Dissertation

# Insect-Inspired Visual Self-Motion Estimation

Der Fakultät für Biologie
der Universität Bielefeld
vorgelegt von

Simon Strübbe

Zur Erlangung des Doktorgrades der Naturwissenschaften
Dr. rer. nat.

November 2018

Betreut von Prof. Dr. Martin Egelhaaf

# Acknowledgements

# Eidesstattliche Versicherung

Hiermit versichere ich, Simon Strübbe, dass die Dissertation von mir selbstständig angefertigt wurde und alle von mir genutzten Hilfsmittel angegeben wurden. Ich versichere, dass alle in Anspruch genommenen Quellen und Hilfen in der Dissertation vermerkt wurden.

_____          _____
Ort, Datum                              Unterschrift

# Contents

# 1 Introduction and Overview

Flying insects, like bees and flies, show an astonishing coordination of their flight maneuvers, which can not easily be matched yet by technical systems. Whereas the special physics of the insects' body allows such maneuvers, neurobiological research on insects addressed the question of the underlying information processing and neuronal control.

It is generally assumed that flight maneuvers, like obstacle avoidance, landing on objects and orientation in space rely to some extent on visual motion processing, as described in section 1.3. During locomotion, the insect uses spatio-temporal brightness changes on its retina to compute local optic flow vectors. The ommatidia, which are composed of little lenses, photoreceptors and further cells, are the components of the compound eyes of insects and lead to a spatial discretization of the light intensity distribution. The displacement of the light intensity on the retina during motion can be described through local vectors, the so-called optical flow vectors, which characterize the length of the displacement and its direction for a given time interval.

The motion-induced local flow vectors carry information about the spatial layout and moving objects in the environment as well as the motion of the insect itself. During translatory movement of the insect objects that are further away from it lead to smaller flow vectors than objects placed near to the insect. Hence, the insect can extract depth information about the environment during locomotion characterized by pure translation or, at least, by a translatory component. As described below, many insect species use a flight strategy where they separate to a large extent rotational movements from translational movements.

Not all retinal image displacements are due to the motion of the insect itself. Small objects like prey and conspecifics may move independently from the insect. Despite the ability to determine depth information about the evironment and to detect moving objects, the overall flow field can be used to estimate the motion of the insect itself, if one assumes that the environment is nearly static. As described below, specific neurons in the visual motion processing pathway are assumed to act as filters for self-motion, where the input of these filters are optic flow vectors and the output describes what kind of self-motion component one such specific neuron detects within its receptive field. The hypothesis that these specific neurons act as self-motion filters got popular in the late 90's and early 2000' where measurements of the sensitivity structure of these neurons were made.

Some biological findings argue in favor of this hypothesis, but other findings suggest
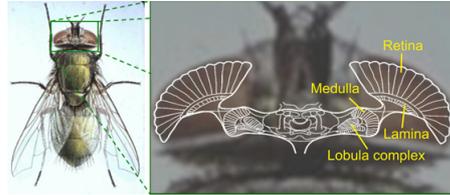
**Figure 1.1:** This figure schematically shows the visual motion pathway of the fly. After the transduction of the light intensities by the photoreceptors in the retina, the signals pass through the lamina, the medulla and the lobula complex. The motion detector model proposed by Reichardt and Hassenstein is assumed to be implemented by the neurons in the medulla. The tangential cells, which are the basis for the MFA, are located in the lobula complex (for a detailed description see section 1.1). This figure is taken from [31].

complementary interpretations. After discussing these arguments, this introduction will shift to the center of my doctoral project: A mathematical analysis of bio-inspired self-motion filters. The self-motion filters are idealized in a way that their receptive fields cover the whole sphere, what is not a restriction of the model because many self-motion filters with a smaller receptive field can be combined to one of these idealized self-motion filters. Each of these self-motion filters detects one of six possible self-motion components: Three translational and three rotational components. To detect self-motion helps the insect to orientate in space on a smaller time scale or provide information about the flown distance and direction on a larger time scale.

It could be shown in my thesis that the proposed bio-inspired self-motion filters have some commonalities with a technical approach to self-motion estimation. The proposed bio-inspired self-motion filters can be extended to an adaptive filter approach which takes changing spatial layouts of the environment into account. This extension leads to the hypothesis of the existence of certain neurons with a specific sensitivity structure.

## 1.1 The insect visual motion pathway and the implementation of the processing stages by simulation

Insects process visual motion in several subsequent layers of the visual system. The neurons which are supposed to act as self-motion filters are located in the labula plate, the third visual neuropile. Before the general characteristics of these cells are considered a short review of the processing stages preceding this filtering is given.
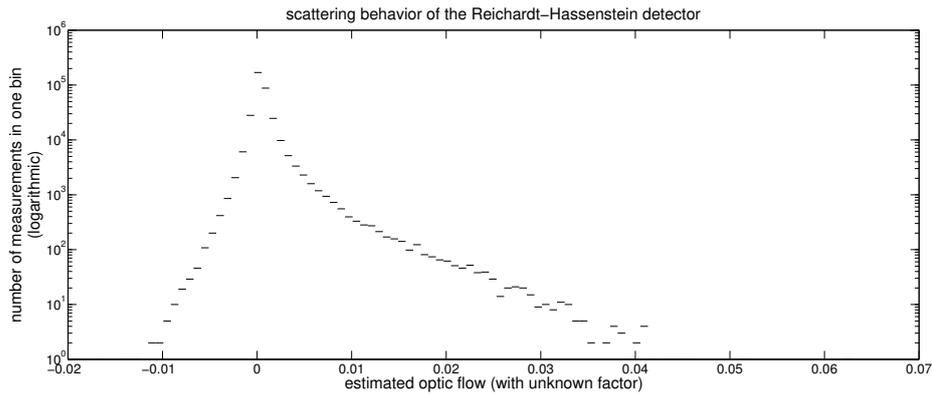
**Figure 1.2:** This figure shows the response distribution of 600*600 standard Reichardt-Hassenstein detectors, which cover a moving image. The image has random light intensities with a frequency statistic of $1/f$ and is shifted in the x-direction with a velocity of two pixels per frame. Hence, the true optic flow is the same all over the image. The detector responses are sorted in small bins. The small horizontal bars show the number of responses which fall in one specific bin. Although the pattern moves at a constant velocity across the entire array of movement detectors, their responses scatter tremendously depending on the local texture and contrast of the pattern. The responses may even indicate the opposite direction of motion (negative responses). Thus, the Reichardt-Hassenstein detector has a non-Gaussian scattering around the true optical flow: It has the maximum at zero with a more pronounced flank in the direction of the motion. The responses with a false sign are outweighted by responses with the correct sign. Hence, the Reichardt-Hassenstein detector does not represent the velocity veridically.
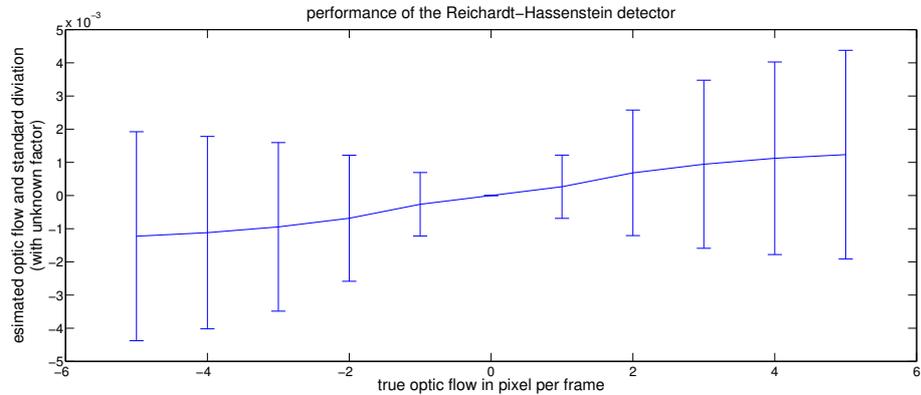
**Figure 1.3:** As in figure 1.2 a shifted image with $1/f$ statistic is the input of 600*600 Reichardt-Hassenstein detectors. The arms of one detector have a distance of five pixels. The velocitiy of the image is changed from minus five pixels to plus five pixels per frame. Although the Reichardt-Hassenstein detector has a non-Gaussian scattering (see Fig. 1.2) the mean response of all detectors is nearly linear with the velocity of the moving image within the tested velocitiy window. The standard deviation is very high for this type of detector. It exceeds the mean value by far.

### 1.1.1 The ommatidia and their photoreceptors

The compound eyes of flies and bees consist of an array of ommatidia. Each ommatidium has a broad acceptance angle for photons, which overlaps with the acceptance angle of neighboring ommatidia. The photons encounter eight photoreceptors within each ommatidium. The photoreceptors can be modeled by a combination of a Lipez transformation and a temporal low-pass filter. The Lipez transformation allows for processing of light intensities of several orders of magnitudes and maps these on the limited dynamic range of nerve cells. Due to the stochastic nature of the arriving photons, the time constants of the temporal low-pass filters have a value, which is a compromise between fast transmission and the reliability of the retinal output [50, 52, 51].

The two compound eyes of insects cover together nearly the whole sphere except of a small region, which is masked by the own body. For simplicity, a whole spherical viewing field is used in the model simulations of this study. The difference to biological reality is considererd to be sufficiently small. In the first part of the results (see section 2.3.6), artificial flow fields are used, whereas the second part of the results (see section 3.3.3) simulates ommatidia receiving rendered images from a simulated environment. In both parts the number of processed flow vectors reflects the number of ommatidia of the blowfly Calliphora [4] which are about 5000. One ommatidium is simulated in the second part of my results through a down-sampling of the image with the help of

a two-dimensional Gaussian filter centered in the middle of the simulated ommatidium which resembled the broad acceptance angle of the biological ommatidium.

### 1.1.2 The first visual neuropile

The photoreceptors transmit the light signals to the first visual neuropile, the lamina (see Fig. 1.1). The neurons in the lamina act as temporal band-pass filters. The band-pass filters increase the sensitivity to contrast changes of the transmitted signal [63, 100]. This band-pass filtering and the evolutionary adjusted photoreceptors are pre-steps to optimize the gained information for motion analysis [101]. These two pre-processing steps are omitted in the simulation in the second result part, because the rendering process used for generating the retinal input leads to images with an already restricted dynamic range of the light intensities. Hence an adaptation to several orders of light intensities is not needed. The band-pass filtering of lamina signal processing is useful for motion detection in the second visual neuropile. However, in the model simulations, a different motion detection mechanism is used which does not need such a preprocessing. The reasons why a different motion detector is used is discussed in section 3.5 and in the section "open questions of this study".

### 1.1.3 The second visual neuropile

It is assumed that the transformation of brightness changes into a motion signal is done by neurons in the second visual neuropile, the medulla (see Fig. 1.1). In the past, the activity of these neurons was hard to record, so that most characteristics of these neurons were indirectly determined at the level of the third visual neuropile. However, thanks to combining genetic and imaging methods together with novel electrophysiological methods the knowledge about the circuits underlying motion detection has been increasing tremendeously during the last years [10, 41, 64, 84].

A computational model explaining how the brightness changes in the retinal image are transformed into a motion signal was first introduced by Reichardt and Hassenstein [78]. This approach, which in its principles is still one basis explaining motion detection of insects today, correlates the light intensities at neighboring ommatidia. The so-called Reichardt-Hassenstein detector consists of two mirror-symmetrical half-detectors. One half-detector temporally delays the brightness signal from ommatidium A and multiplies this signal with the undelayed brightness signal from a neighboring ommatidium B. In a certain velocity range, the output signal of the half-detector increases the faster the image locally moves across these two ommatidia. For even higher image velocities the detector response decreases again. To detect motion in the opposite direction, i.e. from ommatidium B to ommatidium A, the same operations, delaying and multiplication of the incoming signal, are done with inverted roles of both ommatidia. The output signals of both half-detectors are subtracted to give an overall motion signal.

Even in the velocity range where the output of the Reichardt-Hassenstein detector increases monotonically with image velocity, the detector does not provide a veridical velocity signal. Rather, the detector response depends widely on the pattern structure of the image, especially on the local contrast. These dependencies are mostly seen as a deficiency of this detector. Therefore, many elaborations of the standard detector have been proposed to give a more adequate velocity signal (for example see [12, 69, 16]).

The pattern dependencies of the Reichardt-Hassenstein movement detector are immense and lead to the question why insects use such a correlation mechanism to detect motion. One potential answer was given by Potters et al. [77] showing that the Reichardt-Hassenstein detector is optimal for a low signal to noise ratio. Furthermore, Potters et al. [77] show that for motion detection a non-linear processing stage is needed. In the Reichardt-Hassenstein detector this non-linear processing stage is the multiplication of the delayed and undelayed brightness signals. However, a reliable velocity signal could be obtained with a gradient approach which shows much lower pattern dependencies and is explained below. Moreover, some authors argue that these pattern dependencies of the Reichardt-Hassenstein detector may provide important information about the structure of the surroundings, in a way, that the Reichardt-Hassenstein detector does not only provide a velocity signal, but also represents behaviorally relevant characteristics of environments [73, 49].

The standard approach of Reichardt and Hassenstein is a detector with a non-Gaussian scatter behavior (see Fig. 1.2). This leads to unexplored problems for self-motion estimation under the conditions used in the simulation part of this study. Self-motion estimation is studied here under the condition of changing and asymmetrical depth distributions together with the ability to estimate translation and rotation in parallel. If all these conditions come together, the Reichardt-Hassenstein detector leads to unexplained and intolerable high estimation errors. Hence, I decided to use the Lukas-Kanade gradient detector for the model simulation and left the question open of how the Reichardt-Hassenstein detector can be used for this task (see "open questions"). The gradient idea is simple: Velocity is measured in meters per seconds. So a motion signal can be obtained by dividing the temporal derivative of the brightness signal through the spatial derivative of the brightness signal. This simple idea is getting a bit more complicated if one considers not only one dimension but a two-dimensional moving image (see section 3.3.3).

During the testing of the Reichardt-Hassenstein detector the described standard version of the detector was combined with a temporal high-pass filter prior to the delayed and undelayed signal processing to get rid of the mean brightness. This procedure increases the performance of the Reichardt-Hassenstein detector for motion detection. For pattern moving at constant velocity the response distribution across the movement detector array has its peak at zero velocity with two flanks of both sides. If the true motion is positive the positive flank is increased (see Fig. 1.2). Although this scattering of reponses due to the pattern dependencies is non-Gaussian, the mean over several Reichardt-Hassenstein

detectors is proportional to the velocity in a certain velocity range (see Fig. 1.3). The standard deviation of the means of several detectors is larger than the velocity signal, which leads to a small signal-to-(pattern)-noise ratio.

### 1.1.4 The third visual neuropile

The motion signals from the medulla neurons are fed into the neurons of the lobula complex, also called tangential neurons (see Fig. 1.1). Whereas the receptive fields of the medulla neurons are small [88], since they combine the information of only neighboring ommatidia, the tangential neurons have receptive fields, which extend over a large part of the viewing sphere. Collision avoidance, landing on objects, learning the depth structure of a goal location and estimating self-motion need the combined information of several optic flow vectors, and the tangential cells may provide this information by pooling over a certain region of the field of view.

Tangential neurons have locally preferred directions for motion. They are excited locally by motion in a given direction and are inhibited by motion in the opposite direction (for example [46, 59]). The overall response behavior of tangential cells can be described by a mean preferred direction, for example from front to back, or more detailed through a field of all local preferred directions and local sensitivities to motion. This detailed motion sensitivity map indicates to which optical flow field the tangential neuron responds best. Which response fields are needed for self-motion detection are described below.

The Reichardt-Hassenstein detector, which is assumed to be realized by medulla neurons, represents the local image velocity only ambiguously, because its response increases with increasing velocity, reaches a maximum, but then decreases again. This response characteristic was first observed by behavioral experiments. The maximum velocity response depends on the texture of the presented images and the time constant of the delay filters in the Reichardt-Hassenstein detector [8].

Another property of tangential cells, which can be accounted for by the Reichardt-Hassenstein detector is that they do not only react to image velocity but also to higher-order temporal derivatives which occur, for example, as a consequence of sudden changes in image velocity (for example see [29, 102]).

The tangential cells themselves have non-linear spatial summation properties as well as show adaptation to continuous motion. These non-linearities do not match the overall linear behavior of self-motion filters and are discussed later.

## 1.2 Saccadic flight behavior of flying insects

Flying insects, like flies and bees, do not combine translatory and rotatory movements in arbitrary ways. Instead, they appear to avoid any rotational movement components during translatory flight segments, which are interrupted by quick rotatory phases called

saccades [81, 7, 14, 15, 38]. Since only translational optic flow depends on the distance of objects in the environment, this might be a strategy to easier gain depth information about the surroundings.

Although it is possible to get depth information about the environment from combined rotatory and translatory optic flow, the insect had then to determine this rotatory component computationally and eliminate the influence of the rotatory optic flow later. Hence, the behavioral separation of self-translation and self-rotation appears to be a strategy to get spatial information with little computational effort [31].

The saccadic flight behavior concerns this dissertation in two ways:

(1) In this doctoral project the general case of arbitrary self-motion with arbitrary translation and rotation is considered. Of course, a saccadic flight is a restricted motion from the perspective of arbitrary motion and is, thus, covered by the general case. However, if the self-translation or self-rotation is zero, the self-motion estimation problem is simplified and the later proposed adaptive self-motion method forfeits some adavantages over a non-adaptive method (see section 3.5). However, even if the insect tries to separate these two components, the separation is not perfect. During translatory movements, small rotations cannot be avoided and during the saccades, there is a small translatory motion component due to the inertia.

(2) The rotation during a saccade is so quick, up to several thousands of degrees per second, that problems may arise for optical flow detectors. The Reichardt-Hassenstein detector is then in a velocity range where the output of the detector decreases with increasing velocity [55], and the optic flow is totally out of the operating range of the used Lukas-Kanade detector. For the analyzed self-motion detection method this is not an insufficiency. The method can handle arbitrary magnitudes of self-motion if one assumes an underlying optic flow detector, the response of which is nearly linear to image motion.

## 1.3 The utilization of optic flow by flying insects

Optic flow is used by insects to solve a variety of tasks: collision avoidance, learning the spatial layout of goal locations, determining a landing place and self-motion detection. For all these tasks insects rely on the processing of visual motion information (see Fig. 1.1). This section argues that all these tasks are compatible with each other. Some tasks, particularly collision avoidance and learning the spatial layout of a goal location, mainly need depth information about the environment and, hence, rely only on the magnitude of the optical flow. Whereas other tasks, particularly searching for a landing place and self-motion detection rely on the directions of the optic flow vectors which form a specific optic flow pattern.

Collision avoidance may be accomplished by two different mechanisms, both being supported by experimental findings [56, 94, 95, 5]. The simpler of these mechanisms only needs the magnitude of the optic flow induced by a nearby object during flight.

During translational phases of locomotion objects that are nearby elicit larger optic flow values. However, even for a zero rotation component, the induced optic flow depends on the viewing direction relative to the direction of translation. Koenderink and van Doorn [57] determined the relationship between an optic flow vector $\vec{p}_i$ and the inverse distance $\mu_i$ for a given viewing direction $\vec{d}_i$ with the rotational self-motion component $\vec{r}$ and translational self-motion component $\vec{t}$:

$$\mu_i = -\frac{\vec{t} \cdot \left(\vec{p}_i - \vec{d}_i \times \vec{r}\right)}{1 - \left(\vec{t} \cdot \vec{d}_i\right)^2} \ ,$$

(1.1)

$$\mu_i = -\frac{\vec{t} \cdot \vec{p}_i}{1 - \left(\vec{t} \cdot \vec{d}_i\right)^2} \ .$$

(1.2)

In the second formula, the rotation is omitted as it is assumed for the intersaccades of insect flight. The problem of the dependency of the optic flow magnitude on the direction of translation can be overcome, if only the relative optic flow magnitude is considered, i.e. the ratio of the optic flow magnitude induced by a nearby object and the flow induced by the surrounding background. Indeed the insect shows a strong turning response away from the object if the object-induced optic flow is in front of a non-stationary slow moving background [56].

The second mechanism of collision avoidance takes the form of the optic flow pattern into account. If the insect flies forward the focus of expansion of the optic flow field is in front of the insect with optic flow vectors on the side, which point in the direction of the focus. An object on the left or right of the insect increases the magnitude of the optic flow vectors on this side. Hence the probability of eliciting an evasive turn to the other side is increased [94, 95, 5].

In general the direction of translation must be considered to estimate the distances to objects based on the optic flow (see equation 1.2). Bertrand et al. [6] show that in the case of a spherical field of view and translational self-motion, which is restricted to the azimuthal plane, the distances to objects can be determined without taking the direction of motion into account.

Another task where the magnitude of the optic flow and, therefore, depth information or the spatial layout of the environment plays a role is learning a goal location like the nest or food place in the case of hymenopteran insects. Whereas contrast, texture and color of objects in the vicinity of the goal location can be used by bees as cues for finding back to a goal, it was shown by Dittmar et al. [23, 24] that depth information alone is sufficient for finding back. In these experiments a feeder was nearly invisible and the objects surrounding the feeder were camouflaged by using the same texture for these objects as for the background.

