidance for unmanned aerial vehicles. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 92–97. IEEE, 2016. ISBN 1509018212.

[4] Antonis A Argyros, Dimitris P Tsakiris, and Cedric Groyer. Biomimetic centering behavior [mobile robots with panoramic sensors]. *IEEE Robotics Automation Magazine*, 11(4):21–30, 2004. ISSN 1070-9932.

[5] Myo Tun Aung, Rodney Teo, and Garrick Orchard. Event-based plane-fitting optical flow for dynamic vision sensors in fpga. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE, 2018. ISBN 1538648814.

[6] Emily Baird, Torill Kornfeldt, and Marie Dacke. Minimum viewing angle for visually guided ground speed control in bumblebees. *Journal of Experimental Biology*, 213(10):1625–1632, 2010. ISSN 0022-0949.

[7] Emily Baird, Norbert Boeddeker, Michael R Ibbotson, and Mandyam V Srinivasan. A universal strategy for visually guided landing. *Proceedings of the National Academy of Sciences*, 110(46):18686–18691, 2013. ISSN 0027-8424.

[8] H. B. Barlow. Purkinje shift and retinal noise. *Nature*, 179(4553):255–256, 1957. ISSN 1476-4687. doi: 10.1038/179255b0. URL https://doi.org/10.1038/179255b0.

[9] Horace B Barlow. Retinal noise and absolute threshold. *JOSA*, 46(8):634–639, 1956.

[10] Francisco Barranco, Cornelia Fermuller, and Yiannis Aloimonos. Bio-inspired motion estimation with event-driven sensors. In *International Work-Conference on Artificial Neural Networks*, pages 309–321. Springer, 2015.

[11] Ryad Benosman, Sio-Hoi Ieng, Charles Clercq, Chiara Bartolozzi, and Mandyam Srinivasan. Asynchronous frameless event-based optical flow. *Neural Networks*, 27:32–37, 2012. ISSN 0893-6080.

[12] Ryad Benosman, Charles Clercq, Xavier Lagorce, Sio-Hoi Ieng, and Chiara Bartolozzi. Event-based visual flow. *IEEE transactions on neural networks and learning systems*, 25(2):407–417, 2013. ISSN 2162-237X.

[13] AD Blest and Michael Francis Land. The physiological optics of dinopis subrufus l. koch: a fish-lens in a spider. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 196(1123):197–222, 1977. ISSN 0080-4649.

[14] Reinoud J Bootsma and Piet CW van Wieringen. Timing an attacking forehand drive in table tennis. *Journal of experimental psychology: Human perception and performance*, 16(1):21, 1990. ISSN 1939-1277.

[15] Alexander Borst. Drosophila's view on insect vision. *Current biology*, 19(1):R36–R47, 2009. ISSN 0960-9822.

[16] Alexander Borst and Susanne Bahde. Visual information processing in the fly's landing system. *Journal of Comparative Physiology A*, 163(2):167–173, 1988. ISSN 0340-7594.

[17] Alexander Borst and Martin Egelhaaf. Principles of visual motion detection. *Trends in neurosciences*, 12(8):297–306, 1989. ISSN 0166-2236.

[18] Alexander Borst and Thomas Euler. Seeing things in motion: models, circuits, and mechanisms. *Neuron*, 71(6):974–994, 2011. ISSN 0896-6273.

[19] Tobias Brosch, Stephan Tschechne, and Heiko Neumann. On event-based optical flow detection. *Frontiers in neuroscience*, 9:137, 2015. ISSN 1662-453X.

[20] Anna R Bruss and Berthold KP Horn. Passive navigation. *Computer Vision, Graphics, and Image Processing*, 21(1):3–20, 1983. ISSN 0734-189X.

[21] M. Buczko and V. Willert. Monocular outlier detection for visual odometry. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 739–745, 2017. doi: 10.1109/IVS.2017.7995805.

[22] M Burrows and CH Fraser Rowell. Connections between descending visual interneurons and metathoracic motoneurons in the locust. *Journal of comparative physiology*, 85(3):221–234, 1973. ISSN 0340-7594.

[23] J. Byrne and C. J. Taylor. Expansion segmentation for visual collision detection and estimation. In *2009 IEEE International Conference on Robotics and Automation*, pages 875–882, 2009. ISBN 1050-4729. doi: 10.1109/ROBOT.2009.5152487.