As described above the optic flow is not directly proportional to the nearness of objects, since it also depends on the direction of translation. This is most likely not a problem for learning a goal location, because insects, such as bees or wasps, perform prototypical learning flights while facing the vicinity of the goal location and perform characteristic flights on increasing arcs. Hence, the determined distances are almost all perpendicular to the direction of motion and the denominator in equation (1.2) can be neglected, which leads to a proportionality between the optic flow and nearnesses of an objects.

In section 2.3.6 an adaptive mechanism for determining self-motion is introduced. In this approach, the agent remembers nine parameters about the depth distribution of the environment and updates these nine parameters continuously. This kind of memorizing the spatial layout of the environment differs from the learning of the spatial layout of a goal location during the above-mentioned learning flights of bees and wasps. Whereas in the adaptive self-motion estimation model the depth parameters are only relevant until the next update of the depth model is made, the memory of the spatial structure of the environment of a goal location should last in principle over the whole life-time of the insect.

A task where the form of the flow field plays a role and not just its magnitude is determining a landing site by detecting a suitable stationary object. If an object is used as a landing site and not as an obstacle that needs to be avoided, not just the distance to the object plays a role. One can assume that the form of the flow fields induced by the object plays a role if the insect will land on it.

The short introduction above indicates that, on the one hand, only the magnitude of the processed optic flow is necessary for determining depth information about the environment and, on the other hand, specific optic flow patterns are necessary in other behavioural contexts.

This dissertation is concerned with specific tangential neurons that have been concluded to operate as self-motion filters. These self-motion filters are sensitive to patterns in the optic flow field which are typically generated during translatory or rotatory movements of the insect. Self-motion detection uses the whole visible optic flow field and not just parts as the optic flow-based tasks described above. For a small section of the optic flow field, one cannot distinguish the corresponding translatory or rotatory self-motion components because translational optic flow appears very similar to rotational optic flow, if just a small section of the visual field is taken into account. However, if even these two optic flow fields are similar in a specific region of the visual field, they differ in sign for opposite viewing directions.

The self-motion filters are assumed to be linear filters, one filter for each self-motion component: Three translational and three rotational components [33]. Although the tangential neurons, which are assumed to act as such self-motion filters, cover only a part of the entire visual field, they can be combined linearly to cover the whole near-spherical visual field of insects. Although a spherical field of view is the optimal case for self-motion estimation based on biologically inspired methods as well as for technical systems, most

of the results also hold for an arbitrary field of view. However, a spherical field of view simplifies how the dependence on the depth structure of the environment can be taken into account. With a spherical field of view, only nine depth parameters are required to describe the whole depth information needed for self-motion estimation. Based on this finding a depth-adaptive self-motion method is proposed.

The structure of the optimal self-motion filters is obtained by the matched filter theorem [98]. The matched filter theorem states that the optimal filter to detect a specific pattern has the same form as the pattern. The matched filter theorem holds for patterns which are disturbed by Gaussian noise. The output of the Reichardt-Hassenstein detector as well as of the Lukas-Kanade detector depends deterministically on the given moving image. The output of the detectors depends strongly on the velocity of the moving image. So, every deviation from the true velocity of the image is regarded as (pattern) noise. Whereas the standard Reichardt-Hassenstein detector has a non-Gaussian noise distribution (see Fig. 1.2), the noise distribution of the Lukas-Kanade method is nearly Gaussian. If some tangential neurons act as self-motion filters they should have, according to the matched filter theorem, a sensitivity and preferred direction distribution which reflects the optic flow pattern generated by a specific self-motion component.

In principle six matched filters for self-motion detection would be sufficient for the six self-motion components. However, the restricted receptive fields of the tangential neurons lead to the assumption that several tangential neurons act together to determine jointly one self-motion component.

There is a principal ambiguity in detecting self-motion components based on optic flow: An optic flow field remains the same if the translational velocity is doubled and all distances are bisected. In behavioral experiments it could be shown that this ambiguity is not eliminated by bees. Therefore, bees are only able to find back to a goal if they fly through the same environment. If the environment is experimentally changed, the bees make characteristic errors in distance estimation [86, 104]. This ambiguity is also not eliminated by the self-motion method presented here. This study is solely concerned with optic flow-based self-motion estimation. Hence only five instead of six self-motion components can be determined at every point in time: The magnitude of the translational self-motion remains undefined.

Before introducing the mathematical and simulation results about self-motion estimation based on optic flow, a fundamental difference between the model self-motion filters and their biological counterparts, the tangential neurons, will be discussed. The matched filter theorem assumes linear self-motion filters. One such self-motion filter can be seen as an artificial neuron commonly used in artificial neural networks. These artificial neurons weight their individual inputs and summate them linearly to obtain an overall response. A non-linear activation function is not needed here because the problem is linearly solvable.

In biological neural networks, the connectivity between two neurons cannot generally be described by a single real number. The neurons are connected through chemical

processes in the synapses. Two biological neurons are not connected through a single synapse, but often by a complex connectivity structure with several synapses at different locations. This may lead to more or less pronounced non-linearities as observed also for the tangential neurons.

One example, where the biological tangential neurons differs from an idealized artificial neuron is the gain control mechanism [11]. Here the neurons react to parts of the visual field almost as much as during stimulation of the whole visual field. An simple artificial neuron would react less if only parts of the visual field are covered.

An other example is the study of Karmeier et. al [53]. This study could show exemplarily for two tangential neurons that the individual inputs are not summated linearly to an output signal. To obtain the individual inputs of a tangential neuron Karmeier et. al rely on earlier results [60], where a small light dot circulating with a small radius in the receptive field of the tangential neuron. By this procedure the local sensitivities and preferred direction of the tangential neuron can be determined. Karmeier et. al summate these local responses of the neuron to an overall theoretical output signal which is used for a comparison with experimentally measured output signal.

The measured output differs from the predicted linear output as follows: (1) Karmeier et al. analyzed a tangential neuron which has been assumed to react best to self-rotations. However, the theoretically determined preferred axis of rotation differs from the measured axis by 20°. (2) The spike rate of the analyzed tangential neuron never decreases to its resting level as predicted theoretically. (3) The response curve around the preferred direction is broader than the linear prediction.

A widely studied non-linearity which could lead to these differences is a gain control mechanisms [47, 26, 11, 85]. The response of the tangential neurons increases with increasing size of the motion stimulus, but only until a plateau level is reached. The plateau levels of the tangential neurons are different for different stimulus velocities. Hence, the output signal is relatively invariant with respect to the number of input signals, but still depends on velocity.

Although it is not yet entirely clear how the mentioned deviations of tangential cell responses from linearity can be explained in detail, the gain control does not affect the hypothesis that tangential neurons can be regarded as self-motion filters. The simulations of this study are based on an artificial 3D environment which has sufficient contrast in all viewing directions to allow determining the local motion signals. In real-world environments, this is not necessarily the case. In this situation, the gain control may help to filter the self-motion, if some local motion detectors cannot provide a result due to a small local contrast.

The analysis of the properties of the self-motion filters presented here is mathematically difficult. Hence, a short introduction will be given in the following. The self-motion filter approach, which is biologically inspired, is compared with a technical approach, the Koenderink van Doorn algorithm, which determines the self-motion components also on the basis of optic flow fields. In the analysis, certain commonalities and differences

are carved out. In the end, an algorithm is proposed, which is adaptive to the depth structure of the environment and combines the strength of both methods.

## 1.4 The two compared self-motion estimation methods and their commonalities

The biologically inspired self-motion filters are derived theoretically using the matched filters theorem [98]. Therefore, this approach is called matched filter approach (MFA). The MFA was first introduced by Franz and Krapp [36, 33, 32, 34, 35]. It determines the form of the filters not on the basis of electrophysiological experiments on tangential neurons, but rather determines theoretically optimal filters which are compared afterwards with the sensitivity and direction selectivity structure of certain tangential neurons.

A matched filter has the same form as the pattern it detects. Hence, in its simplest version the MFA would be based on six simple filters with forms being identical to the optical flow field induced by the six self-motion possibilities: three translational and three rotational ones. Since the translational optic flow depends on the distance of the environment in each viewing direction the MFA introduced by Franz and Krapp took this fact into account by assuming a mean distance distribution for the environment [33]. This distance assumption is fixed and is assumed not to change during self-motion estimation. The mean distances change the expected optical flow fields and hence the matched filters. The distance assumption resembles a free flight condition in an open environment. It is symmetrical along the azimuth: A certain assumed distance does not change for a rotation around the vertical axis. However, the distances are assumed to change in elevation. It is taken into account by Franz and Krapp that the ground of the environment is usually closer to the flying insect than the distances above the horizon.

This study introduces an MFA that makes more realistic assumptions with respect to the spatial layout of the environment by taking a changing depth distribution into account. It is shown later that a significantly changing environment leads to large errors in the self-motion estimates at points where a fixed distance assumption differs from the actual distances.

Another problem of the early MFA has already been solved by Franz and Krapp [35]. If the MFA is constructed as described above, a single self-motion filter does not only react to the self-motion component it is tuned to, but also to flow fields which are induced by other self-motion components. Franz and Krapp construct a matrix to solve this problem. This matrix, called coupling matrix, describes the coupling between the equations for the matched filters. The coupling matrix is identical to the unit matrix if the field of view is the whole sphere and if the distance is the same for all viewing directions. Hence, in this special case the coupling matrix can be omitted [13]. In general, the coupling matrix differs from the unit matrix. It uncouples the equations for self-motion estimation if its inverse is multiplied with the self-motion filters. With the fixed depth distribution

assumption by Franz and Krapp the coupling matrix has constant values and must be detemined only once for the whole trajectory of the agent. If the MFA adapts to the depth structure, the solely changing parts are the values of the coupling matrix. Hence, these values must be updated during flying along the trajectory.

The Koenderink van Doorn method [57] (KvD algorithm) for self-motion estimation, which is compared with the matched filter approach in this dissertation, does not assume anything about the distance distribution. Given a flow field, it computes the distance distribution in parallel to the self-motion components. For this, it starts with the assumption that all distances are the same and determines the self-motion components for this depth distribution. This leads to a first guess for the self-motion which is used to compute a better estimate of the distance distribution and vice versa. In one step, the distance distribution can be seen as fixed as in the MFA. Dahmen et al. [21] have shown that for one step a simpler version of the KvD algorithm, where some terms are omitted, is equivalent to the first introduced MFA [33]. This simpler MFA does not take into account a non-spherical field of view and different distance values. In this study (see section 2.3.4) the results of Dahmen et al. are generalized. The omitted terms in the KvD algorithms conform with the more complex MFA with coupling matrix [35]. This generalization shows that the two methods, the MFA and the KvD algorithm, are identical except for the treatment of the depth distribution.

The equivalence between the two methods can be illustrated as follow. If one assumes a fixed depth distribution, as does the MFA of Franz and Krapp, or one considers one step in the KvD algorithm, the problem of self-motion estimation is a linear problem. Mathematically it exists one unique solution for a linear estimation problem [20]. Although, the two methods are derived differently, they both represent this unique solution.

## 1.5 The difference between the two self-motion methods

The MFA by Franz and Krapp and the KvD algorithm differ in the treatment of the depth distribution. Whereas the MFA assumes a fixed depth distribution, the KvD algorithm computes the depth distribution in parallel to the self-motion components. If the depth distribution is not given, the linear problem changes to a non-linear problem. This concerns only the Koenderink van Doorn algorithm and the adaptive matched filter approach.

Koenderink and van Doorn [57] used an optimization criterion, which is only correct for a linear problem. However, in the non-linear case, which is assumed by the KvD algorithm, this optimization criterion leads to an estimator with a bias. This is shown mathematically in section 2.3.5. The proof of the occurrence of a bias considers two iteration steps, where the results from the first iteration are inserted in the second iteration. This leads to a non-vanishing error term, even if the number of given flow vectors is infinite. The non-vanishing error term depends on the standard deviation of

the local flow vectors, but not on the number of optical flow vectors (see section 2.5.1 and Fig. 2.1).

The Koenderink van Doorn algorithm can be modified in a way that the non-vanishing term does not occur. This is shown in section 2.3.3 and Fig. 2.1 by a numerical simulation where the number of flow vectors tends to infinity. The modified version behaves in its error reduction like an optimal estimator and exhibits no bias. The mathematical proof that this modified version is already the optimal estimator for the non-linear problem is still open. The proof is a problem of the non-linear estimation theory and is beyond the scope of this dissertation (see section "open questions").

## 1.6 Derivation and analysis of the adaptive matched filter approach

After showing the equivalence of the non-adaptive MFA and the KvD algorithm for the linear problem (see section 2.3.4), in which case the distances are assumed to be known, this study tranfers the depth distribution estimation ability of the KvD algorithm to the MFA, which is then able to adapt to the depth distribution. For this effort, the modified version of the KvD algorithm is used, because it involves no identifiable bias (see section 2.3.5).

The KvD algorithm is an iterative method which alternates the updates of the depth distribution and the self-motion components for a given optical flow field. The algorithm was developed under the assumption that only one point in time is considered. If one assumes that the depth distribution is changing only slightly from one trajectory point of the agent to the next (assuming a discretization of time points) and shows that the adaptation to the depth distribution is faster than the change of the depth distribution over time, one can derive a non-iterative self-motion estimation method. This method, called here adaptive matched filter approach, computes one update of the depth distribution at every point of the trajectory and uses the depth distribution (depth model parameters) of the former trajectory point as the initial guess.

As shown in Fig. 3.3 the adaptation to the depth distribution is exponential. This means that the error due to the depth model is reduced exponentially for every update. This exponential error reduction is a sufficient criterion that the adaptive MFA might function successfully, because even if there exists points where the depth distribution changes abruptly, the mean change in distances can be assumed to be on a linear scale.

The adaptation rate can be lowered, for example, realized by an update at only every 15th trajectory point (see Fig. 3.4), without losing its overall exponential adaptation behavior. In this case, because of the addition of the rotation angles, when the depth model is not updated, one can obtain better results if the self-centered depth model rotates with the agent. This rotation of the depth model can be done easily because, as described below, the depth model consists of only a few parameters which can be rotated

by simple linear operations (equations not shown).

Because the depth distribution must be remembered or explicitly represented by the insect or agent, until the depth update in the next time step is accomplished, it is desirable to represent the depth distribution by a few parameters in a depth model without losing too much relevant information about the depth. These parameters should store global properties of the depth distribution. Hence, an expansion of the depth values by a mathematically complete function set could be the answer to this problem. The spherical harmonic function set seems appropriate for such an expansion.

In physics spherical harmonic functions are used, for example, to expand the electromagnetic field on global properties. Translated to the depth structure, the zeroth order of the function set is the mean depth value, whereas the first order characterizes the asymmetry in x, y and z-direction and so on. With increasing order, the spherical harmonic functions contain more depth details.

An expansion with spherical harmonics functions entails the problem at which order should the expansion be cut off. An answer was found for a biologically relevant special case of the field of view. If the field of view is the whole sphere, which is nearly the case for many insects, the expansion is cut off automatically after the second order. More clearly, if the expanded depth model is put in the equations for the self-motion estimates, higher orders than the second order are dropped off. This astonishing result is proven in this study (see section 2.6.3). This shows that the chosen expansion is not only appropriate, but provides a clear relationship between the spherical harmonic functions and the self-motion estimation problem.

Whereas the zeroth and first order of an expansion of the depth distribution is intuitive, the needed second order is more difficult to visualize. The second order consists of quadrupoles. In particle physics, quadrupole magnets are used to focus electric streams. Translated to the depth expansion a quadrupole resembles narrowings or enlargements of the depth in the environment the agent flies through. As shown by simulation (see Fig. 3.4), the quadrupoles play only a significant role in such depth changes. Nonetheless, the dipoles (first order) play the predominant role in every investigated depth structure.

The quantitative properties of the adaptive MFA are studied by simulations in artificial 3D environments. Two 3D environments are investigated: A simple cube and a tube with narrowing and enlargement parts (see section 3.3.1). These two 3D environments are designed to study asymmetrical surroundings and significant changes in the distance distribution. Although these two 3D environments are simple, it is assumed that the general results hold also for more realistic situations, such as cluttered environment. The cluttered objects in such an environment lead to discontinuities of the distances. However, one can assume that global parameters like the asymmetries in x, y and z-direction, where the distance distribution is pooled over the whole field of view, change slightly from one trajectory point to the next. Nonetheless, at the discontinuity points a motion detector gives wrong results. Hence, a cluttered environment shifts the ratio between reliable and unreliable optic flow vectors.

Within the cube, the agent moved near the bottom to simulate an asymmetrical depth distribution which is nearly the same for the entire trajectory. Hence, the non-adaptive MFA can use a fixed depth model which includes these asymmetries in its filters. Within the tube with narrowing and enlargement parts, the agent moved at a constant altitude, that the distance to the ceiling changes in the different parts. This 3d environment is designed to exhibit large changes in the depth distribution to analyze how well the adaptation operates.

To obtain a flow field, a rendering program is used to get six images by six virtual cameras, respectively two for the x, y and z-direction. Two subsequent images are presented to a grid of elementary motion detectors. The optical flow output of the six directions is combined to a spherical flow field. It was shown by Potters et al. [77] that an elementary motion detector needs a non-linear operation to detect a flow value. Here a gradient detector, the Lukas-Kanade detector, is used as an elementary motion detector. The responses of the tested variant of the Reichardt-Hassenstein detector have such a large variance due to their pattern dependence (see Fig. 1.3) that the effects of the adaptive MFA are totally overlaid. It is beyond the scope of this study to test variants of the Reichardt-Hassenstein detector, which give a more suitable optical flow field. As described below, the properties of the non-adaptive and adaptive MFA do not depend on the choice of the elementary motion detector, because the MFA is simply based on a given flow field with a certain error distribution.

It is shown in section 3.4.3 that the overall error of the self-motion estimation can be divided into three additive parts. The first part is the error due to the scatter of the elementary motion detector responses. The second part is the error due to an inappropriate depth model. The third part is a second order error which can be neglected. Hence, the influence of the depth model can be studied independent from the chosen motion detector.

The simulations show that for changing depth distributions a fixed (non-adapting) depth model could not be found, which is appropriate for all these depth distributions. Further, the error due to an inappropriate depth model predominates the error due to the elementary motion detector response variance (see discussion of paper two for these statements). This leads to the hypothesis (see "open questions") that flying insects are assumed to have some kind of adaptation to the depth structure that the matched filter hypothesis for the tangential cells is meaningful.

## 1.7 Motion adaptation and the adaptive MFA

It was experimentally shown that certain elements in the visual motion pathway (Fig. 1.1) adapt to motion. Some examples, which are in contrast to the adaptive MFA, are given below. Some elements in the visual motion pathway adapt to unchanging (or only slowly changing) motion to make sudden changes in motion more salient. From

an information coding perspective, this is advantageous because unchanging inputs are redundant and one can concentrate on the input changes.

One example, following this idea, is an extension of the Reichardt-Hassenstein detector, where the actual output of the motion detector is divided by the temporally low-pass filtered output of the same detector [65]. This leads to a response decay over time for constant motion, which is also observed in tangential neurons [62]. While the response to constant motion decays, suddenly occuring objects or nearby objects are more salient, because motion signals induced by them differ from the constant background motion. This principle of dividing the output of a motion detector by its temporally low-pass filtered output is possible for every motion detector as well as for the Lukas-Kanade detector.

Another example of motion adaptation concerns the medulla neurons, although the consequences of this adptation has been measured postsynaptically to these neurons in certain tangential neurons. Experiments by Harris et. al [42] show that the sensitivity to contrast is shifted if the tangential neuron is stimulated for a certain time period by a constant velocity. This contrast dependence of adaptation is indirectly an adaptation to motion, because the Reichardt-Hassenstein detector depends on the pattern of the input images, especially on its contrast (for contrast dependencies of the Reichardt-Hassenstein detector see [82, 66]).

The adaptive MFA proposed in this study is not an adaptation in the sense of a redundance-free representation of the motion input. The adaptation here is an adaptation to a (slowly) changing depth distribution of the environment, where a few defined parameters about the environment must be remembered until the next processing step.

Whereas the response to background motion is lowered for a constant relative movement of the environment, the adaptive as well as the earlier non-adaptive MFA uses mainly this background motion to determine self-motion. Due to the amount of overall optical flow vectors, small regions containing salient objects play a subordinated role for self-motion estimation, because only the weighted sum of all flow vector are of interest for self-motion estimation.

The use of the slowly changing background motion for self-motion estimation is not necessarily contradictive to the motion adaptation described above. Self-motion estimation is in its core a linear problem, hence it is possible to obtain only the changes in motion, if the motion base value is stored by the insect or agent elsewhere.

# 2 Insect-Inspired Self-Motion Estimation with Dense Flow Fields – An Adaptive Matched Filter Approach

## 2.1 Abstract

The control of self-motion is a basic, but complex task for both technical and biological systems. Various algorithms have been proposed that allow the estimation of self-motion from the optic flow on the eyes. We show that two apparently very different approaches to solve this task, one technically and one biologically inspired, can be transformed into each other under certain conditions. One estimator of self-motion is based on a matched filter approach; it has been developed to describe the function of motion sensitive cells in the fly brain. The other estimator, the Koenderink and van Doorn (KvD) algorithm, was derived analytically with a technical background. If the distances to the objects in the environment can be assumed to be known, the two estimators are linear and equivalent, but are expressed in different mathematical forms. However, for most situations it is unrealistic to assume that the distances are known. Therefore, the depth structure of the environment needs to be determined in parallel to the self-motion parameters and leads to a non-linear problem. It is shown that the standard least mean square approach that is used by the KvD algorithm leads to a biased estimator. We derive a modification of this algorithm in order to remove the bias and demonstrate its improved performance by means of numerical simulations. For self-motion estimation it is beneficial to have a spherical visual field, similar to many flying insects. We show that in this case the representation of the depth structure of the environment derived from the optic flow can be simplified. Based on this result, we develop an adaptive matched filter approach for systems with a nearly spherical visual field. Then only eight parameters about the environment have to be memorized and updated during self-motion.

19

## 2.2 Introduction

Knowing one's self-motion is crucial for navigation, course control and attitude stabilization. Although GPS can provide information about the position and thus about the self-motion of an agent, this information depends on the reliability of the contact to satellites. GPS is not available to animals which have to rely on other means to gain information about their position and self-motion. A direct method to measure self-motion for a walking artificial or biological agent is counting the steps or, in the case of a wheeled vehicle, to monitor the turns of the wheels. In contrast, most flying agents rely on their visual system to solve this task.

The visual system of an artificial or biological agent obtains information about self-motion from pixel shifts in the retinal image over time. These pixel shifts can be described by vectors, the optic flow vectors. The flow vectors depend on both the rotational and translational components of self-motion as well as on the viewing direction. Moreover, for the translational component it also depends on the distance to objects in the environment.

For small translations and rotations, the flow vector for viewing direction $\vec{d_i}$ is given by (see [57] for derivation)

$$
\begin{aligned}
\vec{p_i}(\vec{t}, \vec{r}, \mu_i) &= -\mu_i \left( \vec{t} - \left( \vec{t} \cdot \vec{d_i} \right) \vec{d_i} \right) - \vec{r} \times \vec{d_i} \\
&= -\mu_i \left( \vec{d_i} \times \vec{t} \times \vec{d_i} \right) - \vec{r} \times \vec{d_i} \; ,
\end{aligned}
\tag{2.1}
$$

where $\mu_i$ is the inverse distance ("nearness") to the object seen in direction $\vec{d_i}$, $\vec{t}$ is the translation vector, and $\vec{r}$ is the rotation vector (defining a rotation of angle $r = |\vec{r}|$ around the axis given by $\vec{r}/|\vec{r}|$). According to equation (2.1), the flow vector $\vec{p_i}$ is perpendicular to the corresponding viewing direction $\vec{d_i}$. (We use 3d vectors to represent optic flow vectors. Otherwise one would have to define a tangential plane for every viewing direction.)

There are two principally different ways to use optic flow information for self-motion estimation. One way is to identify features in the retinal image at one point in time and find the same features at the next time point in order to compute their displacement. Several technical estimation methods for self-motion are based on these feature correspondences [43, 75]. They rely on a small number of corresponding image points and have to concern about outliers. Such estimation methods are widely used in the technical literature to determine the movement and/or the calibration of a camera [70]. When the self-motion steps are small or the frame rates are high an alternative way to extract self-motion information is possible. Instead of extracting features in the image the local pixel shifts on the retina, called optical flow, produced by the self-motion of the agent is determined through spatiotemporal intensity correspondences in the pattern. This can be done by a gradient-based detector like the Lucas-Kanade detector [3], which compares spatial and temporal derivatives, or by a biologically inspired detector, like the

elementary movement detector of the correlation type [78, 8], which uses spatiotemporal auto-correlation signals.

Here we propose a new adaptive approach which combines the advantages of two methods for self-motion estimation based on optical flow, the matched filter approach (MFA) proposed by Franz and Krapp [33] and an algorithm proposed by Koenderink and van Doorn (KvD algorithm) [57]. The MFA estimates self-motion by using linear filters, so called matched filters. Matched filters have the structural form of the pattern they are meant to detect [98] and are the optimal detectors for patterns, which are disturbed by Gaussian errors. In this case the linear filters of the MFA resemble ideal flow fields. Franz and Krapp [33] introduced six filters of this type for the six self-motion components, three for translation and three for rotation. Each of these six filters was tuned to one of the flow fields generated by the six self-motion components, although in general the filters react also to flow generated by the other self-motion components. There is one exception: For a flow field which covers the whole viewing sphere and for isotropic distances, i.e. in the center of a sphere, Borst and Weber [13] showed that model neurons acting as such linear filters are not influenced by other flow fields. To eliminate the influence of other flow fields in the case of an arbitrary field of view Franz et al. [35] introduced a coupling matrix and used its inverse to uncouple the output of the model neurons.

The KvD algorithm is iterative and tries to determine not only the self-motion components but also the distances of the moving agent to objects and surfaces in the environment. These distances influence the translational optic flow and therefore the self-motion estimate. The KvD algorithm starts with a simple distance estimate and determines in the same iteration preliminary self-motion components. In the next iteration these preliminary self-motion components are used as the basis for determining a better distance estimate, which is then used for improving the motion estimate. By the MFA the distances are taken into account statistically and are integrated without further changes in the filters. Dahmen et al. [21] have shown that one iteration step of the KvD algorithm corresponds to the MFA of Franz and Krapp [33] by assuming that some terms in the KvD algorithm are negligibly small. As an important step for the the development of an adaptive approach we show that the MFA with a specific coupling matrix is fully equivalent to one iteration step of the KvD algorithm and not just an approximation. The Gauss-Markov-Theorem [20] gives an explanation of this equivalence. This theorem guarantees the existence of a unique optimal estimator for a linear estimation problem. Both mentioned methods find this optimal solution, although the two approaches seem to be totally different.

The MFA was proposed to explain the motion sensitivity structure of the tangential cells in the fly visual system [33, 46, 58, 9, 27, 96, 31, 17, 19, 37, 105]. These cells are directionally selective for motion within their large receptive fields [48, 59, 30, 9, 27]. The spatial pattern of their motion sensitivity resembles flow fields on the retina generated by self-motion. Therefore, these cells were proposed to act as matched filters for self-motion estimation and to help the fly to solve visual orientation tasks [33]. However, since the

MFA makes a priori assumptions about the 3D structure of the environment, self-motion estimation deteriorates in an environment with variable distance distribution.

It is known that the fly's nervous system can adapt to sensory inputs [71, 42, 79, 12, 67, 62]. With this in mind, we propose a biologically inspired adaptive MFA, which adapts to the depth structure of the environment. This new model avoids the multiple iteration steps used by the KvD algorithm, on the one hand, and the hard-wired distance dependence of the MFA, on the other hand. The adaptive MFA extracts the depth structure from the optic flow field similar to the KvD algorithm. When the distances are not known the self-motion estimation problem becomes non-linear. Although the KvD algorithm is an optimal estimator in the linear case, it is, as we will show, a biased estimator in the non-linear case. The error in the quantities which are estimated does not converge to zero with increasing number of flow vectors. Therefore, we propose a modified version of the KvD algorithm. Numerical simulations indicate that the modified version has no bias.

On the basis of this modified KvD algorithm an adaptive MFA is developed that is inspired by a property of the visual system of insects: Insects have a field of view which nearly covers the whole sphere. It will be discussed that this property is beneficial for self-motion estimation and hence is desirable also for artificial agents which navigate by means of their visual system. The insect or agent should only adapt to the global properties of the depth structure and ignore irrelevant details. To achieve this, the inverse distances are expanded in a complete set of orthonormal functions, the spherical harmonics. It is desirable that the first-orders of this function set contribute most to the solution of the self-motion problem. We show that in the case of spherically distributed flow vectors all orders beyond the second-order of this function set do not contribute to self-motion estimation and can, thus, be neglected without losing information. Hence, if insects or artificial agents adapt to the depth structure they have to be sensitive only to low order depths functions, which are the dipole and quadrupole moments of the depth structure.

## 2.3 Results

A major objective of this study is to show that two well-established self-motion estimators are mathematically equivalent: The MFA equals one iteration step of the KvD algorithm when the inverse distances to objects in the environment are assumed to be known. To achieve this, we derive the MFA in an alternative way.

We then show that the KvD approach leads to an biased estimator in the general case when distances are unknown and have to be estimated together with the self-motion parameters. We present a modification of the KVD iteration equation that removes the bias and derive an adaptive MFA from this corrected version, which includes a simple but, with respect to self-motion estimation, complete depth model.

Before dealing with these topics, both the basic equations underlying the MFA approach and the KvD algorithm need to be introduced.

### 2.3.1 The matched filter approach

In the original MFA [33, 35, 32, 34] the depth structure of the environment is not determined from the current flow field but described statistically with a fixed distribution that is assumed to be known. The first statistical parameter considered in [35] is the average inverse distance $\bar{\mu}_i$, which is measured in every viewing direction over a number of learning flights in different environments. The variability of the distances is given by the covariance matrix $C_\mu$. Secondly, the noise in the flow measurement is determined for each viewing direction $n_i$ where a zero mean is assumed. The noise values are combined in the covariance matrix $C_n$. The third statistical parameter is the distribution of the translations $t$. It is assumed that the agent does not translate in every possible direction with the same probability. The corresponding statistical parameter is the covariance matrix $C_t$.

An optic flow vector $\vec{p}_i$ has only two degrees of freedom because it is the projection of object motion on the retina and thus orthogonal to the corresponding viewing direction $\vec{d}_i$. To consider only these degrees of freedom Franz et al. [35] introduce a local two-dimensional vector space for each viewing direction $\vec{d}_i$ which is orthogonal to the direction $\vec{d}_i$:

$$
\begin{aligned}
x_i &= \vec{p}_i \cdot \vec{u}_i = \vec{p}_i^0 \cdot \vec{u}_i + n_{x,i} & (2.2) \\
y_i &= \vec{p}_i \cdot \vec{v}_i = \vec{p}_i^0 \cdot \vec{v}_i + n_{y,i} & (2.3)
\end{aligned}
$$

where $\vec{u}$ and $\vec{v}$ are the basis vectors of the new vector space. The values $x_i$ and $y_i$ represent the two degrees of freedom of $\vec{p}_i$. The measured vector $\vec{p}_i$ consists of the true optic flow vector $\vec{p}_i^0$ and an additive noise $n_i$.

In [35] the weights $W$ for the matched filters which are multiplied with the optic flow components $\vec{x}$ (where $\vec{x}$ is a 2N dimensional vector containing all flow components $x_i$ and $y_i$, $i = 1, 2, \ldots, N$) to estimate the six self-motion components $\vec{\theta}^{\text{est}}$,

$$
\vec{\theta}^{\text{est}} = W \cdot \vec{x} \,, \tag{2.4}
$$

are derived by a least-square principle:

$$
e = E\left( \left\| \vec{\theta} - \vec{\theta}^{\text{est}} \right\|^2 \right) \,, \tag{2.5}
$$

where $\vec{\theta}$ are the true self-motion components. The weight matrix that minimizes the error $e$ is:

$$
W = \left( F^T C^{-1} F \right)^{-1} \cdot F^T \cdot C^{-1} \,. \tag{2.6}
$$

The covariance matrix $C$ combines the covariance matrices $C_\mu$, $C_n$ and $C_t$. The matrix $F$ is given by

$$F = \begin{pmatrix} -\bar{\mu}_1 \vec{u}_1, & -\bar{\mu}_1 \vec{v}_1, & -\bar{\mu}_2 \vec{u}_2, & -\bar{\mu}_2 \vec{v}_2, & \cdots \\ \vec{u}_1 \times \vec{d}_1, & \vec{v}_1 \times \vec{d}_1, & \vec{u}_2 \times \vec{d}_2, & \vec{v}_2 \times \vec{d}_2, & \end{pmatrix}^T , \qquad (2.7)$$

where $\bar{\mu}_i$ is the average or expected inverse distance for direction $\bar{d}_i$. The introduction of the matched filter approach is kept short because an alternative derivation of this approach is introduced in section 2.3.3, which is more suitable for the comparison with the KvD algorithm.

## 2.3.2 The Koenderink-van-Doorn (KvD) algorithm

As described by Koenderink and van Doorn [57], a straightforward approach for estimating the self-motion parameters is to find, in accordance with equation (2.1), a translation vector $\vec{t}$, a rotation vector $\vec{r}$ and inverse distances $\{\mu_i\}_{i=1,2,...,N}$ that minimize the mean squared error between the theoretical optical flow vectors according to equation (2.1), $\{\vec{p}_i(\vec{t}, \vec{r}, \mu_i)\}_{i=1,2,...,N}$ and the measured optical flow vectors $\{\vec{p}_i\}_{i=1,2,...,N}$:

$$e(\vec{t}, \vec{r}, \{\mu_i\}) = \frac{1}{N} \sum_{i=1}^{N} \left\| \vec{p}_i(\vec{t}, \vec{r}, \mu_i) - \vec{p}_i \right\|^2 . \qquad (2.8)$$

$$= \frac{1}{N} \sum_{i=1}^{N} \left\| -\mu_i \left( \vec{t} - \left( \vec{t} \cdot \vec{d}_i \right) \vec{d}_i \right) - \vec{r} \times \vec{d}_i - \vec{p}_i \right\|^2 \qquad (2.9)$$

Since the optic flow vector (see equation [2.1]) depends on the product of $\vec{t}$ and $\mu_i$, the same flow vector is obtained by multiplying $\vec{t}$ and dividing $\mu_i$ by the same factor. Thus, an additional constraint is imposed to ensure convergence of the minimization procedure. The algorithm described in [57] uses the constraint $\|\vec{t}\| = 1$ and, starting from an initial guess, solves for the motion parameters by iterating the following equations derived from equation (2.8) until convergence:

$$\mu_i = -\frac{\vec{t} \cdot \left( \vec{p}_i - \vec{d}_i \times \vec{r} \right)}{1 - \left( \vec{t} \cdot \vec{d}_i \right)^2} , \qquad (2.10)$$

$$\vec{t} = -\xi \left\{ \langle \mu \vec{p} \rangle + \vec{r} \times \langle \mu \vec{d} \rangle - \left\langle \mu^2 \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle \right\} , \qquad (2.11)$$

$$\vec{r} = \left\langle \vec{p} \times \vec{d} \right\rangle + \vec{t} \times \left\langle \mu \vec{d} \right\rangle + \left\langle \left( \vec{r} \cdot \vec{d} \right) \vec{d} \right\rangle . \qquad (2.12)$$

where $\xi$ is a Lagrange multiplier ensuring the constraint $\|\vec{t}\| = 1$. The brackets $\langle \rangle$ stand for the average over all viewing directions, i.e. the summation over all directions $i = 1, 2, \ldots, N$ divided by the number of directions $N$, e.g. $\langle \mu \vec{p} \rangle = \frac{1}{N} \sum_{i=1}^{N} \mu_i \vec{p}_i$.

### 2.3.3 Alternative derivation and properties of the coupling matrix in the MFA

To compare the two self-motion estimators one needs another mathematical form of the coupling matrix and the filters as given by the MFA (see 2.3.1). This can be achieved in two different ways. One can either transform the original equations (2.4), (2.6) and (2.7) of the MFA or derive the MFA in an alternative manner and show the equivalence to the original MFA afterward. Here, the second way is chosen. The matched filters will be derived according to the theory of optimal filters. The coupling of the estimated self-motion parameters will then be determined by inserting the filters in the flow equation (2.1).

The optimal weights in the MFA depend on the statistics of the distances, of the noise and of the preferred translations. Here we assume that nothing is known about these distributions and thus consider the simplest case: The noise values are set to the same value independent of the viewing direction. We assume no preference for specific translation directions. The average inverse distances $\bar{\mu}_i$ are regarded as known.

The theory of optimal filters states that for a uniform Gaussian noise the best linear filter is a matched filter which has the same form as the pattern it has to detect [98]. Therefore, the templates for estimating translational flow fields have the form of a translational flow field.

$$\vec{T}_{ia}^{t} = -\mu_i(\vec{d}_i \times \vec{e}_a \times \vec{d}_i) . \tag{2.13}$$

We have three translational templates one for each possible direction of translation represented by the three basis vectors $\vec{e}_a$ ($a = 1, 2, 3$).

Similarly, we have three rotational templates

$$\vec{T}_{ia}^{r} = -\vec{e}_a \times \vec{d}_i. \tag{2.14}$$

The three translational and three rotational templates $\vec{T}_a^t$ and $\vec{T}_a^r$ will be called "standard templates" .

The scalar product $\left\langle \vec{T} \cdot \vec{p} \right\rangle$ of a flow field $\vec{p}$ and a template $\vec{T}$, where the brackets stand for the mean over all viewing directions, can be interpreted as the output of a specific model neuron $a = \left\langle \vec{T} \cdot \vec{p} \right\rangle$. In general, the model neurons do not only react to the flow fields they are tuned to, but also to other flow fields. To solve this problem Franz et al. [35] introduced a matrix $\left( F^T C^{-1} F \right)^{-1}$ (see equation (2.6)) which compensates for the coupling to other flow fields. The coupling between the self-motion estimates and therefore the coupling matrix are determined be inserting the templates $\vec{T}_a^t$ and $\vec{T}_a^r$ in the flow equation (2.1). For this, the translation $\vec{t}$ and the rotation $\vec{r}$ must be separated

into their components, $\vec{t} = t_1\vec{e}_1 + t_2\vec{e}_2 + t_3\vec{e}_3$ and $\vec{r} = r_1\vec{e}_1 + r_2\vec{e}_2 + r_3\vec{e}_3$, with the six self-motion parameters $t_1$, $t_2$, $t_3$ and $r_1$, $r_2$, $r_3$.

Since the cross product is linear $\left(\alpha\,\vec{a} + \beta\,\vec{b}\right) \times \vec{c} = \alpha\,(\vec{a} \times \vec{c}) + \beta\left(\vec{b} \times \vec{c}\right)$, the overall flow field is the sum of the six standard templates $\vec{T}_A$ ($A = 1, 2, \ldots, 6$) weighted by the six self-motion components $\theta_A$:

$$
\begin{aligned}
\vec{p}_i &= t_1\vec{T}_{i1}^t + t_2\vec{T}_{i2}^t + t_3\vec{T}_{i3}^t + r_1\vec{T}_{i1}^r + r_2\vec{T}_{i2}^r + r_3\vec{T}_{i3}^r \; , \\
&= \theta_1\vec{T}_{i1} + \theta_2\vec{T}_{i2} + \theta_3\vec{T}_{i3} + \theta_4\vec{T}_{i4} + \theta_5\vec{T}_{i5} + \theta_6\vec{T}_{i6} \; .
\end{aligned}
\tag{2.15}
$$

Following our notation, the response of model neuron $a_A$ with corresponding template $\vec{T}_A$ to the flow field $\vec{p}$ (equation (2.15)) is

$$
a_A = \left\langle \vec{T}_A \cdot \vec{p} \right\rangle = \left\langle \vec{T}_A \cdot \sum_B \vec{T}_B \theta_B \right\rangle = \sum \left\langle \vec{T}_A \cdot \vec{T}_B \right\rangle \theta_B = \sum_B \hat{M}_{AB} \theta_B \; .
\tag{2.16}
$$

Combining the responses of all six model neurons in one equation gives

$$
\vec{a} = \hat{M} \cdot \vec{\theta} \; ,
\tag{2.17}
$$

where the vectors $\vec{a}$ and $\vec{\theta}$ are considered as six dimensional vectors. Each entry of the $6 \times 6$ dimensional matrix $\hat{M}$, the coupling matrix, can be seen as the generalized scalar product of two of the six standard templates:

$$
\hat{M}_{AB} = \left\langle \vec{T}_A \cdot \vec{T}_B \right\rangle
\tag{2.18}
$$

We can also write the coupling matrix as

$$
M = \begin{pmatrix} M^{tt} & M^{tr} \\ M^{rt} & M^{rr} \end{pmatrix} \; ,
\tag{2.19}
$$

where the indices $t$ and $r$ of the $3 \times 3$ sub-matrices indicate which templates are multiplied.

Using the inverse of the coupling matrix we can estimate the motion parameters $\vec{\theta}$ from the responses of the model neurons,

$$
\vec{\theta}^{\text{est}} = \hat{M}^{-1}\vec{a} = \hat{M}^{-1} \begin{pmatrix} \left\langle \vec{T}_1^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_2^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_3^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_1^r \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_2^r \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_3^r \cdot \vec{p} \right\rangle \end{pmatrix} \; .
\tag{2.20}
$$

The product $\left\langle \vec{T} \cdot \vec{p} \right\rangle$ and the multiplication with $\hat{M}^{-1}$ are linear transformations of the optical flow. One can therefore define new templates $T'$ which include the linear transformation given by the matrix $\hat{M}^{-1}$.

Dahmen et al. [21] tested previously the self-motion estimation performance for different fields of view. The self-motion estimation performance even for error prone flow vectors is high, if the flow fields corresponding to different self-motion components differ essentially over the field of view. For a restricted field of view, for example a small region in front of the agent, upward translation cannot be distinguished from a pitch rotation of the agent. In this case the coupling matrix with constant distances becomes nearly singular and cannot be properly inverted.