[24] K. Cheng and Y. Wang. Using mobile gpu for general-purpose computing – a case study of face recognition on smartphones. In *Proceedings of 2011 International Symposium on VLSI Design, Automation and Test*, pages 1–4, 2011. doi: 10.1109/VDAT.2011.5783575.

[25] Gangik Cho, Jongyun Kim, and Hyondong Oh. Vision-based obstacle avoidance strategies for mavs using optical flows in 3-d textured environments. *Sensors*, 19(11):2523, 2019.

[26] X. Clady, Charles Clercq, Sio-Hoi Ieng, Fouzhan Houseini, M. Randazzo, L. Natale, C. Bartolozzi, and R. Benosman. Asynchronous visual event-based time-to-contact. *Frontiers in Neuroscience*, 8, 2014.

[27] Xavier Clady, Charles Clercq, Sio-Hoi Ieng, Fouzhan Houseini, Marco Randazzo, Lorenzo Natale, Chiara Bartolozzi, and Ryad Benjamin Benosman. Asynchronous visual event-based time-to-contact. *Frontiers in neuroscience*, 8:9, 2014. ISSN 1662-453X.

[28] Mark N Coleman and Doug M Boyer. Inner ear evolution in primates through the cenozoic: implications for the evolution of hearing. *The Anatomical Record: Advances in Integrative Anatomy and Evolutionary Biology*, 295(4):615–631, 2012. ISSN 1932-8486.

[29] SP Collin. *Behavioural ecology and retinal cell topography*, pages 509–535. Springer, 1999.

[30] Fabien Colonnier, Luca Della Vedova, Rodney Swee Huat Teo, and Garrick Orchard. Obstacle avoidance using event-based visual sensor and time-to-contact processing. In *2018 Australasian Conference on Robotics and Automation*, 2018.

[31] Fabien Colonnier, Luca Della Vedova, R. Teo, and G. Orchard. Obstacle avoidance using event-based visual sensor and time-to-contact processing. 2018.

[32] Christine A Curcio, Kenneth R Sloan, Robert E Kalina, and Anita E Hendrickson. Human photoreceptor topography. *Journal of comparative neurology*, 292(4):497–523, 1990. ISSN 0021-9967.

[33] Charles T David. Compensation for height in the control of groundspeed bydrosophila in a new,'barber's pole'wind tunnel. *Journal of comparative physiology*, 147(4):485–493, 1982. ISSN 0340-7594.

[34] Tobi Delbruck. Frame-free dynamic digital vision. In *Proceedings of Intl. Symp. on Secure-Life Electronics, Advanced Electronics for Quality Life and Society*, pages 21–26. Citeseer, 2008.

[35] Kristian Donner. Visual latency and brightness: an interpretation based on the responses of rods and ganglion cells in the frog retina. *Visual neuroscience*, 3(1):39–51, 1989. ISSN 1469-8714.

[36] Hubert Eichner, Maximilian Joesch, Bettina Schnell, Dierk F Reiff, and Alexander Borst. Internal structure of the fly elementary motion detector. *Neuron*, 70(6):1155–1164, 2011. ISSN 0896-6273.

[37] Ecole polytechnique fédérale de Lausanne (EPFL). Curved artificial compound eyes, 2020. URL https://www.epfl.ch/labs/lis/research/completed/curvace/. Accessed: July 2020.

[38] Omid Esrafilian and Hamid D Taghirad. Autonomous flight and obstacle avoidance of a quadrotor by monocular slam. In *2016 4th International Conference on Robotics and Mechatronics (ICROM)*, pages 240–245. IEEE, 2016. ISBN 1509032223.

[39] Davide Falanga, Elias Mueggler, Matthias Faessler, and Davide Scaramuzza. Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 5774–5781. IEEE, 2017. ISBN 150904633X.

[40] Aaron J Fath, Mats Lind, and Geoffrey P Bingham. Perception of time to contact of slow-and fast-moving objects using monocular and binocular motion information. *Attention, Perception, Psychophysics*, 80(6):1584–1590, 2018. ISSN 1943-3921.

[41] Rikard Frederiksen and Eric J Warrant. The optical sensitivity of compound eyes: theory and experiment compared. *Biology letters*, 4(6):745–747, 2008. ISSN 1744-9561.

[42] Michael A Freed. Rate of quantal excitation to a retinal ganglion cell evoked by sensory input. *Journal of Neurophysiology*, 83(5):2956–2966, 2000. ISSN 1522-1598.