In section 2.6.4 of the appendix it is shown by means of a coordinate transformation of two dimensional flow vectors (in tangent planes) into three dimensional ones (on the sphere) that equation (2.20) is equivalent to equation (2.4) with the weights given by equation (2.6).

### Properties of the coupling matrix

The equivalence between the MFA and one iteration of the KvD algorithm is shown in three steps. The first step was the alternative derivation of the MFA described above. The following second step is a simplification of the four sub-matrices of the coupling matrix.

Three cases have to be considered when calculating the entries of the matrix: the scalar product between two translational templates, the scalar product between two rotational templates and the scalar product between a translational and a rotational template.

The scalar product between two translational templates leads to the following expression:

$$
\begin{aligned}
\left( M^{tt} \right)_{ab} &= \left\langle \vec{T}_a^t \cdot \vec{T}_b^t \right\rangle \\
&= \left\langle \mu^2 \right\rangle \vec{e}_a \cdot \vec{e}_b - \left\langle \mu^2 \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right\rangle \; .
\end{aligned}
\tag{2.21}
$$

The scalar product between two rotational templates results in

$$
\begin{aligned}
\left( M^{rr} \right)_{ab} &= \left\langle \vec{T}_a^r \cdot \vec{T}_b^r \right\rangle \\
&= \vec{e}_a \cdot \vec{e}_b - \left\langle \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right\rangle \; .
\end{aligned}
\tag{2.22}
$$

Finally, the scalar product between a translational and a rotational template leads to

$$
\begin{aligned}
\left( M^{tr} \right)_{ab} &= \left( -M^{rt} \right)_{ab} = \left\langle \vec{T}_a^t \cdot \vec{T}_b^r \right\rangle \\
&= \left( \vec{e}_b \times \vec{e}_a \right) \cdot \left\langle \mu \vec{d} \right\rangle \; .
\end{aligned}
\tag{2.23}
$$

27

Equation (2.21) informs us that the estimates of the three translation parameters $t_1$, $t_2$ and $t_3$ are coupled unless the term $\left\langle \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right\rangle$ in $M^{tt}$ is proportional to the identity matrix. The same holds for the coupling between the three rotation parameters $r_1$, $r_2$ and $r_3$ described by $M^{rr}$, see equation (2.22). Similarly, the term $\left\langle \mu \vec{d} \right\rangle$ in $M^{tr}$, equation (2.23), must be zero for the translation and rotation estimates to be uncoupled.

**The case of constant distances and a spherical field of view**

Borst and Weber [13] showed that for viewing directions homogeneously covering the whole sphere, and for identical distances ($\mu_i$= constant for all $i$) the model neurons respond only to the components of the flow field they are tuned to. This result can be easily verified within the conceptual framework provided here by replacing the sums in (2.21), (2.22) and (2.23) with integrals over the unit sphere and by introducing spherical coordinates. The direction vectors $\vec{d}_i$ are then replaced by the vectors $\vec{d}_{\vartheta\varphi}$ that depend on the elevation angle $\vartheta$ and azimuth angle $\varphi$. In appendix 2.6.3 it is shown that the direction vectors $\vec{d}_{\vartheta\varphi}$ in the spherical coordinate system have the same form as the three real-valued dipole functions of the spherical harmonics. Due to the orthogonality of the spherical harmonics the integral $\int \left( \vec{e}_a \cdot \vec{d}_{\vartheta\varphi} \right) \cdot \left( \vec{e}_b \cdot \vec{d}_{\vartheta\varphi} \right) \sin \vartheta d\vartheta d\varphi$ becomes zero in the case of spherically distributed flow vectors, if $\vec{e}_a$ and $\vec{e}_b$ denote different basis vectors. The scalar product between a translational and a rotational template (2.23) leads to the integral $\int \left( \vec{e}_b \times \vec{e}_a \right) \cdot \vec{d}_{\vartheta\varphi} \sin \vartheta d\vartheta d\varphi$ which can be regarded as the product between a first-order dipole function and the zeroth order spherical harmonic function (which is a constant). Due to the orthogonality of the spherical harmonics this integral is zero.

### 2.3.4 The relationship between the MFA and the KvD algorithm

In a final step we will show that the MFA with the coupling matrix is equivalent to one iteration of the KvD algorithm [57] for known distances. Hence, the two approaches do not represent principally different methods, but rather one method with two different ways of dealing with the depth structure. An additional way to take the depth structure into account is given in section 2.3.6 where an adaptive MFA is proposed.

Dahmen et al. [21] have already demonstrated that a MFA can be derived from the KvD algorithm, if certain terms are small and can be neglected. We will show that these terms are identical to entries of the coupling matrix, which was derived in the previous chapter.

**The equivalence of the MFA and one iteration step of the KvD algorithm**

Not only in the MFA, but also in the KvD approach coupling terms can be identified. In the translation equation (2.11), the second term on the right side comprises the rotation. The rotation equation (2.12) in turn contains a translational term. These terms were called

"apparent rotation" and "apparent translation" by Dahmen et al. [21], and they couple translation and rotation. The third terms in both equations couple different components of the translation or the rotation whenever $\left\langle \mu^2 \vec{d} \otimes \vec{d} \right\rangle$ in $\left\langle \mu^2 \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle = \left\langle \mu^2 \vec{d} \otimes \vec{d} \right\rangle \cdot \vec{t}$ or $\left\langle \vec{d} \otimes \vec{d} \right\rangle$ in $\left\langle \left( \vec{r} \cdot \vec{d} \right) \vec{d} \right\rangle = \left\langle \vec{d} \otimes \vec{d} \right\rangle \cdot \vec{r}$ contain off-diagonal components. The following derivation will show that these coupling terms are equivalent to the terms of the coupling matrix in the MFA.

The derivation starts with the MFA including the coupling matrix. From equation (2.17) one obtains together with the equations for the coupling matrix (2.21), (2.22) and (2.23):

$$\vec{a} \; = \; \hat{M} \cdot \vec{\theta} \; = \; \left( \begin{array}{cc} M^{tt} & M^{tr} \\ M^{rt} & M^{rr} \end{array} \right) \cdot \left( \begin{array}{c} \vec{t} \\ \vec{r} \end{array} \right) = \left( \begin{array}{c} \left\langle \vec{T}_1^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_2^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_3^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_1^r \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_2^r \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_3^r \cdot \vec{p} \right\rangle \end{array} \right) \tag{2.24}$$

$$\left( M^{tt} \right)_{ab} \; = \; \left\langle \mu^2 \right\rangle \vec{e}_a \cdot \vec{e}_b - \left\langle \mu^2 \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right\rangle \tag{2.25}$$

$$\left( M^{tr} \right)_{ab} \; = \; \left\langle \mu \left( \vec{e}_a \times \vec{e}_b \right) \cdot \vec{d} \right\rangle \tag{2.26}$$

$$\left( M^{rt} \right)_{ab} \; = \; - \left\langle \mu \left( \vec{e}_a \times \vec{e}_b \right) \cdot \vec{d} \right\rangle \tag{2.27}$$

$$\left( M^{rr} \right)_{ab} \; = \; \vec{e}_a \cdot \vec{e}_b - \left\langle \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right\rangle \tag{2.28}$$

On the left side of equation (2.24), the vectors $\vec{t}$ and $\vec{r}$ are multiplied with the entries of the coupling matrix. On the right side the optical flow vectors $\vec{p}$ are multiplied with the templates. After some algebraic simplifications, which are given in appendix 2.6.1, and a rearrangement of the terms one obtains the known equations for translation (2.11) and rotation (2.12) of the KvD algorithm:

$$\vec{t} \; = \; -\frac{1}{\langle \mu^2 \rangle} \left\{ \langle \mu \vec{p} \rangle + \vec{r} \times \left\langle \mu \vec{d} \right\rangle - \left\langle \mu^2 \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle \right\} \; ,$$

$$\vec{r} \; = \; \left\langle \vec{p} \times \vec{d} \right\rangle + \vec{t} \times \left\langle \mu \vec{d} \right\rangle + \left\langle \left( \vec{r} \cdot \vec{d} \right) \vec{d} \right\rangle \; ,$$

where $\frac{1}{\langle \mu^2 \rangle}$ represents the Lagrange multiplier $\xi$.

Hence, the MFA and the KvD algorithm are identical for one iteration step.

## 2.3.5 The bias of the KvD algorithm

The equivalence of the KvD algorithm and the MFA for known distances follows directly from the Gauss-Markov theorem [20], which states that an ordinary least-squares estimator is the best unbiased estimator for an estimation problem which is linear and has uncorrelated errors with equal variances. Both methods start with such a least-squares approach. The MFA minimizes the quadratic error between the six true self-motion values and the estimated values as can be seen in equation (2.5), whereas the KvD algorithm minimizes the quadratic error between the measured optic flow and the theoretical optic flow as can be seen in equation (2.8). For known distances, the optic flow and the self-motion values are connected through a linear transformation. Thus, the two least-square approaches lead to the same self-motion estimator. This estimator is the unique optimal estimator as stated by the Gauss-Markov theorem.

The situation is different if the distances are not known and have to be estimated by the estimator together with the self-motion values. Then the problem is no longer linear and the Gauss-Markov theorem does not hold. Nonetheless, it seems likely that a least-square approach like in equation (2.8) leads to an optimal estimator in the sense that the error in the estimated self-motion components approaches zero with increasing number of measured optical flow values. However, as will be shown in this section, the KvD algorithm in general does not converge to the true self-motion values for an infinite number of flow vectors. It is still an open question from which minimization principle an optimal estimator can be derived. The increasing number of flow vectors raises the problem that for each flow vector which gives two additional error-prone values one additional value has to be estimated: the inverse distance in the respective viewing direction. Although the number of measured values increases towards infinity, the ratio between the number of estimated and measured values does not decrease to zero. Hence, even for an infinite number of flow vectors the estimated inverse distances are still afflicted with errors. However, it should still be possible to correctly estimate the fixed number of self-motion values for an infinite number of flow vectors. In section 2.3.5 a modified KvD algorithm will be derived. The modification is tested numerically under two conditions where the original KvD algorithm turns out to be biased (section 2.3.5).

### The non-vanishing error term

The KvD algorithm is an unbiased estimator only under certain conditions. To show this, the propagation of the error in the flow vectors $\vec{p}_i$ over the iterations will be analyzed. We model the measured flow vectors $\vec{p}_i$ as the sum of the true vector $\vec{p}_i^0$ and a random error vector $\Delta\vec{p}_i$, $\vec{p}_i = \vec{p}_i^0 + \Delta\vec{p}_i$. Similar to vector $\vec{p}_i$, the vectors $\vec{p}_i^0$ and $\Delta\vec{p}_i$ have only two degrees of freedom. It will be assumed that the random vectors $\Delta\vec{p}_i$ are unbiased, i.e. the expectation values for all directions $i$ are zero, $E(\Delta\vec{p}_i) = 0$.

Two special conditions will be considered in the following:

1. The viewing directions $\vec{d}_i$ are equally distributed over the sphere.

2. The random vectors $\Delta\vec{p}_i$ are uncorrelated and their variances constant, independent of the directions $i$, $\mathrm{var}(\Delta\vec{p}_i) = E(\Delta\vec{p}_i \cdot \Delta\vec{p}_i) = \text{constant}$.

The Gauss-Markov theorem assumes condition (2) to be fulfilled. We will also consider deviations from this condition, because the KvD algorithm and its modified version that will be derived in section 2.3.5 behave differently then. Condition (2) is violated if, for example, the error of the optic flow measurement depends on the length of the measured optic flow vectors.

In subsection 2.6.2 of the appendix it is shown that, in general, the translation estimated by the KvD algorithm contains errors that do not vanish even for an increasing number of flow vectors. There are two error terms which are additive to the real translation $\vec{t}^0$.

$$
\begin{aligned}
\vec{t} &= \vec{t}^0 + \Delta\vec{t} = \vec{t}^0 + a + b \\
a &\propto \left\langle \frac{\Delta\vec{p} \otimes \Delta\vec{p}}{1 - (\vec{t}^0 \cdot \vec{d})^2} \right\rangle_\infty \cdot \vec{t}^0 \\
b &\propto \left\langle \frac{\vec{t}^0 \cdot (\Delta\vec{p} \otimes \Delta\vec{p}) \cdot \vec{t}^0}{(1 - (\vec{t}^0 \cdot \vec{d})^2)^2}(\vec{d} \otimes \vec{d}) \right\rangle_\infty \cdot \vec{t}^0
\end{aligned}
$$

The index $\infty$ of the brackets $\langle \rangle$ stands for the limit of an infinite number of flow vectors. The discrete direction vectors $\vec{d}_i$ can be exchanged by their continuous counterparts $\vec{d}_{\vartheta\varphi}$, and the sum over $i$ can be replaced by an integral over the field of view.

The iteration equation for $\vec{r}$, equation (2.12), does not contain terms that lead to a bias in the estimated motion values. However, the estimated rotation will be affected indirectly by errors in the estimated translation.

By disregarding the numerators, the integrals containing the denominators $D_1 = 1 - \left(\vec{t}^0 \cdot \vec{d}\right)^2$ and $D_2 = \left(1 - \left(\vec{t}^0 \cdot \vec{d}\right)^2\right)^2$ are analyzed. The integrals over $\frac{1}{D_1}$ and $\frac{1}{D_2}$ are zero only if the first condition of equally distributed flow vectors is fulfilled. To avoid the singularity for $\vec{d} = \vec{t}^0$ a small constant $\varepsilon$ was added to the denominators.

If conditions (1) and (2) are fulfilled, the terms $a$ and $b$ are zero and the KvD algorithm is an unbiased estimator.

If condition (1) holds but condition (2) does not as, for instance, in the case of realistic EMDs or gradient-based detectors, we have to integrate over a direction dependent function resulting from the direction dependent flow errors $\Delta\vec{p}$. Hence the terms $a$ and $b$ converge to finite values.

The error terms $a$ and $b$ do not play a role if they are proportional to the identity matrix, because of the rescaling of the translation vector, which ensures $\|\vec{t}\| = 1$. The matrices $E_1 = \langle \Delta\vec{p} \otimes \Delta\vec{p} \rangle$ and $E_2 = \left\langle (\Delta\vec{p} \otimes \Delta\vec{p}) \cdot (\vec{d} \otimes \vec{d}) \right\rangle$ are proportional to the unit

matrix, if and only if both conditions (1) and (2) are satisfied. This can be shown by taking into account the symmetry of viewing directions and the constant variances of the flow errors.

Most interestingly, if condition (2) is fulfilled (a pre-condition of the Gauss-Markov theorem) but condition (1) is not, the terms $a$ and $b$ converge to finite values. In this case the integrals over the denominators $D_1$, $D_2$ and the integrals over the numerators $E_1$, $E_2$ have finite values. This means that the ordinary least-squares approach from equation (2.8) leads to a biased self-motion estimator.

**Modification of the KvD iteration equations**

To improve the KvD algorithm a modified version of the iteration equation (2.11) will be derived. The flow equation (2.1) can be transformed:

$$\vec{p}_i = -\mu_i \left( \vec{t} - \left( \vec{t} \cdot \vec{d}_i \right) \vec{d}_i \right) - \vec{r} \times \vec{d}_i \qquad (2.29)$$

$$\mu_i \vec{t} = -\vec{p}_i - \vec{r} \times \vec{d}_i + \mu_i \left( \vec{t} \cdot \vec{d}_i \right) \vec{d}_i \quad . \qquad (2.30)$$

By taking the average and solving for $\vec{t}$ we obtain

$$\langle \mu \vec{t} \rangle = -\langle \vec{p} \rangle - \left\langle \vec{r} \times \vec{d} \right\rangle + \left\langle \mu \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle \quad ,$$

$$\vec{t} = \langle \mu \rangle^{-1} \left\{ -\langle \vec{p} \rangle - \left\langle \vec{r} \times \vec{d} \right\rangle + \left\langle \mu \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle \right\} \quad ,$$

$$\vec{t} = -\xi \left\{ \langle \vec{p} \rangle + \vec{r} \times \left\langle \vec{d} \right\rangle - \left\langle \mu \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle \right\} \quad , \qquad (2.31)$$

where $\xi$ ensures that $\vec{t}$ is normalized. Compared with the original equation for the translation (2.11) the additional factor $\mu_i$ is absent. Nonetheless, the above equation for the translation depends still on the distances. An analog derivation leads to the same iteration equation for the rotation as in the original KvD algorithm.

If the flow vectors have no errors the modified version converges, as does the original algorithm, to the true self-motion parameters. In contrast to the result for the original KvD algorithm in appendix 2.6.2, the iteration equations of the modified version do not contain $\mu \vec{p}$ or $\mu^2$ and the true self-motion values are fix points of the iteration (only when the true values are fix points in the iteration the algorithm can converge to these values).

**Numerical tests of the original and modified KvD algorithm**

In Fig. 2.1 the bias of the KvD algorithm is shown numerically (see section 'Material and Methods', 2.5.3 for a detailed description of the numerical test). The left part of the figure shows simulation results for flow vectors with added errors of equal variance. The field of view given by the viewing directions $\vec{d}_i$ is non-equally distributed: The flow
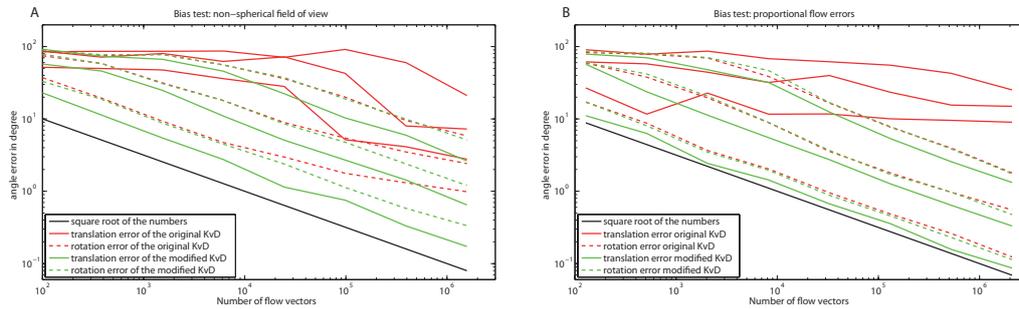
**Figure 2.1:** The averaged angle error, $\arccos\left(\dfrac{\vec{\theta}_N^{\text{est}} \cdot \vec{\theta}}{\left|\vec{\theta}_N^{\text{est}}\right| \cdot \left|\vec{\theta}\right|}\right)$, between the estimation $\vec{\theta}_N^{\text{est}}$, which depends on the number of flow vectors $N$, and the true values $\vec{\theta}$, is shown for translation (solid lines) and rotation (dashed lines). Red curves show the errors for original KvD algorithm, green curves for the modified KvD algorithm. The errors are averaged over 40 trials (see methods 2.5.3). For each trial the true self-motion parameters are chosen randomly, equally distributed over the sphere, in such a way, that the resulting translational flow equals the resulting rotational flow in magnitude. The distances, also determined randomly, lie with equal probability between one and three in arbitrary units. The results for three different variances $(\triangle\vec{p})^2$ are shown (from bottom to top): 1, 3, and 9 times the flow vector length, where the factor is interpreted differently in the two graphs. **A)** Results for non-equally distributed flow vectors with equal variance of the flow vector errors $\triangle\vec{p}_i$ (the variance is matched to the mean flow vector). **B)** Results for equally distributed flow vectors, where the variance of the errors $\triangle\vec{p}_i$ depends linearly on the length of the $\vec{p}_i$ (the variance is matched to the local flow vector).

vectors are equally distributed except for two regions of the sphere which do not contain any flow vectors. The two regions are quarters of the half-sphere which lie opposite to each other in the upper half-sphere. Thus, the simulation result provides an example where condition (2) of section 2.3.5 is fulfilled but condition (1) is not. The translation error of the original KvD algorithm is significantly larger and increasingly deviates from that of the modified version with increasing number of flow vectors. Due to the coupling of the iteration equations, the error of the rotation, in case of the original KvD algorithm, is affected by the translation error and, thus, also deviates from that of the modified version.

The right part of the figure shows results where the standard deviation of $\triangle\vec{p}_i$ is proportional to the length of the local flow vector $\vec{p}_i$. This time, the viewing directions cover the whole sphere homogeneously and, thus, condition (1) of section 2.3.5 holds while condition (2) does not. Again, the original KvD algorithm is biased (see also section 2.3.5). However, the error of the rotation in the original KvD algorithm is not influenced

by the translation error as a consequence of the spherically distributed viewing directions.

The error of the modified KvD algorithm is in both analyzed cases inversely proportional to the square root of the number of flow vectors (see black line in Fig. 2.1). The modified KvD algorithm shows therefore an error behavior as is characteristic of an unbiased linear estimator.