[43] Fabrizio Gabbiani, Holger G Krapp, Christof Koch, and Gilles Laurent. Multiplicative computation in a visual neuron sensitive to looming. *Nature*, 420(6913):320–324, 2002. ISSN 1476-4687.

[44] Guillermo Gallego and Davide Scaramuzza. Accurate angular velocity estimation with an event camera. *IEEE Robotics and Automation Letters*, 2(2):632–639, 2017. ISSN 2377-3766.

[45] Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3867–3876, 2018.

[46] Guillermo Gallego, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew Davison, Joerg Conradt, and Kostas Daniilidis. Event-based vision: A survey. *arXiv preprint arXiv:1904.08405*, 2019.

[47] Guillermo Gallego, Mathias Gehrig, and Davide Scaramuzza. Focus is all you need: loss functions for event-based vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12280–12289, 2019.

[48] Laszlo Zsolt Garamszegi, Anders Pape Møller, and Johannes Erritzøe. Coevolving avian eye size and brain size in relation to prey capture and nocturnality. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 269(1494):961–967, 2002. ISSN 0962-8452.

[49] James J. Gibson. *The ecological approach to visual perception*. Resources for ecological psychology. Lawrence Erlbaum Associates, Publishers, Hillsdale, New Jersey, 1986.

[50] T Goodman, T Bergen, P Blattner, Y Ohno, J Schanda, and T Uchida. The use of terms and units in photometry–implementation of the cie system for mesopic photometry (cie tn 004: 2016). *Commission Internationale de l'Eclairage (CIE)*, 2016.

[51] Birgit Greiner, Willi A Ribi, William T Wcislo, and Eric J Warrant. Neural organisation in the first optic ganglion of the nocturnal bee megalopta genalis. *Cell and tissue research*, 318(2):429–437, 2004. ISSN 0302-766X.

[52] Selig Hecht, Simon Shlaer, and Maurice Henri Pirenne. Energy, quanta, and vision. *The Journal of general physiology*, 25(6):819–840, 1942. ISSN 1540-7748.

[53] Christopher P Heesy and Margaret I Hall. The nocturnal bottleneck and the evolution of mammalian vision. *Brain, behavior and evolution*, 75(3):195–203, 2010. ISSN 0006-8977.

[54] Lionel Heng, Lorenz Meier, Petri Tanskanen, Friedrich Fraundorfer, and Marc Pollefeys. Autonomous obstacle avoidance and maneuvering on a vision-guided mav using on-board processing. In *2011 IEEE International Conference on Robotics and Automation*, pages 2472–2477. IEEE, 2011. ISBN 1612843859.

[55] Suzana Herculano-Houzel. Scaling of brain metabolism with a fixed energy budget per neuron: implications for neuronal activity, plasticity and evolution. *PloS one*, 6(3):e17514–e17514, 2011. ISSN 1932-6203.

[56] Cheng Hu, Farshad Arvin, Caihua Xiong, and Shigang Yue. Bio-inspired embedded vision system for autonomous micro-robots: the lgmd case. *IEEE Transactions on Cognitive and Developmental Systems*, 9(3):241–254, 2016. ISSN 2379-8920.

[57] Rui Huang and S. Ericson. An efficient way to estimate the focus of expansion. *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, pages 691–695, 2018.

[58] Maesoon Im and Shelley I Fried. Directionally selective retinal ganglion cells suppress luminance responses during natural viewing. *Scientific reports*, 6(1):1–9, 2016. ISSN 2045-2322.

[59] S Judge and F Rind. The locust dcmd, a movement-detecting neurone tightly tuned to collision trajectories. *Journal of Experimental Biology*, 200(16):2209–2216, 1997. ISSN 0022-0949.

[60] Song Kai-Tai and Huang Jui-Hsiang. Fast optical flow estimation and its application to real-time obstacle avoidance. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, volume 3, pages 2891–2896 vol.3, 2001. ISBN 1050-4729. doi: 10.1109/ROBOT. 2001.933060.

[61] Roland Kern, Norbert Boeddeker, Laura Dittmar, and Martin Egelhaaf. Blowfly flight characteristics are shaped by environmental features and controlled by optic flow information. *Journal of Experimental Biology*, 215(14):2501–2514, 2012. ISSN 0022-0949.

[62] In-Jung Kim, Yifeng Zhang, Masahito Yamagata, Markus Meister, and Joshua R Sanes. Molecular identification of a retinal cell type that responds to upward motion. *Nature*, 452(7186):478–482, 2008. ISSN 1476-4687.

[63] E Christopher Kirk. Effects of activity pattern on eye size and orbital aperture size in primates. *Journal of Human Evolution*, 51(2):159–170, 2006. ISSN 0047-2484.

[64] Kuno Kirschfeld. Die projektion der optischen umwelt auf das raster der rhabdomere im komplexauge von musca. *Experimental Brain Research*, 3(3):248–270, 1967. ISSN 0014-4819.

[65] Kuno Kirschfeld. The absolute sensitivity of lens and compound eyes. *Zeitschrift für Naturforschung C*, 29(9-10):592–596, 1974. ISSN 1865-7125.

[66] Karl Kral. Side-to-side head movements to obtain motion depth cues:: A short review of research on the praying mantis. *Behavioural Processes*, 43(1):71–77, 1998. ISSN 0376-6357.

[67] Holger G Krapp and Roland Hengstenberg. Estimation of self-motion by optic flow processing in single visual interneurons. *Nature*, 384(6608):463–466, 1996. ISSN 1476-4687.

[68] Trevor D Lamb, Shaun P Collin, and Edward N Pugh. Evolution of the vertebrate eye: opsins, photoreceptors, retina and eye cup. *Nature Reviews Neuroscience*, 8(12):960–976, 2007. ISSN 1471-0048.

[69] MF Land. Optics and vision in invertebrates. *Handbook of sensory physiology Vol VII/6B*, pages 471–592, 1981.

[70] Michael F Land. Visual acuity in insects. *Annual review of entomology*, 42(1):147–177, 1997. ISSN 0066-4170.

[71] Julien Lecoeur, Marie Dacke, Dario Floreano, and Emily Baird. The role of optic flow pooling in insect flight control in cluttered environments. *Scientific reports*, 9(1):1–13, 2019. ISSN 2045-2322.

[72] David N Lee. A theory of visual control of braking based on information about time-to-collision. *Perception*, 5(4):437–459, 1976. ISSN 0301-0066.

[73] David N Lee and Paul E Reddish. Plummeting gannets: a paradigm of ecological optics. *Nature*, 293 (5830):293–294, 1981. ISSN 1476-4687.

[74] DN Lee, DS Young, PE Reddish, S Lough, and TMH Clayton. Visual timing in hitting an accelerating ball. *The Quarterly Journal of Experimental Psychology*, 35(2):333–346, 1983. ISSN 0272-4987.

[75] G. Leuba and R. Kraftsik. Changes in volume, surface estimate, three-dimensional shape and total number of neurons of the human primary visual cortex from midgestation until old age. *Anatomy and Embryology*, 190(4):351–366, 1994. ISSN 1432-0568. doi: 10.1007/BF00187293. URL https://doi.org/10.1007/BF00187293.

[76] P. Lichtsteiner, C. Posch, and T. Delbruck. A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. ISSN 1558-173X. doi: 10.1109/JSSC.2007.914337.

[77] PG Lillywhite and SB Laughlin. Transducer noise in a photoreceptor. *Nature*, 277(5697):569–572, 1979. ISSN 1476-4687.

[78] Min Liu and Tobi Delbruck. Abmof: A novel optical flow algorithm for dynamic vision sensors. *arXiv preprint arXiv:1805.03988*, 2018.

[79] Shih-Chii Liu and Tobi Delbruck. Neuromorphic sensory systems. *Current opinion in neurobiology*, 20 (3):288–295, 2010. ISSN 0959-4388.

[80] Hugh Christopher Longuet-Higgins and Kvetoslav Prazdny. The interpretation of a moving retinal image. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 208(1173):385–397, 1980. ISSN 0080-4649.

[81] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *1981 Seventh International Joint Conference on Artificial Intelligence (IJCAI-81)*, pages 674–679, 1981.

[82] Aashi Manglik, Xinshuo Weng, Eshed Ohn-Bar, and Kris M Kitani. Forecasting time-to-collision from monocular video: Feasibility, dataset, and challenges. *arXiv preprint arXiv:1903.09102*, 2019.

[83] Graham R Martin. An owl's eye: Schematic optics and visual performance instrix aluco l. *Journal of comparative physiology*, 145(3):341–349, 1982. ISSN 0340-7594.

[84] AZO Materials. How does the human eye perceive light? photopic and scotopic vision, 2020. URL https://www.azom.com/article.aspx?ArticleID=14971. Accessed: June 2020.