### 2.3.6  An adaptive MFA

The approach of Franz and Krapp [33] is based on the assumption that the statistics of the depth structure of the environment is fixed as well as the preferred translation directions of the agent are known. For the simplest statistical model, which assumes that the distance variability, the noise in the flow measurements and the preferred translation of the agent are independent from the viewing direction, the dedicated covariance matrices are proportional to the identity matrix. Most important, one has to specify the depth structure of the environment by defining the average inverse distances $\langle \mu \rangle_i$ or, as in [33], the average distances $\langle D \rangle_i$. Franz and Krapp [33] modeled the distances $\langle D \rangle_i$ by

$$\langle D \rangle_i = \begin{cases} D_0 & \varepsilon_i \geq 0 \ , \\ \frac{\beta D_0}{\sqrt{1 + (\beta^2 - 1)\cos^2(\varepsilon_i)}} & \varepsilon_i < 0 \ , \end{cases} \tag{2.32}$$

where $D_0$ denotes a typical distance in the upper hemisphere, $\varepsilon_i$ the elevation angle of viewing direction $i$ and $\beta = \frac{h}{D_0} < 1$ the ratio of the average flight altitude $h$ and $D_0$. This model takes into account that the distances are usually smaller for viewing directions below the horizon.

It is obvious that the performance of self-motion estimation is poor when the fixed depth model (2.32) is not a good description of the depth structure at the current position of the agent. Therefore, we propose an adaptive MFA, which allows the agent to adapt its depth model to the current environment. This is only possible, if we can assume that the depth structure properties of the environment do not change abruptly from one time step to the next. A time constant describes the intervals in which the depth model will be updated. Between updates the depth model remains fixed and has to be memorized by the agent. The depth information is obtained from the optic flow, as well as the self-motion values. Since the original KvD algorithm is biased in this case, we use our modified KvD version as the starting point for deriving the adaptive model.

As will be shown in the following, it is not necessary to represent the full depth information, because it is sufficient to memorize just eight distance dependent parameters. Three parameters are contained in the distance-dependent term of the rotation equation (2.12), $\vec{t} \times \left\langle \mu \vec{d} \right\rangle$ or $(t_1, t_2, t_3)^T \times (\mu \sin(\vartheta)\cos(\varphi), \mu \sin(\vartheta)\sin(\varphi), \mu \cos(\vartheta))^T$. In the translation equation of the modified KvD algorithm (2.31) only $\left\langle \mu \vec{d} \otimes \vec{d} \right\rangle \cdot \vec{t}$ depends

on $\mu$. As the matrix $\left\langle \mu \vec{d} \otimes \vec{d} \right\rangle$ is symmetric it contains only six different $\mu$-dependent elements.

The first three orders of spherical harmonic functions, the zeroth, first and second order, comprise nine parameters, but only eight parameters can be determined from the optic flow. The zeroth order remains undefined, because one cannot distinguish between the situation, where all objects have half the distance to the agent, and the situation, where the agent translates with double speed: The optical flow will be the same. Hence, from optic flow fields only the direction of the translation can be identified.

Between updates of the depth model, rotations, in particular, impair its validity, but can be easily compensated by counter-rotating the depth model. This is achieved by multiplying the depth dependent coupling matrix with a rotation matrix obtained from the rotation parameters of the current self-motion estimate.

The adaptive model will be derived first for the general case with an arbitrary field of view. Then a more biologically plausible adaptive model with a spherical field of view will be presented. A spherical field of view facilitates an intuitive interpretation of the depth model. In this specific case, the nine depth-dependent parameters are exactly the coefficients of the first three orders of the spherical harmonics expansion, i.e the dipole and quadrupole moments of the depth structure $\mu_{\vartheta\varphi}$.

### Motivation for an adaptive MFA

In Fig. 2.2 the initialization phase of the adaptive model is shown. The agent flies inside a sphere in such a way, that the depth model is the same for every trajectory step (see Fig. 2.2, Fig. 2.4B and methods 2.5.4 for details). The error in the estimated self-motion parameters decreases exponentially. Hence, the error is corrected to a large extent in the first few iteration steps.

In natural environments one can detect subspaces where the distances to a moving agent do not change over a certain time [80]. Hence, one can assume that the overall depth model changes only slightly from one step to the next in a given environment and can expect good self-motion estimates even for only a single iteration step of the KvD algorithm based on the old depth model. Furthermore, the depth model can be used even for a longer time interval. In this case the depth model is not updated instantaneously after receiving new optic flow information, but less frequently after several optic flow processing steps. Because the self-motion parameters and the depth model are formulated in the body coordinate system of the agent, the depth model has to be rotated together with the agent.

### Matched filters and depth-dependent coupling matrix for the adaptive MFA

The equations of the modified KvD algorithm form the basis of the adaptive MFA (a small constant $\varepsilon$ is inserted in (2.33), the equation of the inverse distance, to avoid the
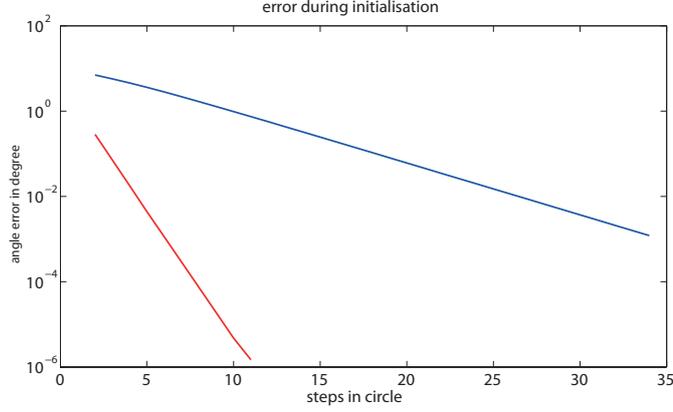
**Figure 2.2:** An agent flies inside the sphere on a circular trajectory (see Fig. 2.4A). The center of the circle does not coincide with the center of the sphere to avoid symmetries in the depth model. The correct depth model is constant in this configuration, only the initialization of the depth model is tested. The y-axis shows the angle error $\arccos\left(\frac{\vec{\theta}^{\text{est}}\cdot\vec{\theta}}{\left|\vec{\theta}^{\text{est}}\right|\cdot\left|\vec{\theta}\right|}\right)$, between the estimated self-motion axis of $\vec{\theta}^{\text{est}}$ and the axis of the true self-motion values $\vec{\theta}$. The depth model is initialized with constant distances. With every update step of the depth model the error decreases exponentially for the translation (blue) as well as for the rotation (red).

singularity in case $\vec{d}_i = \vec{t}$):

$$\mu_i = -\frac{\vec{t}\cdot\left(\vec{p}_i - \vec{d}_i \times \vec{r}\right)}{1 - \left(\vec{t}\cdot\vec{d}_i\right)^2 + \varepsilon} \ , \tag{2.33}$$

$$\vec{t} = \frac{-1}{\langle\mu\rangle}\left(\langle\vec{p}\rangle + \vec{r}\times\left\langle\vec{d}\right\rangle - \left\langle\mu\left(\vec{t}\cdot\vec{d}\right)\vec{d}\right\rangle\right) \ , \tag{2.34}$$

$$\vec{r} = \left\langle\vec{p}\times\vec{d}\right\rangle + \vec{t}\times\left\langle\mu\vec{d}\right\rangle + \left\langle\left(\vec{r}\cdot\vec{d}\right)\vec{d}\right\rangle \ . \tag{2.35}$$

As shown in section 2.3.4 these iteration equations can be decomposed into the product of the current optical flow and a standard template, on the one side, and a coupling matrix, on the other side (see following equation). The coupling matrix is the part of the adaptive model that contains the depth values $\mu_i$:

$$\begin{pmatrix} M^{tt} & M^{tr} \\ M^{rt} & M^{rr} \end{pmatrix}\cdot\begin{pmatrix} \vec{t} \\ \vec{r} \end{pmatrix} = \begin{pmatrix} \left\langle-\vec{p}\right\rangle \\ \left\langle\vec{p}\times\vec{d}\right\rangle \end{pmatrix} \ , \tag{2.36}$$
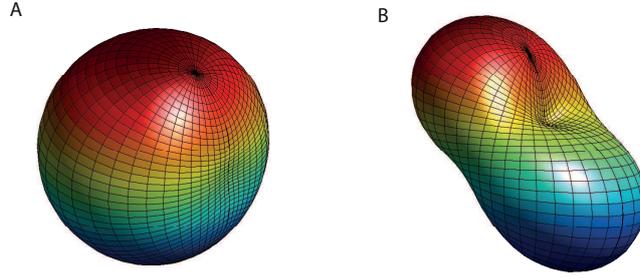
**Figure 2.3:** Spherical harmonic functions from the expansion of the inverse distances $\mu_i$. **A)** The sum of the zeroth order function and a first-order dipole-function. **B)** The sum of the zeroth order function and a second-order quadrupole-function.

$$M^{tt} = \langle\mu\rangle I - \left\langle \mu\vec{d}\otimes\vec{d} \right\rangle , \qquad (2.37)$$

$$M^{tr} = -\left\langle [\vec{d}\times] \right\rangle , \qquad (2.38)$$

$$M^{rt} = \left\langle \mu[\vec{d}\times] \right\rangle , \qquad (2.39)$$

$$M^{rr} = I - \left\langle \vec{d}\otimes\vec{d} \right\rangle , \qquad (2.40)$$

where the matrix $[\vec{d}\times]$ is defined by

$$[\vec{d}\times]\vec{v} = \vec{d}\times\vec{v} ,$$

$$[\vec{d}\times] = \begin{pmatrix} 0 & -d_3 & d_2 \\ d_3 & 0 & -d_1 \\ -d_2 & d_1 & 0 \end{pmatrix} ,$$

i.e. the cross product of the two vectors $\vec{d}$ and $\vec{v}$ can be expressed as multiplication of matrix $[\vec{d}\times]$ and vector $\vec{v}$.

### The case of spherically distributed flow vectors

For the special case of spherically distributed flow vectors the depth-dependent coupling matrix (see subsection (2.3.3) is given explicitly. In the simplest case of an agent being in the center of a sphere, i.e. if all inverse distances have the same value, the depth-dependent coupling matrix is proportional to the identity matrix (see end of subsection (2.3.3). In the general case, i.e. when the inverse distances can have arbitrary values, the entries of the depth-dependent coupling matrix are the expansion coefficients of the first three orders of the real valued spherical harmonic functions.

Then the environmental depth structure $\mu_i$ can be described by a real-valued function $\mu_{\vartheta\varphi}$ parameterized by the azimuth angle $\varphi$ and the elevation angle $\vartheta$ (in a spherical coordinate system). Such functions can be described by an expansion of real-valued spherical harmonic functions [74] (see appendix 2.6.3 and Fig. 2.3). Lower orders of these functions contain less details than higher order functions.

Given a function $\mu(\vartheta, \varphi)$ depending on azimuth angle $\varphi$ and elevation angle $\vartheta$ the expansion coefficients $a_{ln}$ are

$$a_{ln} = \int_{\varphi=0}^{2\pi} \int_{\vartheta=0}^{\pi} R_{ln}(\vartheta, \varphi) \cdot \mu(\vartheta, \varphi) \cdot \sin\vartheta d\vartheta d\varphi . \tag{2.41}$$

$R_{ln}(\vartheta, \varphi)$ represent, for example, a dipole function for specific $l$ and $n$. The corresponding coefficient $a_{ln}$ provides information about how pronounced the specific dipole part is in the expanded function $\mu(\vartheta, \varphi)$.

With the help of all coefficients $a_{ln}$ the expanded function $\mu(\vartheta, \varphi)$ is given by the reverse transformation

$$\mu(\vartheta, \varphi) = \sum_{l=0}^{N} \sum_{n=-l}^{+l} a_{ln} R_{ln}(\vartheta, \varphi) , \tag{2.42}$$

with $N = 0, 1, 2, \ldots$.

The terms $\langle \mu \rangle$ and $\left\langle \mu \vec{d} \otimes \vec{d} \right\rangle$ in $M^{tt}$ and the term $\left\langle \mu[\vec{d}\times] \right\rangle$ in $M^{rt}$ are the sole terms which contain the inverse distances $\mu_i$. In appendix (2.6.3) it is shown, that $\vec{d} \otimes \vec{d}$ and $\vec{d}$ can be expressed through real valued spherical harmonic functions when these term are given by their continuous counterparts $\vec{d}_{\vartheta\varphi}$. It can be directly seen that the continuous counterparts of $\vec{d}$ are the first-order real valued harmonic functions (except for a constant factor), whereas some transformations are needed to see that the continuous counterparts of $\vec{d} \otimes \vec{d}$ are linear combinations of the zeroth and second order real valued harmonic functions.

Due to the linearity of an integral expression and the orthogonality of the spherical harmonic functions [74],

$$\int R_{ln}(\vartheta, \varphi) R_{l'n'}(\vartheta, \varphi) d\Omega = \delta_{ll'} \delta_{nn'} , \tag{2.43}$$

the terms $\langle \mu \rangle$, $\left\langle \mu \vec{d} \otimes \vec{d} \right\rangle$ and $\left\langle \mu[\vec{d}\times] \right\rangle$ can be interpreted as the definition equations for specific coefficients of a spherical harmonic expansion of $\mu(\vartheta, \varphi)$ [see equation (2.41)]. Hence, in the spherical case $\langle \mu \rangle$, $\left\langle \mu \vec{d} \otimes \vec{d} \right\rangle$ and $\left\langle \mu[\vec{d}\times] \right\rangle$ in the coupling matrix (equation (2.38) until (2.40)) can be replaced by the coefficients of spherical harmonic functions:

$$
\begin{aligned}
M^{tt} &= \sqrt{4\pi} \cdot a \cdot \left( I - \frac{1}{3}I \right) - \sqrt{\frac{4\pi}{15}} \cdot C \\[2mm]
C &= \begin{pmatrix}
-c_4 - \sqrt{\frac{1}{3}} \cdot c_1 & c_5 & c_2 \\
c_5 & c_4 - \sqrt{\frac{1}{3}} \cdot c_1 & c_3 \\
c_2 & c_3 & \sqrt{\frac{4}{3}} \cdot c_1
\end{pmatrix} \\[2mm]
M^{tr} &= 0 \\[2mm]
M^{rt} &= \sqrt{\frac{4\pi}{3}} \cdot B \\[2mm]
B &= \begin{pmatrix}
0 & -b_3 & b_2 \\
b_3 & 0 & -b_1 \\
-b_2 & b_1 & 0
\end{pmatrix} \\[2mm]
M^{rr} &= \frac{2}{3}I
\end{aligned}
$$

where $a$ is the coefficient of the zeroth spherical harmonic function, which cannot be determined by the algorithm as explained earlier. The three $b$'s are the three coefficients of the first-order spherical harmonic functions, the dipole functions. The five $c$'s are the five coefficients of the second-order spherical harmonic functions, the quadrupole functions.

The only non-constant parameters in the depth-dependent matrix are the nine coefficients of the expansion by spherical harmonics. The nine parameters the agent has to memorize during flight have, in the case regarded here, a physical interpretation: They are the dipole and quadrupole parts of the depth distribution of the environment. The non-existence of higher-order coefficients in the depth-dependent matrix indicates that these orders contain no information for solving the self-motion problem. If the distances are constant, the matrix $M$ is the identity matrix (except for a normalization constant) as mentioned before.

The computation of the nine coefficients might not seem to be biologically plausible at first sight. However, the computation of one coefficient $a_{ln}$ of the expansion corresponds to a weighted wide-field integration and is reminiscent of the function of LPTCs in flies [9, 27, 30, 59]. One could imagine a neuron for each of the nine parameters $a_{ln}$, which represents a specific global property of the depth structure.

With regard to the computational effort a full inversion of a $6 \times 6$ matrix is not required. The submatrix $M^{tr}$ is zero in the spherical case, hence only an inversion of the submatrix $M^{tt}$ is required. If the quadrupoles in this submatrix are sufficiently small, the inverse matrix can be linearly approximated by a Neumann series [103], $(I - A)^{-1} = I + A + A^2 + A^3 + \ldots.$
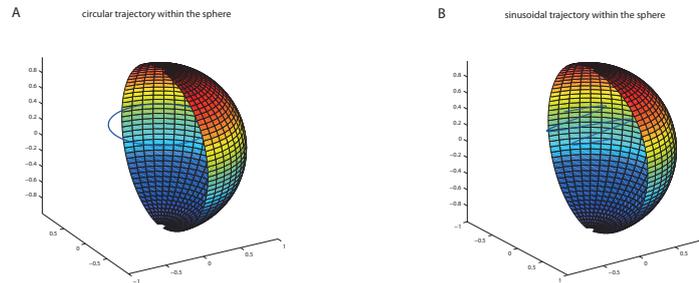
**Figure 2.4: A)** shows a circular trajectory to analyze the initialization phase of the adaptive MFA. The height of the trajectory lies above the middle point of the sphere to avoid trivial depth models. Due to symmetry the depth model for this configuration is the same at every trajectory point. **B)** shows a sinusoidal trajectory. It is used to analyze the self-motion estimation error during adaptation. Again the height of the trajectory is lifted up against the middle point of the sphere to make the depth model more complex in relation to an agent, which flies along the trajectory.

**Test of the adaptive MFA**

In this section we compare quantitatively the adaptive and the original MFA. We present a simulation in a very simple environment where the agent moves inside a sphere (see Fig. 2.4 and methods 2.5.4). Nothing is known in advance about the flight directions and whereabouts of the agent inside the sphere. Hence, the covariance matrices of the original MFA are set proportional to the unit matrix. The chosen trajectory in this setting is a sinusoidal curve. On this trajectory the current depth distribution differs essentially from that in the center of the sphere (Fig. 2.4A).

The amount of translation and rotation in each step varies along the trajectory. The steps are chosen so that the maximum rotation angle is about 8 degrees per step ensuring that the approximation of the KvD algorithm, which is valid for small rotation angles, still holds. The maximum rotation angles between the processed images is in the same range as the maximum rotation angles during saccades of flying insects [81, 45].

On the whole, the adaptive MFA performs better than the original non-adaptive MFA (Fig. 2.5). The second y-axes on the right side of the figure show the averaged ratio between the rotational and translational optic flow at every time step. Due to the sinusoidal trajectory of the agent this ratio varies between a factor of hundred in favor of the rotational or translational optic flow. When an optic flow component is overlayed hundredfold by the other flow component the coupling terms $\vec{r} \times \left\langle \vec{d} \right\rangle$ and $\vec{t} \times \left\langle \mu\vec{d} \right\rangle$ between the flow components are the dominant parts in the related estimation equations (2.34) and (2.35). For a spherical field of view the term $\vec{r} \times \left\langle \vec{d} \right\rangle$ is zero. But the term
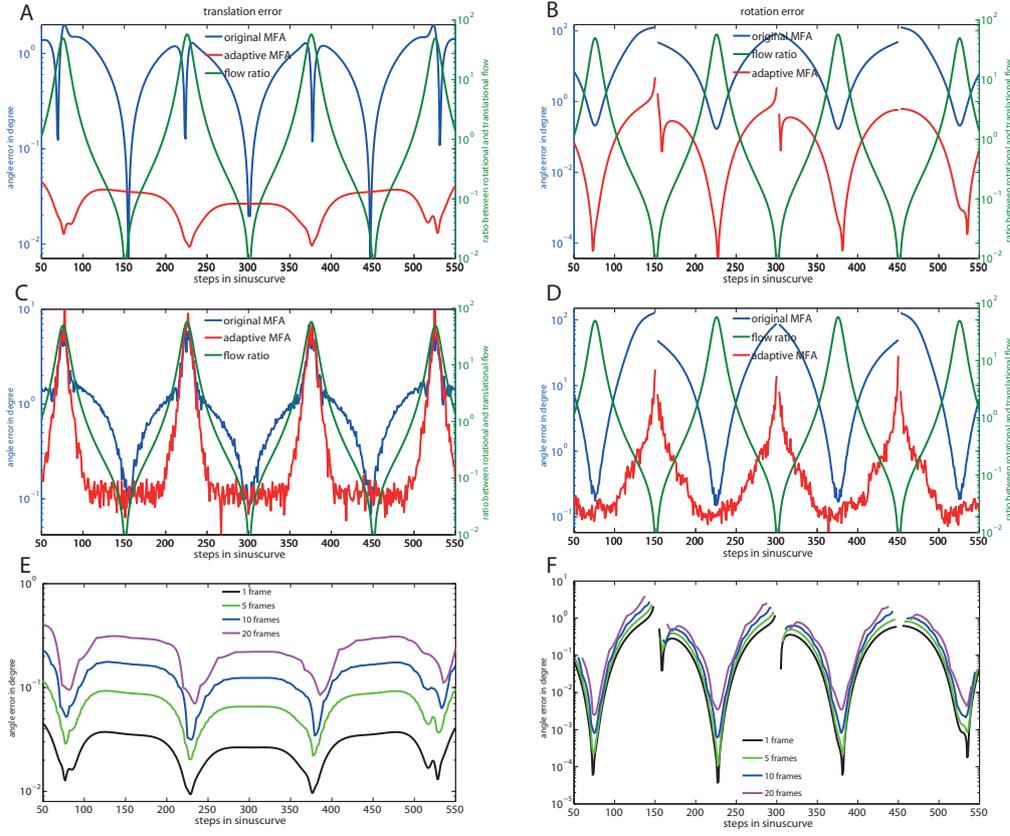
**Figure 2.5:** As in Fig. 2.2 an agent flies inside a sphere (see methods 2.5.4). The trajectory is a sinus curve with two full periods. The amplitude of the sinus curve is 0.5 of the radius of the sphere. The sinus curve is lifted in the sphere by 0.3 units in z-direction to avoid symmetries in the depth model. The agent performs 600 steps which result in rotation angles of up to 8 degrees per frame. The initialization phase is not shown. The left figures shows the error of the translation estimates and the right figures shows the error of the rotation estimates. The y-axes show the angle error $\arccos \left( \frac{\bar{\theta}^{\text{est}} \cdot \vec{\theta}}{\left| \bar{\theta}^{\text{est}} \right| \cdot |\vec{\theta}|} \right)$, between the estimated self-motion axis of $\bar{\theta}^{\text{est}}$ and the axis of the true self-motion values $\vec{\theta}$. The estimated angle errors have a pole, when the rotation gets zero at the inflection points of the sinusoidal curve (see methods 2.5.4). The small region around the poles are cut out in the figures. **A,B)** The two figures show the error of the adaptive MFA (red curve) and the original MFA (blue curve) with a constant inverse distant assumption for the original MFA. The adaptive MFA is updated every time step. The right y-axes of the figures show the averaged ratio between the rotational and translational optic flow. **C,D)** shows the adaptive MFA and the original MFA as in figures A and B, but with an error of 10% added to the optical flow. **E,F)** show different update frequencies of the depth model. All models rotate with the agent. The update frequencies are: black = 1 frame, green = 5 frames, blue = 10 frames and violet = 20 frames.