[85] Moritz B Milde, Olivier JN Bertrand, Ryad Benosmanz, Martin Egelhaaf, and Elisabetta Chicca. Bioinspired event-driven collision avoidance algorithm based on optic flow. In *2015 International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*, pages 1–7. IEEE, 2015. ISBN 1467378887.

[86] Moritz B Milde, Alexander Dietmüller, Hermann Blum, Giacomo Indiveri, and Yulia Sandamirskaya. Obstacle avoidance and target acquisition in mobile robots equipped with neuromorphic sensory-processing systems. In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4. IEEE, 2017. ISBN 1467368539.

[87] Makoto Mizunami. Functional diversity of neural organization in insect ocellar systems. *Vision research*, 35(4):443–452, 1995. ISSN 0042-6989.

[88] Diederik Paul Moeys, Daniel Neil, Federico Corradi, Emmett Kerr, Philip Vance, Gautham Das, Sonya A Coleman, Thomas M McGinnity, Dermot Kerr, and Tobi Delbruck. Pred18: Dataset and further experiments with davis event camera in predator-prey robot chasing. *arXiv preprint arXiv:1807.03128*, 2018.

[89]   Robert S Molday and Orson L Moritz. Photoreceptors at a glance. *Journal of cell science*, 128(22):4039–4045, 2015. ISSN 0021-9533.

[90]   Tomoyuki Mori and Sebastian Scherer. First results in detecting and avoiding frontal obstacles from a monocular camera for micro unmanned aerial vehicles. In *2013 IEEE International Conference on Robotics and Automation*, pages 1750–1757. IEEE, 2013. ISBN 1467356433.

[91]   Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017. ISSN 1552-3098.

[92]   Ken Nakayama and Gerald H Silverman. The aperture problem—i. perception of nonrigidity and motion direction in translating sinusoidal lines. *Vision research*, 28(6):739–746, 1988. ISSN 0042-6989.

[93]   Miquel Perello Nieto. Visual pathways, 2020. URL `https://creativecommons.org/licenses/by-sa/4.0`. Accessed: June 2020.

[94]   D. E. Nilsson. Evolutionary links between apposition and superposition optics in crustacean eyes. *Nature*, 302(5911):818–821, 1983. ISSN 1476-4687. doi: 10.1038/302818a0. URL `https://doi.org/10.1038/302818a0`.

[95]   Jeremy E. Niven and Simon B. Laughlin. Energy limitation as a selective pressure on the evolution of sensory systems. *Journal of Experimental Biology*, 211(11):1792–1804, 2008. doi: 10.1242/jeb.017574. URL `https://jeb.biologists.org/content/jexbio/211/11/1792.full.pdf`.

[96]   H. Nyquist. Thermal agitation of electric charge in conductors. *Physical Review*, 32(1):110–113, 1928. doi: 10.1103/PhysRev.32.110. PR.

[97]   Nathalie Nériec and Claude Desplan. From the eye to the brain: Development of the drosophila visual system. *Current topics in developmental biology*, 116:247–271, 2016. doi: 10.1016/bs.ctdb.2015.11.032.

[98]   FJ Ollivier, DA Samuelson, DE Brooks, PA Lewis, ME Kallberg, and AM Komáromy. Comparative morphology of the tapetum lucidum (among selected species). *Veterinary ophthalmology*, 7(1):11–22, 2004. ISSN 1463-5216.

[99]   Garrick Orchard, Ryad Benosman, Ralph Etienne-Cummings, and Nitish V Thakor. A spiking neural network architecture for visual motion estimation. In *2013 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pages 298–301. IEEE, 2013. ISBN 1479914711.

[100] Federico Paredes-Vallés, Kirk Yannick Willehm Scheper, and Guido Cornelis Henricus Eugene De Croon. Unsupervised learning of a hierarchical spiking neural network for optical flow estimation: From events to global motion perception. *IEEE transactions on pattern analysis and machine intelligence*, 2019. ISSN 0162-8828.

[101] Parrot. Parrot bebop 2 - specifications, 2020. URL `https://www.parrot.com/drones/parrot-bebop-2`. Accessed: June 2020.

[102] DA Parry. The function of the insect ocellus. *Journal of Experimental Biology*, 24(3-4):211–219, 1947. ISSN 0022-0949.