$\vec{t} \times \left\langle \mu \vec{d} \right\rangle$ depends on the dipole components of the inverse distance $\mu$ (see appendix 2.6.3). Hence, only the rotation estimate is affected by errors in the estimated dipoles. Because the original MFA does not determine the values for the dipole components of the inverse distance for the current optical flow, the rotation estimates become totally useless, whenever the translational flow component is dominant. The angle error between the estimated rotation axis and the true rotation axis increases to a value of about hundred degrees. Whereas the angle error of the adaptive MFA does not exceed few degrees.

If an Gaussian error is added to the optical flow with a standard deviation of ten percent of the averaged overall flow value (Fig. 2.5), one could expect that a translational or rotational flow component, which is overlayed hundredfold by the other flow component disappears totally in this flow error, because the error is tenfold higher as the flow component in this situation. But the estimators use hundreds or a few thousands of flow vectors to estimate self-motion. Due to the large number of flow vectors (insects usually have between a few hundred (e.g. fruit fly) and a few thousand ommatidia (bee, dragonfly) per eye), the self-motion can be still estimated in this case within a useful error range. The results are shown for about 5000 flow vectors (Fig. 2.5). The error increases for both the translational and the rotational self-motion estimate to a value of about 10 degrees. This error is additive to the error described in the upper panels of Fig. 2.5 and affects the adaptive MFA in the same way as the original MFA.

In the bottom panels of Fig. 2.5 different update rates are tested. Even for an update at only every twentieth optical flow processing step the errors remain in a useful range.

Albeit the simplicity of the simulation it shows some basic features of the compared algorithms. The simulation does not generate any outliers due to moving objects or depth discontinuities. A small number of outliers will not affect the MFA as a consequence of the linear summation over thousands of optic flow vectors. More complex simulations in virtual environments with rendered images and EMDs is left to future work (Strübbe et al., in prep.).

## 2.4 Discussion

The aim of this study is to develop an adaptive matched filter approach to self-motion estimation which could be in principle the underlying concept of self-motion estimation in flying insects. As a novel characteristic, this approach assumes an adaptation to the depth structure in the insect visual motion pathway, an assumption that is supported by recent experimental evidence [67, 68, 99]. Our approach starts from a theoretical point of view by analysing and unifying the non-adaptive matched filter approach (MFA) to self-motion estimation together with the Koenderink van Doorn (KVD) algorithm which incorporates an estimation of the depth values.

To take advantage of both algorithms, some mathematical problems had to be solved. First, it was shown that the two algorithms are equivalent in case the distances to objects

in the environment are assumed to be known. Secondly, a bias in the KvD algorithm was removed by a small correction of the iteration equations. And last but not least, an analysis of the specific case of a spherical field of view, reminiscent of that of flying insects, shows that the depth structure can be represented by only eight parameters without losing relevant information for self-motion estimation and that these eight parameters are the dipole and quadrupole moments of the spherical harmonics.

Technical and biological systems have different origins and often operate under different conditions. Biological systems arise through evolutionary adaptation. They usually have to operate in a great variety of environments. Hence, the neural computations underlying the animal's behavior need to be particularly robust. In addition, the animal has restrictions with respect to its computational hardware. Neuronal circuits can perform linear transformations in parallel by a dendritic weighted summation of the inputs of a neuron and non-linear operations through the non-linear response behavior of a neuron to its overall input. Nonetheless, a non-linear operation, such as computing the inverse of a matrix, with changing entries, is not easy to implement by neuronal hardware. The bio-inspired computational model analyzed here is the MFA of self-motion estimation. It is a linear model for a fixed depth assumption, derived under the side condition of maximal robustness against errors in the measured optical flow field (see equation (2.5) from Franz et al. [35]).

The KvD algorithm of self-motion estimation which is compared with the MFA model was derived analytically in a technical framework on the basis of a minimization principle. The resulting iteration equations represent a gradient descent where the current self-motion parameters are used to determine a better depth model and the new depth model is used to determine a better estimate of self-motion in the next iteration step. If one considers only one iteration step, where the depth model is seen as fixed, the self-motion estimation is linear. It is not only linear, but also equivalent to the biologically inspired MFA which uses a fixed depth distribution.

The equivalence of both models becomes evident within the framework of linear estimator theory. There exists an unique optimal estimator for a linear estimation problem with error-prone inputs. The Gauss-Markov theorem describes this estimator. Both compared methods represent this optimal solution.

However, some differences exists. In the MFA Franz et al. [35] weighted the filters by matrices that represent additional assumptions about the situation under which the self-motion is estimated. If these assumptions are correct, the weighting of the filters improves self-motion estimation; however, if these assumptions are incorrect in the current situation, the estimator gets worse. Hence, the additional matrices make the estimator more specific. These matrices can also be implemented in the KvD algorithm by a modification of the minimization principle. When, for example, it is known, that the optical flow can be measured more accurately below the horizon, because the objects are generally closer there, this knowledge can be taken into account by introducing weights in the initial equation. We argue that it is not always useful to take knowledge about

the preferred self-motion directions into account. Even when the moving agent solely translates in the forward direction a disturbance can lead to a passive translation also in other directions.

From a mathematical point of view the bias of the KvD algorithm is remarkable. When the depth distribution of the environment has to be determined together with the self-motion parameters, the estimation problem is no longer linear. The standard procedure for estimating parameters from inputs, which are disturbed by Gaussian errors, is the minimization of the mean squared error. It might be counter-intuitive that the true self-motion values are not even a local minimum. The standard approach fails, because the standard condition assumes that an increasing number of measured values, here the flow vectors, are accompanied by a fixed number of estimation parameters. However, for the non-linear estimation problem the number of distance values increases together with the number of flow vectors. Only the number of self-motion parameters remains constant. With every additional flow vector additional information about the self-motion parameters is obtained, because one gets two additional independent values from the flow vector, but only one additional parameter has to be estimated (the distance corresponding to the flow vector). However, the standard approach does not use this additional information in an optimal way.

Here we derived a modified version of the KvD algorithm that is not derived from a minimization principle. Hence it is not clear whether it leads to the best estimate for a given finite number of flow vectors. Rather, the numerical simulation indicates that the modified version has the desired property that the algorithm converges to the real self-motion values for an infinite number of flow vectors. It is left to further mathematical work to analyze optimal criteria for the non-linear estimation problem in case of a finite number of flow vectors.

Based on this modified version of the KvD algorithm an adaptive MFA was derived. It was shown that it is a critical issue to correctly determine the dipole components. If a small rotation is superimposed by a large translation the non-adaptive MFA cannot provide useful rotation estimates, whereas the adaptive MFA is accurate up to a few degrees. A situation where a relatively large translation encounters a relatively small rotation is given in the inter-saccadic phases of insect flight [81, 45, 7]. In these phases the insect tries to avoid any rotation. If the insect stabilizes its flight with the help of the visual system, the non-adaptive MFA cannot be the underlying concept to detect small rotations in these phases in environments it is not tuned to. To estimate rotations, which are superimposed by a large translation, one has to determine the current dipole components, as is done by the adaptive MFA.

The adaptive MFA was inspired by the finding that for a spherical field of view the depth structure of the environment can be represented by only eight parameters without losing relevant information for self-motion estimation and by the fact that the visual system of insects has an almost spherical field of view. The spherical field of view is also a desirable property of technical systems which are designed to estimate their self-motion

on the basis of optical flow fields. Such systems can be realized by panoramic cameras [91, 92].

Adaptation to the depth structure of the environment means that the adaptation takes place on another time scale than the image processing itself. Hence, some information about the depth structure has to be memorized by the system. The result that exact self-motion can only be estimated for a spherical field of view, if eight parameters about the depth structure of the environment are known, is therefore in accordance with the limited computational resources of insects.

Motion adaptation was analyzed in the insect visual pathway and found to depend on the overall velocity in the visual field [71, 67, 62, 68, 61, 99]. Since, at least during translational motion, the overall retinal velocity depends on the depth distribution of the environment. The experimentally characterized processes of motion adaptation may well play a role in an adaptive mechanism of self-motion estimation as proposed in the present study. Here we give a short analysis from a theoretical point of view which components are needed for the adaptive MFA. Minimalistically, one needs eight model neurons for the eight depth parameters. The weighted summation over the inputs of one of these model neurons corresponds to one of the eight integrals over the depth distribution, where the spherical harmonic functions play the role of the weighting parameters. Examples of neurons performing such an integration are the LPTC neurons of flies, the neuronal candidates for the six model neurons, which represent the matched filters for the six self-motion components. Given the properties of LPTCs [48, 59, 30, 9, 27], it is likely that one hypothetical model neuron for depth representation does not cover the whole sphere. Due to the linearity of self-motion estimation LPTCs can be combined to represent one self-motion component. On this basis, it might be possible that one LPTC codes information for both translation and rotation when the corresponding flow fields resemble each other within the receptive field of the neuron. Hence, the hypothetical depth neurons could be realized by a network of neurons with each neuron receiving input from only part of the visual field.

The hypothetical neurons representing the depth structure need some pre- and post-processing. In the adaptive MFA, only the pre-processing contains non-linear operations, namely the transformation of the optical flow into local depth values that are integrated afterwards by the hypothetical model neurons. One can assume that the determination of the depth values is simplified during nearly pure translation as is characteristic of the insect saccadic flight strategy [31]. In these phases the depth structure can be determined more easily, because the optic flow is not superimposed by a rotational component [28, 82].

The post-processing concerns the determination of the depth-dependent matrix which corrects the outputs of the six model neurons, corresponding to the motion sensitive LPTC cells. From a mathematical point of view, two subsequent linear transformations can be combined in a single linear transformation. In the adaptive MFA we have two subsequent linear transformations: the fixed linear transformation by the six model

neurons that receive direct optical flow input and the adaptive linear transformation by the depth-dependent matrix, the entries of which are the responses of the eight depth neurons. There are two options where the adaptation could take place: The two linear transformations could be merged into one linear transformation, which means that the adaptation takes place at an early stage of optic flow processing. Alternatively, one could assume that the two linear transformations are spatially separated, and the depth-dependent matrix is realized by an adapting linear circuitry, which wires the early stage neurons.

The linear transformation given by the depth dependent matrix can be obtained without a matrix inversion by applying an appropriate linearization of the inverse depth dependent matrix (see section 2.3.6). With this simplification and the above simplification of depth capturing the adaptive MFA can be realized by relatively simple neuronal circuits.

## 2.5 Materials and Methods

### 2.5.1 Numerical test and simulation

We used a numerical test (Fig. 2.1) and a simulation (Fig. 2.5) to show the performance of the considered self-motion estimators based on optical flow fields. The optical flow fields used here are computed in all cases from the flow equation (2.1) where the distances, the viewing directions and the self-motion parameters are given. In some cases a Gaussian error was added to the optical flow values. The task of the self-motion estimators is to determine the self-motion parameters from these flow fields.

Whereas the distances in the numerical test are obtained by a random process, the distances in the simulation are defined by the environment. In the numerical test the individual estimates are independent of each other. In the simulation a trajectory through the environment is constructed to enable an agent to use the depth model of the environment for a series of estimates at subsequent trajectory points.

For programming and testing the algorithm the programming language Matlab was used.

### 2.5.2 Construction of a spherical field of view

For the simulation and the numerical test a spherical visual field of view is needed. It is not a trivial task to arrange a number of viewing directions on a sphere in a way that the density is equally distributed. Here we used an iterative solution. The iteration starts with a Platonic solid, an octahedron. An octahedron has eight faces: Eight equilateral triangles with the same side lengths, which surround a symmetric solid.

In each iteration step every triangle is replaced by four new triangles with one triangle placed in the middle of the old triangle and the other three are placed in corners. The mid-points of the sides of the old triangle coincide with the corners of the new triangles.

These mid-points are projected radially on the surface of the sphere which surround the object so that all corners of the new triangles lie on this sphere.

In each iteration step the sphere is covered with nearly identical triangles. After the last iteration step the mid-points of each triangle give a viewing directions. With an arbitrary number of iterations $n$ the number of viewing directions are $8 \cdot 4^n$.

### 2.5.3 Numerical test of the bias of the KvD algorithm

In the numerical test of the bias of the KvD algorithm two configurations are tested, one with a spherical field of view and one with a non-spherical field. The non-spherical field of view is realized by the same procedure as in the spherical case (see methods 2.5.2), but two starting triangles are left out. The omitted triangles are opposite triangles both placed in the upper part of the sphere. This configuration has some obvious symmetry, but avoids the case that each omitted viewing direction has an omitted counterpart on the other side of the sphere.

The number of viewing directions is increased by increasing the number of iterations starting from an octahedron. Each step increases the number of viewing directions by four. For each number of viewing directions 40 self-motion estimation trials are tested.

### 2.5.4 Simulation of the adaptive MFA

The environment used for the simulation is a unit sphere in which the agent flies on a sinusoidal trajectory. The effect of adaptation should be larger if, for example, the agent flies outdoors and then enters a tunnel which is a common experimental set-up for navigation studies in flying insects [2, 1, 54, 87]. However, the test of the adaptive MFA in this kind of environment needs a 3-d engine for rendering images, on which the flow vectors are estimated with motion detectors like the Reichardt detector or the Lucas-Kanade detector. This issue will be addressed in another study (Strübbe et al., in prep.).

For the trajectory a sinusoidal curve was chosen (see Fig. 2.4 right). Although it is typical that flying insects use a saccadic flight strategy to separate translation and rotation, the self-motion estimators in this study were analyzed under the condition of a combined translation and rotation. In each run the agent flies along the trajectory and the angle error between the true self-motion axes and the estimated axes of the tested estimators are shown in Fig. 2.5.

## Supporting Information

## 2.6 Appendix

### 2.6.1 Derivation of the equivalence of the MFA and KvD algorithm

From equation (2.17) one obtains together with the equations for the coupling matrix (2.21), (2.22) and (2.23) and the definitions of the templates (2.13) and (2.14):

$$\vec{a} = \hat{M} \cdot \vec{\theta} = \begin{pmatrix} M^{tt} & M^{tr} \\ M^{rt} & M^{rr} \end{pmatrix} \cdot \begin{pmatrix} \vec{t} \\ \vec{r} \end{pmatrix} = \begin{pmatrix} \left\langle \vec{T}_1^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_2^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_3^t \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_1^r \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_2^r \cdot \vec{p} \right\rangle \\ \left\langle \vec{T}_3^r \cdot \vec{p} \right\rangle \end{pmatrix} \tag{2.44}$$

$$\begin{aligned}
\left( M^{tt} \right)_{ab} &= \left\langle \mu^2 \right\rangle \vec{e}_a \cdot \vec{e}_b - \left\langle \mu^2 \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right\rangle & (2.45) \\
\left( M^{tr} \right)_{ab} &= \left\langle \mu \left( \vec{e}_a \times \vec{e}_b \right) \cdot \vec{d} \right\rangle & (2.46) \\
\left( M^{rt} \right)_{ab} &= - \left\langle \mu \left( \vec{e}_a \times \vec{e}_b \right) \cdot \vec{d} \right\rangle & (2.47) \\
\left( M^{rr} \right)_{ab} &= \vec{e}_a \cdot \vec{e}_b - \left\langle \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right\rangle & (2.48) \\
\vec{T}_{ia}^t &= -\mu_i (\vec{d}_i \times \vec{e}_a \times \vec{d}_i) & (2.49) \\
\vec{T}_{ia}^r &= -\vec{e}_a \times \vec{d}_i & (2.50)
\end{aligned}$$

Multiplying $\vec{t}$ and $\vec{r}$ with the matrices and $\vec{p}$ with the templates leads to the two

equations:

$$(M^{tt} \cdot \vec{t} + M^{tr} \cdot \vec{r})_a = \langle \mu^2 \rangle t_a - \left\langle \mu^2 \sum_b \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \cdot t_b \right\rangle$$

$$+ \left\langle \mu \sum_b \left( (\vec{e}_a \times \vec{e}_b) \cdot \vec{d} \right) \cdot r_b \right\rangle \tag{2.51}$$

$$= \left\langle \vec{T}_a^t \cdot \vec{p} \right\rangle = \left\langle \mu \vec{p} \cdot \left( -\vec{d} \times \vec{e}_a \times \vec{d} \right) \right\rangle \tag{2.52}$$

$$(M^{rt} \cdot \vec{t} + M^{rr} \cdot \vec{r})_a = - \left\langle \mu \sum_b \left( (\vec{e}_a \times \vec{e}_b) \cdot \vec{d} \right) \cdot r_b \right\rangle$$

$$+ t_a - \left\langle \sum_b \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \cdot t_b \right\rangle \tag{2.53}$$

$$= \left\langle \vec{T}_a^r \cdot \vec{p} \right\rangle = \left\langle \vec{p} \cdot \left( \vec{d} \times \vec{e}_a \right) \right\rangle \tag{2.54}$$

The term $\sum_b \left[ \left( \vec{e}_a \cdot \vec{d} \right) \cdot \left( \vec{e}_b \cdot \vec{d} \right) \right]_{ab} \cdot v_b$ with an arbitrary vector $\vec{v}$ can be expressed through the dyadic product $\sum_b \left[ \vec{d} \otimes \vec{d} \right]_{ab} \cdot v_b = \left( \vec{v} \cdot \vec{d} \right) d_a$, and the term $\sum_b \left( (\vec{e}_a \times \vec{e}_b) \cdot \vec{d} \right) \cdot v_b$ can be transformed with the triple product rule into

$$\sum_b \left( (\vec{e}_a \times \vec{e}_b) \cdot \vec{d} \right) \cdot v_b = \left( \vec{d} \times \vec{e}_a \right) \cdot \sum_b \vec{e}_b \cdot v_b = \left( \vec{d} \times \vec{e}_a \right) \cdot \vec{v}$$

$$= \left( \vec{v} \times \vec{d} \right) \cdot \vec{e}_a = \left[ \vec{v} \times \vec{d} \right]_a \tag{2.55}$$

The right side of the equation (2.52), $\left\langle \mu \vec{p} \cdot \left( -\vec{d} \times \vec{e}_a \times \vec{d} \right) \right\rangle$, is equal to

$$\left\langle \mu \vec{p} \cdot \left( -\vec{e}_a + \left( \vec{e}_a \cdot \vec{d} \right) \vec{d} \right) \right\rangle = \langle -\mu \vec{p} \cdot \vec{e}_a \rangle = - \left[ \langle \mu \vec{p} \rangle \right]_a \ ,$$

because $\vec{p}$ and $\vec{d}$ are orthogonal. With the triple product identity $\left\langle \vec{p} \cdot \left( \vec{d} \times \vec{e}_a \right) \right\rangle$ in equation (2.54) is transformed into $\left\langle \vec{e}_a \cdot \left( \vec{p} \times \vec{d} \right) \right\rangle = \left[ \left\langle \vec{p} \times \vec{d} \right\rangle \right]_a$.

Writing the $a$ indexed values as vectors, we obtain

$$\langle \mu^2 \rangle \vec{t} - \left\langle \mu^2 \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle + \vec{r} \times \left\langle \mu \vec{d} \right\rangle = - \langle \mu \vec{p} \rangle$$

$$- \vec{t} \times \left\langle \mu \vec{d} \right\rangle + \vec{r} - \left\langle \left( \vec{r} \cdot \vec{d} \right) \vec{d} \right\rangle = \left\langle \vec{p} \times \vec{d} \right\rangle \ ,$$

which results in the iteration equations (2.11) and (2.12) of the KvD algorithm:

$$\vec{t} = - \frac{1}{\langle \mu^2 \rangle} \left\{ \langle \mu \vec{p} \rangle + \vec{r} \times \left\langle \mu \vec{d} \right\rangle - \left\langle \mu^2 \left( \vec{t} \cdot \vec{d} \right) \vec{d} \right\rangle \right\} \ ,$$

$$\vec{r} = \left\langle \vec{p} \times \vec{d} \right\rangle + \vec{t} \times \left\langle \mu \vec{d} \right\rangle + \left\langle \left( \vec{r} \cdot \vec{d} \right) \vec{d} \right\rangle \ ,$$

49

where $\frac{1}{\langle \mu^2 \rangle}$ represents the Lagrange multiplier $\xi$.

### 2.6.2 Bias-term of the KvD algorithm

The iteration equations (2.10), (2.11) and (2.12) do not converge to the true parameters $\vec{t}^0$, $\vec{r}^0$ and $\mu_i^0$ in general, even in the limit of an infinite number of flow vectors. We consider unbiased and independently distributed flow vector errors, $E(\Delta \vec{p}_i) = 0$, $\text{cov}(\Delta \vec{p}_i, \Delta \vec{p}_j) = 0$ if $i \neq j$. Their variance $\text{var}(\Delta \vec{p}_i)$ may depend on the direction $i$.

The mean of an infinite number of independent flow vector errors converges to the mean of the expectation value,

$$\langle \Delta \vec{p}_i \rangle_\infty = 0 \ , \tag{2.56}$$

for an arbitrary integration area around viewing direction $i$. Because the error vectors are independent from each other, the mean of the product of the error vectors and an arbitrary direction dependent vector function $\vec{f}_i$ is zero,

$$\left\langle \vec{f}_i \cdot \Delta \vec{p}_i \right\rangle_\infty = 0 \ . \tag{2.57}$$

To show that the KvD algorithm does not converge to the real values for an infinite number of flow vectors, we start by assuming the opposite. If the KvD algorithm had a minimum at the true values $\vec{t}^0$, $\vec{r}^0$ and $\mu_i^0$ we could insert these values in the iteration equations (2.10), (2.11) and (2.12) and would get the same values back. Substituting the true values for translation and rotation in the equation for the nearness (2.10), we get the nearness error $\triangle \mu_i$ which directly depends on the flow vector errors:

$$\triangle \mu_i = -\frac{\vec{t}^0 \cdot \Delta \vec{p}_i}{1 - \left( \vec{t}^0 \cdot \vec{d}_i \right)^2} \ .$$

The estimated translation after an infinite number of iterations $\vec{t}(\infty) = \lim_{n \to \infty} \vec{t}(n)$ shows the following equation. Please note that we consider two limits: the limit of infinite iteration steps, $\lim_{n \to \infty}$, and the limit of an infinite number of flow vectors, indicated by an infinity symbol as index of the brackets which stands for the mean over all viewing directions.

$$
\begin{aligned}
\vec{t}(\infty) &= \vec{t}^0 + \triangle \vec{t} = -\xi \left\{ \langle \mu \vec{p} \rangle_\infty + \vec{r}^0 \times \left\langle \mu \vec{d} \right\rangle_\infty - \left\langle \mu^2 \left( \vec{t}^0 \cdot \vec{d} \right) \vec{d} \right\rangle_\infty \right\} \\
&= -\xi \left\{ \left\langle (\mu^0 + \triangle \mu)(\vec{p}^0 + \Delta \vec{p}) \right\rangle_\infty + \vec{r}^0 \times \left\langle (\mu^0 + \triangle \mu) \vec{d} \right\rangle_\infty \right. \\
&\qquad \left. - \left\langle (\mu^0 + \triangle \mu)^2 \left( \vec{t}^0 \cdot \vec{d} \right) \vec{d} \right\rangle_\infty \right\} \ .
\end{aligned}
$$

If $\vec{t}(\infty)$ is a stable minimum $\triangle \vec{t}$ must vanish:

$$\triangle \vec{t} = -\xi \left\{ \left\langle \mu^0 \Delta \vec{p} + \triangle \mu \vec{p}^0 + \triangle \mu \Delta \vec{p} \right\rangle_\infty + \vec{r}^0 \times \left\langle \triangle \mu \vec{d} \right\rangle_\infty \right.$$
$$\left. - \left\langle (2\mu^0 \triangle \mu + \triangle \mu^2) \left( \vec{t}^0 \cdot \vec{d} \right) \vec{d} \right\rangle_\infty \right\}$$

$$= -\xi \left\{ \left\langle \triangle \mu \Delta \vec{p} \right\rangle_\infty - \left\langle \triangle \mu^2 \left( \vec{t}^0 \cdot \vec{d} \right) \vec{d} \right\rangle_\infty \right\}$$

$$= \xi \left\{ \left\langle \frac{\vec{t}^0 \cdot \Delta \vec{p}}{1 - \left( \vec{t}^0 \cdot \vec{d} \right)^2} \Delta \vec{p} \right\rangle_\infty + \left\langle \frac{(\vec{t}^0 \cdot \Delta \vec{p})(\Delta \vec{p} \cdot \vec{t}^0)}{\left( 1 - \left( \vec{t}^0 \cdot \vec{d} \right)^2 \right)^2} \left( \vec{t}^0 \cdot \vec{d} \right) \vec{d} \right\rangle_\infty \right\}$$

$$= \xi \left\{ \left\langle \frac{\Delta \vec{p} \otimes \Delta \vec{p}}{1 - \left( \vec{t}^0 \cdot \vec{d} \right)^2} \right\rangle_\infty \vec{t}^0 + \left\langle \frac{\vec{t}^0 \cdot (\Delta \vec{p} \otimes \Delta \vec{p}) \cdot \vec{t}^0}{\left( 1 - \left( \vec{t}^0 \cdot \vec{d} \right)^2 \right)^2} (\vec{d} \otimes \vec{d}) \right\rangle_\infty \vec{t}^0 \right\}$$

For the analysis of the conditions under which the term $\triangle \vec{t}$ vanishes, see the text (subsection 2.3.5).

### 2.6.3 Expression of $\vec{d}$ and $\vec{d} \otimes \vec{d}$ by spherical harmonics

The vector $\vec{d}$ and the dyadic product $\vec{d} \otimes \vec{d}$ are expressed by linear combinations of real-valued spherical harmonics, which are itself linear combinations of complex spherical harmonics $Y_{lm}$,

$$R_{lm} = \begin{cases} \frac{1}{\sqrt{2}} \left( Y_{lm} + (-1)^m Y_{l(-m)} \right) & m > 0 \\ Y_{l0} & m = 0 \\ \frac{1}{\sqrt{2}i} \left( Y_{l(-m)} - (-1)^m Y_{lm} \right) & m < 0 \end{cases} \tag{2.58}$$

The real-valued spherical harmonics from zero order to second-order are:

**Zeroth-Order**

$$g_0 = \sqrt{\frac{1}{4\pi}} \tag{2.59}$$

**First-order**

$$f_1 = \sqrt{\frac{3}{4\pi}} \sin \vartheta \cos \varphi \propto \sin \vartheta \cos \varphi \tag{2.60}$$

$$f_2 = \sqrt{\frac{3}{4\pi}} \sin \vartheta \sin \varphi \propto \sin \vartheta \sin \varphi \tag{2.61}$$

$$f_3 = \sqrt{\frac{3}{4\pi}} \cos \vartheta \propto \cos \vartheta \tag{2.62}$$

**Second-order**

$$h_1 = \sqrt{\frac{5}{16\pi}} \left(3\cos^2\vartheta - 1\right) \propto 3\cos^2\vartheta - 1 \tag{2.63}$$

$$h_2 = \sqrt{\frac{15}{4\pi}} \sin\vartheta\cos\vartheta\cos\varphi \propto \sin\vartheta\cos\vartheta\cos\varphi \tag{2.64}$$

$$h_3 = \sqrt{\frac{15}{4\pi}} \sin\vartheta\cos\vartheta\sin\varphi \propto \sin\vartheta\cos\vartheta\sin\varphi \tag{2.65}$$

$$h_4 = \sqrt{\frac{15}{16\pi}} \sin^2\vartheta\cos\left(2\varphi\right) \propto \sin^2\vartheta\cos\left(2\varphi\right) \tag{2.66}$$

$$h_5 = \sqrt{\frac{15}{16\pi}} \sin^2\vartheta\sin\left(2\varphi\right) \propto \sin^2\vartheta\sin\left(2\varphi\right) \tag{2.67}$$

**Expression of $\vec{d}$ through spherical harmonics**

The components of $\vec{d}$ correspond to the first-order real-valued spherical harmonics $f_i$ ($i = 1, 2, 3$). This can be seen when the components of $\vec{d}$ are written in a spherical coordinate system:

$$\vec{d} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} = \begin{pmatrix} \sin\vartheta\cos\varphi \\ \sin\vartheta\sin\varphi \\ \cos\vartheta \end{pmatrix} = \begin{pmatrix} \sqrt{\frac{4\pi}{3}} \cdot f_1 \\ \sqrt{\frac{4\pi}{3}} \cdot f_2 \\ \sqrt{\frac{4\pi}{3}} \cdot f_3 \end{pmatrix} . \tag{2.68}$$

The components $d_1$, $d_2$ and $d_3$ are equal in this arrangement to the first-order functions $f_1$, $f_2$ and $f_3$ except of a normalization factor.

**Expression of $\vec{d} \otimes \vec{d}$ through spherical harmonics**

First the dyadic product $\vec{d} \otimes \vec{d}$ is formulated in a spherical coordinate system

$$\vec{d} \otimes \vec{d} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} \otimes \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} = \begin{pmatrix} d_1^2 & d_1 d_2 & d_1 d_3 \\ d_2 d_1 & d_2^2 & d_2 d_3 \\ d_3 d_1 & d_3 d_2 & d_3^2 \end{pmatrix}$$

$$= \begin{pmatrix} \sin^2\vartheta\cos^2\varphi & \sin^2\vartheta\sin\varphi\cos\varphi & \sin\vartheta\cos\vartheta\cos\varphi \\ \sin^2\vartheta\sin\varphi\cos\varphi & \sin^2\vartheta\sin^2\varphi & \sin\vartheta\cos\vartheta\sin\varphi \\ \sin\vartheta\cos\vartheta\cos\varphi & \sin\vartheta\cos\vartheta\sin\varphi & \cos^2\vartheta \end{pmatrix} . \tag{2.69}$$

The following analysis shows that the off-diagonal elements of this matrix can each be expressed by a single second-order real-valued harmonic function, whereas the diagonal elements are linear combinations of the zeroth-order and several second-order real-valued spherical harmonic functions.

**Off-diagonal elements** The element $d_1 d_3$ is proportional to the function $h_2$ and the element $d_2 d_3$ is proportional to the function $h_3$.

$$d_1 d_3 = \sqrt{\frac{4\pi}{15}} \cdot h_2$$

$$d_2 d_3 = \sqrt{\frac{4\pi}{15}} \cdot h_3$$

To see that the element $d_1 d_2$ is proportional to the function $h_5$, $h_5$ must be rearranged,

$$h_5 = \sqrt{\frac{15}{16\pi}} \cdot \sin^2 \vartheta \sin(2\varphi) = \sqrt{\frac{15}{4\pi}} \cdot \sin^2 \vartheta \sin \varphi \cos \varphi , \qquad (2.70)$$

$$d_1 d_2 = \sqrt{\frac{4\pi}{15}} \cdot h_5 , \qquad (2.71)$$

via the addition theorem: $\sin(2x) = 2 \cdot \sin(x) \cos(x)$.

**Diagonal elements** The diagonal element $d_3^2 = \cos^2 \vartheta$ can be expressed by a proper linear combination of $h_1 = \sqrt{\frac{5}{16\pi}} \cdot (3\cos^2 \vartheta - 1) = \sqrt{\frac{45}{16\pi}} \cdot \cos^2 \vartheta - \sqrt{\frac{5}{16\pi}}$ and $g_0 = \sqrt{\frac{1}{4\pi}}$, in such a way that the constant in $h_1$ compensate the constant in $g_0$.

$$h_1 + \sqrt{\frac{5}{4}} \cdot g_0 = \sqrt{\frac{45}{16\pi}} \cdot \cos^2 \vartheta$$

$$d_3^2 = \sqrt{\frac{16\pi}{45}} \cdot \left( h_1 + \sqrt{\frac{5}{4}} \cdot g_0 \right) = \sqrt{\frac{16\pi}{45}} \cdot h_1 + \sqrt{\frac{4\pi}{9}} \cdot g_0$$

For the expression of $d_1^2$ and $d_2^2$ the function $h_4$ needs a closer examination. With $\cos(2x) = \cos^2 x - \sin^2 x$ the function $h_4$,

$$h_4 = \sqrt{\frac{15}{16\pi}} \sin^2 \vartheta \cos(2\varphi) = \sqrt{\frac{15}{16\pi}} \sin^2 \vartheta \left( \cos^2 \varphi - \sin^2 \varphi \right) , \qquad (2.72)$$

is rearranged in the two versions $h_4^a$ and $h_4^b$,

$$
\begin{aligned}
h_4^a &= \sqrt{\frac{15}{16\pi}}\sin^2\vartheta\left(1 - 2\cdot\sin^2\varphi\right)\\
&= \sqrt{\frac{15}{16\pi}}\sin^2\vartheta - \sqrt{\frac{15}{4\pi}}\cdot\sin^2\vartheta\sin^2\varphi\ , & (2.73)
\end{aligned}
$$

$$
\sin^2\vartheta\sin^2\varphi = -\sqrt{\frac{4\pi}{15}}h_4^a + \sqrt{\frac{1}{4}}\sin^2\vartheta \qquad (2.74)
$$

$$
\begin{aligned}
h_4^b &= \sqrt{\frac{15}{16\pi}}\sin^2\vartheta\left(-1 + 2\cdot\cos^2\varphi\right)\\
&= -\sqrt{\frac{15}{16\pi}}\sin^2\vartheta + \sqrt{\frac{15}{4\pi}}\cdot\sin^2\vartheta\cos^2\varphi & (2.75)
\end{aligned}
$$

$$
\sin^2\vartheta\cos^2\varphi = \sqrt{\frac{4\pi}{15}}h_4^b + \sqrt{\frac{1}{4}}\sin^2\vartheta\ , \qquad (2.76)
$$

with the use of $\cos^2\varphi + \sin^2\varphi = 1$. The second term in $h_4^a$ has the same form as the element $d_1^2$ and the second term in $h_4^b$ equals $d_2^2$. It is left to show, that the term $\sin^2\vartheta$ can be expressed by the zero and the second-order real-valued spherical harmonic functions.

To see this the term $\sqrt{\frac{1}{4}}\sin^2\vartheta$ will be rearranged,

$$
\begin{aligned}
\sqrt{\frac{1}{4}}\sin^2\vartheta &= \sqrt{\frac{1}{4}} - \sqrt{\frac{1}{4}}\cos^2\vartheta = -\sqrt{\frac{1}{36}}\left(3\cos^2\vartheta - 1\right) + \sqrt{\frac{1}{9}}\ , & (2.77)\\
&= -\sqrt{\frac{4\pi}{45}}h_1 + \sqrt{\frac{4\pi}{9}}g_0 & (2.78)
\end{aligned}
$$

which is a linear combination of $h_1$ and $g_0$.

Together $d_1^2$ and $d_2^2$ can be expressed through the following spherical harmonics:

$$
\begin{aligned}
d_1^2 &= -\sqrt{\frac{4\pi}{15}}h_4 - \sqrt{\frac{4\pi}{45}}h_1 + \sqrt{\frac{4\pi}{9}}g_0\\
d_2^2 &= \sqrt{\frac{4\pi}{15}}h_4 - \sqrt{\frac{4\pi}{45}}h_1 + \sqrt{\frac{4\pi}{9}}g_0
\end{aligned}
$$

### 2.6.4 The weight matrix of the original MFA

To see that equation (2.20) corresponds to equation (2.4) as derived by Franz et al. in 2004 [35] for the original MFA, a change of the basis of the vector space is applied. In [35] the coordinates of each flow vector $\vec{p}_i$ were given with respect to different basis vectors $\vec{u}_i$, $\vec{v}_i$, spanning the tangential plane on the sphere for viewing direction $\vec{d}_i$. In the following we derive the transformation matrix to transform the coordinates of a vector

given in the standard Euclidean basis vectors $\vec{e}_1$, $\vec{e}_2$ and $\vec{e}_3$ into the basis defined by $(\vec{u}_i, \vec{v}_i, \vec{d}_i)_{i=1,2,\ldots,N}$. With $N$ optical flow vectors we define a $3 \times N$ dimensional vector space, which is the $N$-fold Cartesian product of the three basis vectors $\vec{e}_1$, $\vec{e}_2$ and $\vec{e}_3$.

All flow vectors $\vec{p}_i$ are represented now by a single stacked column vector $\vec{p}$,

$$\vec{p} = \begin{pmatrix} \vec{p}_1 \\ \vec{p}_2 \\ \vdots \\ \vec{p}_N \end{pmatrix} \, , \tag{2.79}$$

which has the dimension of $3 \times N$. In the same way the templates are written as

$$\vec{T}_A = \begin{pmatrix} \vec{T}_{1,A} \\ \vec{T}_{2,A} \\ \vdots \\ \vec{T}_{N,A} \end{pmatrix} \, , \tag{2.80}$$

where the index $A$ stands for one of the six standard templates. The six templates $\vec{T}_A$ are combined to matrix $\hat{T}$ with six columns and $3 \times n$ rows :

$$\hat{T} = \begin{pmatrix} \vec{T}_{1,1} & \vec{T}_{1,2} & \cdots & \vec{T}_{1,6} \\ \vec{T}_{2,1} & \vec{T}_{2,2} & \cdots & \vec{T}_{2,6} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{T}_{N,1} & \vec{T}_{N,2} & \cdots & \vec{T}_{N,6} \end{pmatrix} \, . \tag{2.81}$$

The responses of the six model neurons $\vec{a}$ in this notation are

$$\vec{a} = \hat{T}^T \cdot \vec{p} \, , \tag{2.82}$$

where $\hat{T}^T$ stands for the transpose of $\hat{T}$. Together with the coupling matrix one obtains for the self-motion components (see equation (2.20)):

$$\vec{\theta}^{\text{est}} = \hat{M}^{-1} \cdot \vec{a} = \hat{M}^{-1} \cdot \hat{T}^T \cdot \vec{p} \, . \tag{2.83}$$

The complete vector basis of the above templates and flow vectors is described by two indices $j = 1, 2, 3$ and $i = 1, 2, \ldots, N$ and has the form

$$B = (\vec{e}_{1,1}, \vec{e}_{2,1}, \vec{e}_{3,1}, \vec{e}_{1,2}, \vec{e}_{2,2}, \vec{e}_{3,2}, \ldots, \vec{e}_{1,N}, \vec{e}_{2,N}, \vec{e}_{3,N}) \, , \tag{2.84}$$

where the $\vec{e}_{1,i}$, $\vec{e}_{2,i}$ and $\vec{e}_{3,i}$ are the same for each $i$. The basis vectors in the notation of Franz et al. [35] are

$$B' = \left( \vec{u}_1, \vec{v}_1, \vec{d}_1, \vec{u}_2, \vec{v}_2, \vec{d}_2, \ldots, \vec{u}_n, \vec{v}_n, \vec{d}_n \right) \, , \tag{2.85}$$

where the viewing directions $\vec{d_i}$ supplemented the local vector spaces to a three dimensional space.

The new basis vectors $B'$ can be expressed by the old $B$,

$$\begin{aligned}
\vec{u}_i &= u_{1,i}\vec{e}_{1,i} + u_{2,i}\vec{e}_{2,i} + u_{3,i}\vec{e}_{3,i} \ , \\
\vec{v}_i &= v_{1,i}\vec{e}_{1,i} + v_{2,i}\vec{e}_{2,i} + v_{3,i}\vec{e}_{3,i} \ , \\
\vec{d}_i &= d_{1,i}\vec{e}_{1,i} + d_{2,i}\vec{e}_{2,i} + d_{3,i}\vec{e}_{3,i} \ ,
\end{aligned}$$

where $u_{1,i}$ etc. is one component of $\vec{u}_1$. The transformation matrix $\hat{V}$ between the two bases with the dimension $[3 \times N] \cdot [3 \times N]$ is

$$\hat{V} = \begin{pmatrix}
u_{1,1} & u_{2,1} & u_{3,1} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
v_{1,1} & v_{2,1} & v_{3,1} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
d_{1,1} & d_{2,1} & d_{3,1} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & u_{1,2} & u_{2,2} & u_{3,2} & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & v_{1,2} & v_{2,2} & v_{3,2} & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & d_{1,2} & d_{2,2} & d_{3,2} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & u_{1,N} & u_{2,N} & u_{3,N} \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & v_{1,N} & v_{2,N} & v_{3,N} \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & d_{1,N} & d_{2,N} & d_{3,N}
\end{pmatrix} .$$

The matrix $\hat{V}$ transforms the coordinates of a vector given in basis $B$, equation (2.84), into basis $B'$, equation (2.85). Equation (2.83) becomes

$$\begin{aligned}
\vec{\theta}^{\text{est}} &= \hat{M}^{-1} \cdot \hat{T}^T \cdot \vec{p} = \left(\hat{T}^T \cdot \hat{T}\right)^{-1} \cdot \hat{T}^T \cdot \vec{p} \\
&= \left(\hat{T}^T \cdot \hat{V}^T \cdot \hat{V} \cdot \hat{T}\right)^{-1} \cdot \hat{T}^T \cdot \hat{V}^T \cdot \hat{V} \cdot \vec{p} \\
&= \left(\left(\hat{V} \cdot \hat{T}\right)^T \cdot \hat{V} \cdot \hat{T}\right)^{-1} \cdot \left(\hat{V} \cdot \hat{T}\right)^T \cdot \hat{V} \cdot \vec{p} \ .
\end{aligned}$$

The term $\hat{V} \cdot \vec{p}$ is equivalent to $\vec{x}$ in equation (2.4) from the original MFA of Franz et al. [35]. The term $\vec{x}$ combines all optic flow components of the local two dimensional vector spaces as spanned by the two local vectors $\vec{u}$ and $\vec{v}$. The third component is zero because $\vec{p}$ and $\vec{d}$ are orthogonal.