[103] Bas J Pijnacker Hordijk, Kirk YW Scheper, and Guido CHE De Croon. Vertical landing for micro air vehicles using event-based optical flow. *Journal of Field Robotics*, 35(1):69–90, 2018. ISSN 1556-4959.

[104] Werner Reichardt. Evaluation of optical motion information by movement detectors. *Journal of Comparative Physiology A*, 161(4):533–547, 1987. ISSN 1432-1351. doi: 10.1007/BF00603660.

[105] Werner Reichardt and Hans Wenking. Optical detection and fixation of objects by fixed flying flies. *Naturwissenschaften*, 56(8):424–424, 1969. ISSN 0028-1042.

[106] Stephan Reichelt, Ralf Häussler, Gerald Fütterer, and Norbert Leister. Depth cues in human visual perception and their realization in 3d displays. In *Three-Dimensional Imaging, Visualization, and Display 2010 and Display Technologies and Applications for Defense, Security, and Avionics IV*, volume 7690, page 76900B. International Society for Optics and Photonics, 2010.

[107] Willi A Ribi. Fine structure of the first optic ganglion (lamina) of the cockroach, periplaneta americana. *Tissue and Cell*, 9(1):57–72, 1977. ISSN 0040-8166.

[108] F. Claire Rind and Peter J. Simmons. Seeing what is coming: building collision-sensitive neurones. *Trends in Neurosciences*, 22(5):215–220, 1999. ISSN 0166-2236. doi: https://doi.org/10.1016/S0166-2236(98)01332-0.

[109] Jens Rister, Dennis Pauls, Bettina Schnell, Chun-Yuan Ting, Chi-Hon Lee, Irina Sinakevitch, Javier Morante, Nicholas J Strausfeld, Kei Ito, and Martin Heisenberg. Dissection of the peripheral motion channel in the visual system of drosophila melanogaster. *Neuron*, 56(1):155–170, 2007. ISSN 0896-6273.

[110] Stéphane Ross, Narek Melik-Barkhudarov, Kumar Shaurya Shankar, Andreas Wendel, Debadeepta Dey, J Andrew Bagnell, and Martial Hebert. Learning monocular reactive uav control in cluttered natural environments. In *2013 IEEE international conference on robotics and automation*, pages 1765–1772. IEEE, 2013. ISBN 1467356433.

[111] Samuel Rossel. Binocular stereopsis in an insect. *Nature*, 302(5911):821–822, 1983. ISSN 1476-4687.

[112] Bodo Rueckauer and Tobi Delbruck. Evaluation of event-based algorithms for optical flow with ground-truth from inertial measurement sensor. *Frontiers in neuroscience*, 10:176, 2016. ISSN 1662-453X.

[113] Llewyn Salt, David Howard, Giacomo Indiveri, and Yulia Sandamirskaya. Differential evolution and bayesian optimisation for hyper-parameter selection in mixed-signal neuromorphic circuits applied to uav obstacle avoidance. *arXiv preprint arXiv:1704.04853*, 2017.

[114] Llewyn Salt, Giacomo Indiveri, and Yulia Sandamirskaya. Obstacle avoidance with lgmd neuron: towards a neuromorphic uav implementation. In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4. IEEE, 2017. ISBN 1467368539.

[115] Nitin J Sanket, Chethan M Parameshwara, Chahat Deep Singh, Ashwin V Kuruttukulam, Cornelia Fermüller, Davide Scaramuzza, and Yiannis Aloimonos. Evdodgenet: Deep dynamic obstacle dodging with event cameras. *arXiv preprint arXiv:1906.02919*, 2019.

[116] Alexander Schaub and Darius Burschka. Spatio-temporal prediction of collision candidates for static and dynamic objects in monocular image sequences. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 1052–1058. IEEE, 2013. ISBN 1467327557.

[117] Alexander Schaub, Daniel Baumgartner, and Darius Burschka. Reactive obstacle avoidance for highly maneuverable vehicles based on a two-stage optical flow clustering. *IEEE Transactions on Intelligent Transportation Systems*, 18(8):2137–2152, 2016. ISSN 1524-9050.

[118] G. R. Schlotterer. Response of the locust descending movement detector neuron to rapidly approaching and withdrawing visual stimuli. *Canadian Journal of Zoology*, 55(8):1372–1376, 1977. ISSN 0008-4301. doi: 10.1139/z77-179. URL https://doi.org/10.1139/z77-179.

[119] Lars Schmitz and Ryosuke Motani. Morphological differences between the eyeballs of nocturnal and diurnal amniotes revisited from optical perspectives of visual environments. *Vision research*, 50(10): 936–946, 2010. ISSN 0042-6989.

[120] Ivan R Schwab, Carlton K Yuen, Nedim C Buyukmihci, Thomas N Blankenship, and Paul G Fitzgerald. Evolution of the tapetum. *Transactions of the American Ophthalmological Society*, 100:187, 2002.

[121] M. Serdar Guzel and R. Bicker. Optical flow based system design for mobile robots. In *2010 IEEE Conference on Robotics, Automation and Mechatronics*, pages 545–550, 2010. doi: 10.1109/RAMECH.2010.5513134.

[122] Peter J Simmons. The function of insect ocelli. *Trends in Neurosciences*, 5:182–183, 1982. ISSN 0166-2236.

[123] Kahlouche Souhila and Achour Karim. Optical flow based robot obstacle avoidance. *International Journal of Advanced Robotic Systems*, 4(1):2, 2007. doi: 10.5772/5715.

[124] M Srinivasan, Shaowu Zhang, M Lehrer, and TS Collett. Honeybee navigation en route to the goal: visual flight control and odometry. *Journal of Experimental Biology*, 199(1):237–244, 1996. ISSN 0022-0949.

[125] Mandyam V Srinivasan, Michael Poteser, and Karl Kral. Motion detection in insect orientation and navigation. *Vision research*, 39(16):2749–2766, 1999. ISSN 0042-6989.

[126] MV Srinivasan, M Lehrer, WH Kirchner, and SW Zhang. Range perception through apparent image speed in freely flying honeybees. *Visual neuroscience*, 6(5):519–535, 1991. ISSN 1469-8714.

[127] S. Stabinger, A. Rodríguez-Sánchez, and J. Piater. Monocular obstacle avoidance for blind people using probabilistic focus of expansion estimation. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, 2016. doi: 10.1109/WACV.2016.7477608.

[128] Timo Stoffregen and Lindsay Kleeman. Event cameras, contrast maximization and reward functions: an analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12300–12308, 2019.

[129] Anna Lisa Stöckl, David Charles O'Carroll, and Eric James Warrant. Neural summation in the hawkmoth visual system extends the limits of vision in dim light. *Current Biology*, 26(6):821–826, 2016. ISSN 0960-9822.

[130] Jae Kyu Suhr, Ho Gi Jung, Kwanghyuk Bae, and Jaihie Kim. Outlier rejection for cameras on intelligent vehicles. *Pattern Recognition Letters*, 29(6):828 – 840, 2008. ISSN 0167-8655. doi: https://doi.org/10.1016/j.patrec.2007.11.019. URL http://www.sciencedirect.com/science/article/pii/S0167865507003893.

[131] Shin-ya Takemura. Connectome of the fly visual circuitry. *Microscopy*, 64(1):37–44, 2014. ISSN 2050-5698. doi: 10.1093/jmicro/dfu102.

[132] David C Van Essen, Charles H Anderson, and Daniel J Felleman. Information processing in the primate visual system: an integrated systems perspective. *Science*, 255(5043):419–423, 1992. ISSN 0036-8075.

[133] MCW Van Rossum, Brendan J O'Brien, and Robert G Smith. Effects of noise on the spike timing precision of retinal ganglion cells. *Journal of neurophysiology*, 89(5):2406–2419, 2003. ISSN 1522-1598.

[134] Hermann Wagner. Flow-field variables trigger landing in flies. *Nature*, 297(5862):147–148, 1982. ISSN 0028-0836.

[135] George Wald. Human vision and the spectrum. *Science*, 1945. ISSN 1095-9203.

[136] GK Wallace. Visual scanning in the desert locust schistocerca gregaria forskål. *Journal of Experimental Biology*, 36(3):512–525, 1959. ISSN 0022-0949.

[137] Eric Warrant. Vision in the dimmest habitats on earth. *Journal of Comparative Physiology A*, 190(10): 765–789, 2004. ISSN 0340-7594.

[138] Eric J Warrant. The remarkable visual capacities of nocturnal insects: vision at the limits with small eyes and tiny brains. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1717): 20160063, 2017. ISSN 0962-8436.

[139] Eric J Warrant, Shaun P Collin, and N Adam Locket. *Eye design and vision in deep-sea fishes*, pages 303–322. Springer, 2003.

[140] Edith A Widder. Bioluminescence in the ocean: origins of biological, chemical, and ecological diversity. *Science*, 328(5979):704–708, 2010. ISSN 0036-8075.

[141] Martin Wilson. The functional organisation of locust ocelli. *Journal of comparative physiology*, 124(4): 297–316, 1978. ISSN 0340-7594.

[142] Harry J Wyatt and Nigel W Daw. Directionally sensitive ganglion cells in the rabbit retina: specificity for stimulus direction, size, and speed. *Journal of Neurophysiology*, 38(3):613–626, 1975. ISSN 0022-3077.

[143] Chengxi Ye, Anton Mitrokhin, Cornelia Fermüller, James A Yorke, and Yiannis Aloimonos. Unsupervised learning of dense optical flow, depth and egomotion from sparse event data. *arXiv preprint arXiv:1809.08625*, 2018.

[144] Johannes M Zanker, Mandyam V Srinivasan, and Martin Egelhaaf. Speed tuning in elementary motion detectors of the correlation type. *Biological cybernetics*, 80(2):109–116, 1999. ISSN 0340-1200.

[145] Xu Zhang, Bin Xian, Bo Zhao, and Yao Zhang. Autonomous flight control of a nano quadrotor helicopter in a gps-denied environment using on-board vision. *IEEE Transactions on Industrial Electronics*, 62(10): 6392–6403, 2015. ISSN 0278-0046.

[146] Alex Zihao Zhu, Nikolay Atanasov, and Kostas Daniilidis. Event-based feature tracking with probabilistic data association. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4465–4470. IEEE, 2017. ISBN 150904633X.

[147] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Ev-flownet: Self-supervised optical flow estimation for event-based cameras. *arXiv preprint arXiv:1802.06898*, 2018.

[148] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 989–997, 2019.

[149] Simon Zingg, Davide Scaramuzza, Stephan Weiss, and Roland Siegwart. Mav navigation through indoor corridors using optical flow. In *2010 IEEE International Conference on Robotics and Automation*, pages 3361–3368. IEEE, 2010. ISBN 1424450381.

Bibliography

[1] Ryad Benosman, Charles Clercq, Xavier Lagorce, Sio-Hoi Ieng, and Chiara Bartolozzi. Event-based visual flow. *IEEE transactions on neural networks and learning systems*, 25(2):407–417, 2013. ISSN 2162-237X.

[2] J. Borenstein and Y. Koren. The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation*, 7(3):278–288, 1991. doi: 10.1109/70.88137.

[3] Blender Online Community. Blender - a 3d modelling and rendering package, 2018. URL `http://www.blender.org`.

[4] MAV Lab TU Delft. Optic flow field, 2021. URL `https://www.tudelft.nl/en/2021/tu-delft/appreciating-a-flowers-texture-color-and-shape-leads-to-better-drone-landings`. Accessed: March 2021.

[5] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. pages 226–231. AAAI Press, 1996.

[6] N. Wessendorp J. Dupeyroux, R. Dinaux. The obstacle detection avoidance dataset, 2021. URL `https://github.com/tudelft/ODA_Dataset`. Accessed: March 2021.

[7] Oussama Khatib. *Real-time obstacle avoidance for manipulators and mobile robots*, pages 396–404. Springer, 1986.

[8] P. Lichtsteiner, C. Posch, and T. Delbruck. A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. ISSN 1558-173X. doi: 10.1109/JSSC.2007.914337.

[9] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. Esim: an open event camera simulator. In *Conference on Robot Learning*, pages 969–982, 2018.

[10] Alexander Schaub and Darius Burschka. Spatio-temporal prediction of collision candidates for static and dynamic objects in monocular image sequences. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 1052–1058. IEEE, 2013. ISBN 1467327557.

[11] Stanford Artificial Intelligence Laboratory et al. Robotic operating system. URL `https://www.ros.org`.

[12] Nikhil Wessendorp, Raoul Dinaux, Julien Dupeyroux, and Guido de Croon. Obstacle avoidance onboard mavs using a fmcw radar, 2021.

[13] Guang-Zhong Yang, Jim Bellingham, Pierre E Dupont, Peer Fischer, Luciano Floridi, Robert Full, Neil Jacobstein, Vijay Kumar, Marcia McNutt, Robert Merrifield, et al. The grand challenges of science robotics. *Science robotics*, 3(14):eaar7650, 2018.