To see that $\hat{V} \cdot \hat{T}$ is $F = \begin{pmatrix} -\bar{\mu}_1\vec{u}_1 & -\bar{\mu}_1\vec{v}_1 \\ \vec{u}_1 \times \vec{d}_1 & , & \vec{v}_1 \times \vec{d}_1 & , \ldots \end{pmatrix}^T$ the first row of $\hat{V} \cdot \hat{T}$ is scrutinized. The six components are

$$\vec{u}_1 \cdot \vec{T}_{1,1}, \ \vec{u}_1 \cdot \vec{T}_{1,2}, \ \vec{u}_1 \cdot \vec{T}_{1,3}, \ \vec{u}_1 \cdot \vec{T}_{1,4}, \ \vec{u}_1 \cdot \vec{T}_{1,5}, \ \vec{u}_1 \cdot \vec{T}_{1,6} \ .$$

The first three components are $(j = 1, 2, 3)$

$$-\mu_1 \cdot \vec{u}_1 \cdot \left( \vec{d}_1 \times \vec{e}_j \times \vec{d}_1 \right) = -\mu_1 \cdot \vec{u}_1 \cdot \left( \vec{e}_j - \left( \vec{e}_j \cdot \vec{d}_1 \right) \vec{d}_1 \right) = -\mu_1 \cdot \vec{u}_{j,1} \ ,$$

and the second three components are

$$-\vec{u}_1 \cdot \left( \vec{e}_j \times \vec{d}_1 \right) = \vec{e}_j \cdot \left( \vec{u}_1 \times \vec{d}_1 \right) = \left( \vec{u}_1 \times \vec{d}_1 \right)_j \ .$$

# 3 Learning Depth Models for Egomotion Estimation with Dense Flow Fields – A Test of the Adaptive Matched Filter Approach

## 3.1 Abstract

Optical flow caused by the motion of an agent can be used to determine its self-motion. The depth structure of an environment is important for self-motion estimation, because the amplitude of optical flow vectors caused by translation of the agent is proportional to the nearnesses (inverse distances) of objects in the environment. While the biologically inspired matched filter approach of Franz and Krapp [36, 33, 32, 34, 35] is based on a fixed depth model of the environment, an adaptive matched filter approach that continuously updates an efficient representation of the depth structure has been proposed recently and analysed mathematically, assuming idealized flow fields [89]. Here we present a quantitative analysis of the adaptive matched filter approach based on flow fields estimated from rendered image sequences in virtual 3D environments. Motivated by the extremely large fields of view found in flying insects, we focus on the case of a spherical field of view. For this spherical case, it was shown [89] that the depth model reduces to explicitly defined nine parameters. We show that the adaptive matched filter approach performs significantly better than the original non-adaptive matched filter approach in environments with a changing depth distributions.

## 3.2 Introduction

Insects have an astonishing capability to find back to their nest or food places they have explored before. Diverse studies have shown that they can use for this task multiple cues like color, the spatial configuration of landmarks as well as path integration [106]. Whereas walking insects can count their steps, flying insects seem to rely predominantly on their visual system for distance estimation. A precondition for visual path integration is the ability to estimate the self-motion. In this context it has been hypothesized that so-called tangential cells in the visual system of flies function as matched filters for flow fields generated during self-motion [36, 33, 32, 34, 35]. However, flow fields do

not only depend on the self-motion of the insect but also on the depth structure of the environment. Therefore, an additional adaptation process has been proposed recently taking into account the changing environment [89].

It has been shown that nine depth-dependent parameters contain all information relevant for self-motion estimation. For a fully spherical field of view, the nine parameters are the zeroth, first and second order coefficients of an expansion of the depth structure with spherical harmonic functions [89]. As the coefficients are calculated as the weighted sum of depth values, they could theoretically be represented by nine neurons in the visual system.

In flies, the depth-dependent adaption may directly occur in the tangential cells. Although motion adaptation has been characterised in the tangential cells [42, 65], the detected change of optic flow could have many reasons. Here we postulate an adaptation mechanism which is based on the detection of nine specific environmental parameters.

So far, the adaptive matched filter model has only been tested with simple and artificially generated flow fields[89]. Here we evaluate the characteristics of the model based on flow fields from local motion detectors. The motion detectors operate on images rendered from the view point of an agent flying in a virtual enviroment. The advantage of using a virtual environment is that estimated optical flow as well as estimated egomotion can be compared easily with ground truth. We use the Lucas-Kanade motion detector which is a gradient-based method [18]. In accordance with biological optic flow estimation the motion detectors were arranged in a 2D grid, and ego-motion estimation is based on dense flow fields and not on tracking or matching of selected image features [83].

The results from the model simulations support our theoretical calculations predicting that the error of the estimated self-motion is the sum of the error due to the variance of the flow detectors and the error due to an incorrect depth model with the latter dominating the former.

As adaptation can only be effective if it takes place at a higher rate than the rate of change in the environmental structure, we also show that the adaptation decreases the difference between the true and estimated depth parameter exponentially over time.

## 3.3 Methods

### 3.3.1 3D models

In order to test and analyze the adaptive matched filter and compare its performance with the non-adaptive version [36, 33, 32, 34, 35] two 3D environments were created using VRML (Virtual Reality Modeling Language). As we wanted to evaluate the adaptive matched filter model with a spherical field of view, the 3D environments are closed spaces that provide contrast in all viewing directions.

The geometry of the first 3D environment corresponds to a cube. The trajectory of the agent inside the cube is close to the bottom wall, so that the depth structure from the
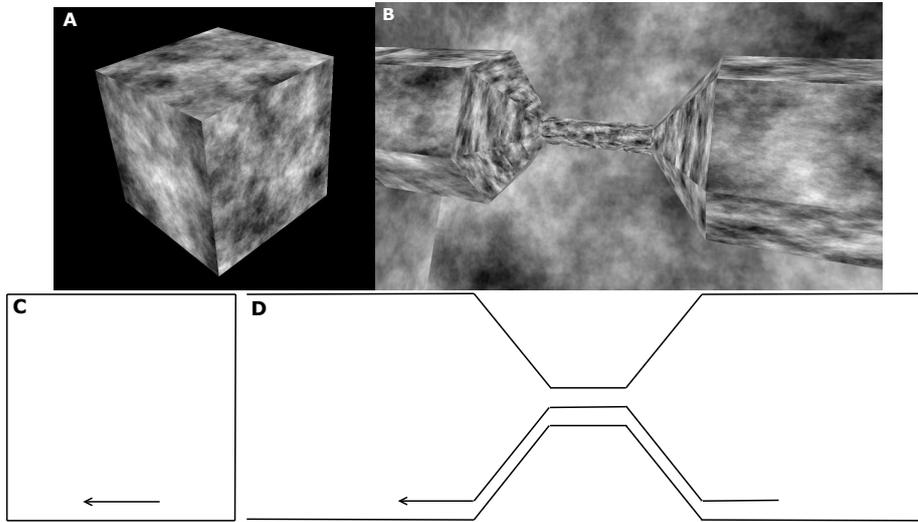
**Figure 3.1:** A: Exterior view of the tested cube 3D model environment. The cube has a side length of 300 units. The agent flies 25 units above the ground of the cube, which leads to a large depth asymmetry in the z-direction. The resulting depth structure as seen from the viewpoint of the agent is asymmetrical but nearly the same over the whole trajectory. B: The second 3D model environment is designed to exhibit a significantly changing depth structure during the flight. It has a narrow tube in its center (50 units in diameter) and wider cylindrical parts at the end and beginning (300 units in diameter) which are connected by conic funneling elements. In the wider cylindrical parts, the agent flies with a constant flight altitude of 25 units which leads to a large depth asymmetry, whereas within the narrow tube the agent flies in the middle. C-D: True-to-scale representations of a side view of the trajectories within the cube (subfigure C) and the 3D model with the narrow tube (subfigure D)

perspective of the agent is asymmetrical but nearly the same over the whole trajectory (see Fig. 3.1A). The second, more complex 3D model environment (Fig. 3.1B) is also surrounded by a cube, but contains in its center a narrow tube that can be entered by funnel-like structures on both sides providing a changing depth structure.

All surfaces of the 3D models are covered by random contrast textures that have spatial frequency amplitudes proportional to $1/frequency^{1.5}$, imitating the spectrum of natural images [97].

The simulated agent receives visual input from six cameras with optical axes oriented along the six basic directions of an agent-centered 3D coordinate system. These six images received at every trajectory point are the basis for all following calculations.

### 3.3.2 Implementation of the trajectories

The trajectories for the two 3D environments are built in a way that the depth distribution of the environment is asymmetrical (3D-cube environment) or exhibits additionally large changes (3D-constriction environment). The trajectories of the agent in the two 3D environments are shown in figure 3.1C-D. To control the maximum translational optic flow the agent has a constant flight altitude. The translational optic flow is overlaid by a rotational optic flow in a way that these two optic flow components have similar magnitudes. Here, we tested the original non-adaptive matched filter approach (MFA) against the proposed adaptive MFA and avoided the problem that one of the two optic flow components is much smaller than the other. Otherwise, the smaller component would be more difficult to be detected or may even be lost due to the motion detector variance. In doing so, the overall optic flow is kept always in the limited operating range of the used motion detector (see section 3.3.3).

Whereas the translation of the agent follows strictly the trajectories shown in figure 3.1C-D, its rotation is characterized by a random rotation axis and a random magnitude varying in a certain range, ensuring that the caused optic flow is between one and five pixels per frame (see section 3.3.3). At one step, a random rotation is applied to the agent, but in the next step the agent is reoriented in the original direction. In this way the agent corrects an imposed rotational deviation from its trajectory.

### 3.3.3 Implementation of the Lucas-Kanade motion detector

The Lucas-Kanade detector is a gradient-based motion detector (basically, it determines the quotient of the temporal and spatial gradient of the image) estimating the image velocity at a certain point in the image [18]. Due to its simplicity and good perfomance the Lukas-Kanade detector is a standard motion detector in computer vision and employed in many studies [3].

A single point in the image is not sufficient to solve the equations for the image velocities in x-direction $v_x$ and y-direction $v_y$. Hence, the algorithm assumes that the image velocities are the same within a small window. The Lucas-Kanade detector solves the following overdetermined equation system:

$$\begin{bmatrix} I_{x1} & I_{y1} \\ I_{x2} & I_{y2} \\ \vdots & \vdots \\ I_{xn} & I_{yn} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} -I_{t1} \\ -I_{t2} \\ \vdots \\ -I_{tn} \end{bmatrix},$$

where $I$ is the image intensity, $I_x$, $I_y$ are the spatial intensity gradients in the x- and y-direction, $I_t$ is the time derivative and $1, 2, \cdots, n$ are different points in a small window. Gradients are estimated from discrete approximations, i.e. neighboring pixels for spatial gradients and brightness differences between consectutive images for temporal gradients.

We implemented a Lukas-Kanade detector which is not pyramidal [93] limiting the range of image velocities that can be detected. As a pre-processing step the original camera image is scaled down by a factor of five using a 2D Gaussian window function leading to 6 x 45^2 = 12150 flow values for all six cameras together. The original Lukas-Kanade detector compares neighboring pixels and computes spatial intensity gradients with them. The down sampling of the image by a factor of five leads to an operating range of up to five pixels per frame.

For all forthcoming calculations we use three-dimensional optic flow vectors. To obtain them, $v_x$ and $v_y$ are projected on the viewing sphere and combined with the corresponding orthogonal direction vector $\vec{d_i}$.

### 3.3.4 Matched filter algorithms for egomotion estimation

The output of the motion detector array is fed into two different variants of self-motion estimation algorithms employing matched filters. The first variant uses a fixed-depth model and represents the original matched filter approach of Franz and Krapp [36, 33, 32, 34, 35]. The second one, the recently proposed adaptive matched filter approach [89] has an additional adaptation step that allows to update the internal depth model. The depth model is updated at each step of the trajectory. However, we also tested a variant which updates at a much lower rate.

**The adaptive matched filter approach**   The adaptive matched filter model was derived from the Koenderink and van Doorn algorithm [57], which is an iterative approach to self-motion estimation. It determines both the self-motion parameters and the depth values in parallel from a given optical flow field. Because the depth values influence the estimate of self-motion, and the self-motion parameters have impact on the depth model, they are corrected iteratively until convergence. Since an iterative algorithm is biologically implausible, the adaptive matched filter approach is based on the assumption that the environment changes its global depth properties only slowly relative to the adaptation time constant so that the depth model is sufficiently correct over several time steps.

The adaptation process to the correct depth values of the environment is the same for the KvD algorithm and the adaptive MFA. However, the former improves the estimation of the depth values only at one trajectory step, while the latter combines all depth values into a depth model and adapts this along the entire trajectory. To test the adaptation rate one arbitrary trajectory point is holding fixed, so that the true environmental parameters do not change during the adaptation process (see section 3.4.2).

It has been shown in [89] that the original MFA with an additional correction matrix [35] is equivalent to the KvD algorithm, if the distances to objects in the environment

are known. Both approaches lead to the following equations for self-motion estimation:

$$\begin{pmatrix} M^{tt} & M^{tr} \\ M^{rt} & M^{rr} \end{pmatrix} \cdot \begin{pmatrix} \vec{t} \\ \vec{r} \end{pmatrix} = \begin{pmatrix} \langle -\vec{p} \rangle \\ \langle \vec{p} \times \vec{d} \rangle \end{pmatrix} . \tag{3.1}$$

Where the $\vec{p}_i$ are the estimated optical flow vectors and the $\vec{d}_i$ the corresponding viewing directions (unit vectors) in which the flow is measured. The brackets $\langle \rangle$ stand for the summation over all directions, $\langle x \rangle = \frac{1}{N} \sum_{i=1}^{N} x_i$, with $N$ the number of viewing directions. The estimated self-motion parameters are $\vec{t}$ for the translation and $\vec{r}$ for the rotation. The $6 \times 6$ correction matrix $M$ is expressed by the four $3 \times 3$ matrices $M^{tt}$, $M^{tr}$, $M^{rt}$ and $M^{rr}$. To estimate the self-motion the matrix M must be inverted.

Whereas the matrix $M$ is constant in the original MFA approach the elements of M vary in the adaptive MFA. Here we analyze the case of a spherical field of view. For this case, it was shown [89] that the variation of the correction matrix $M$ can be described by only nine parameters characterizing the depth distribution of the environment. Accordingly, we assume nine model neurons in the adaptive MFA which are sensitive to the depth structure.

When approximating depth by a continuous function of spherical coordinates these parameters can be represented as the first nine coefficients of an expansion of the depth function $\mu(\vartheta, \varphi)$ by spherical harmonic functions [89]. Here we replace the depth values $\mu_i$ by their respective continuous counterpart $\mu(\vartheta, \varphi)$ with the azimuth angle $\varphi$ and the elevation angle $\vartheta$. The nine coefficients, which represent the output of the nine model neurons, are given by weighted summations over the depth $\mu(\vartheta, \varphi)$, where the weighting functions are the real valued spherical harmonic functions $R_{ln}(\vartheta, \varphi)$:

$$a = \int_{\varphi=0}^{2\pi} \int_{\vartheta=0}^{\pi} R_0(\vartheta, \varphi) \cdot \mu(\vartheta, \varphi) \cdot \sin\vartheta d\vartheta d\varphi$$

$$b_{1...3} = \int_{\varphi=0}^{2\pi} \int_{\vartheta=0}^{\pi} R_{1,1...3}(\vartheta, \varphi) \cdot \mu(\vartheta, \varphi) \cdot \sin\vartheta d\vartheta d\varphi$$

$$c_{1...5} = \int_{\varphi=0}^{2\pi} \int_{\vartheta=0}^{\pi} R_{2,1...5}(\vartheta, \varphi) \cdot \mu(\vartheta, \varphi) \cdot \sin\vartheta d\vartheta d\varphi$$

The real-valued spherical harmonics from zero order to second order are:

**Zero Order**

$$R_0 = \sqrt{\frac{1}{4\pi}}$$

**First Order**

$$R_{1,1} = \sqrt{\frac{3}{4\pi}} \sin\vartheta \cos\varphi$$

$$R_{1,2} = \sqrt{\frac{3}{4\pi}} \sin\vartheta \sin\varphi$$

$$R_{1,3} = \sqrt{\frac{3}{4\pi}} \cos\vartheta$$

**Second Order**

$$R_{2,1} = \sqrt{\frac{5}{16\pi}} \left(3\cos^2\vartheta - 1\right)$$

$$R_{2,2} = \sqrt{\frac{15}{4\pi}} \sin\vartheta \cos\vartheta \cos\varphi$$

$$R_{2,3} = \sqrt{\frac{15}{4\pi}} \sin\vartheta \cos\vartheta \sin\varphi$$

$$R_{2,4} = \sqrt{\frac{15}{16\pi}} \sin^2\vartheta \cos\left(2\varphi\right)$$

$$R_{2,5} = \sqrt{\frac{15}{16\pi}} \sin^2\vartheta \sin\left(2\varphi\right)$$

The expansion ranges only up to the second order of the spherical harmonic functions. The coefficient $a$ represents the zeroth order, the coefficients $b_{1\ldots3}$ represents the dipoles and the coefficients $c_{1\ldots5}$ represents the quadrupoles of the depth distribution of the environment. It has been shown [89] that for a spherical field of view the higher orders contain no information about self-motion.

The four $3 \times 3$ matrices $M^{tt}$, $M^{tr}$, $M^{rt}$ and $M^{rr}$ consist of the varying coeffients $a, b, c$

and constants:

$$M^{tt} = \frac{\sqrt{4\pi}}{a} \cdot \left( I - \frac{1}{3}I - \sqrt{\frac{4\pi}{15}} \cdot C \right) \tag{3.2}$$

$$C = \begin{pmatrix} -c_4 - \sqrt{\frac{1}{3}} \cdot c_1 & c_5 & c_2 \\ c_5 & c_4 - \sqrt{\frac{1}{3}} \cdot c_1 & c_3 \\ c_2 & c_3 & \sqrt{\frac{4}{3}} \cdot c_1 \end{pmatrix} \tag{3.3}$$

$$M^{tr} = 0 \tag{3.4}$$

$$M^{rt} = \sqrt{\frac{4\pi}{3}} \cdot B \tag{3.5}$$

$$B = \begin{pmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{pmatrix} \tag{3.6}$$

$$M^{rr} = \frac{2}{3}I \tag{3.7}$$

Although the computation of the depth values can be biologically simplified by taking, for example, the saccadic flight behavior of insects [81, 7] into account, we use the exact equation from the KvD approach to determine these values:

$$\mu_i = -\frac{\vec{t} \cdot \left( \vec{p}_i - \vec{d}_i \times \vec{r} \right)}{1 - \left( \vec{t} \cdot \vec{d}_i \right)^2}. \tag{3.8}$$

When the self-motion parameters $\vec{t}$ and $\vec{r}$ are determined by the algorithm at one point of the trajectory, one can obtain the depth values and so the nine coefficients of the depth model for the next steps.

The used algorithm has the following steps:

1. Initialization of the depth model
   The depth model is initialized with constant nearness values ($\mu_i = 1$). All dipoles and quadrupoles are zero in the beginning.

2. Estimation of the optical flow field and of the self-motion parameters $\vec{t}$ and $\vec{r}$
   The optical flow field is computed by the Lukas-Kanade motion detector applied on the six camera images leading to six optical flow fields, which are combined to one spherical optical flow field. The current depth model, which consists of the nine values $a$, $b_{1...3}$, $c_{1...5}$, determines the matrix M. The matrix M and the optical flow vectors $\vec{p}_i$ determine the self-motion parameters $\vec{t}$ and $\vec{r}$.

3. Updating the depth model
   The self-motion parameters $\vec{t}$ and $\vec{r}$ together with the optical flow vectors $\vec{p_i}$ determine the nearnesses $\mu_i$. The nearnesses $\mu_i$ are used to estimate the new nine depth model parameters. (We also test a variant in which this updating overleaps several trajectory steps.)

4. Go to the next trajectory point and begin with (2.).

## 3.4 Results

### 3.4.1 Characteristics of the Lukas-Kanade detector

In this section different properties of the performance of the Lukas-Kanade motion detector are examined and presented. The input to a 2D array of $600 \times 600$ Lukas-Kanade motion detectors is a vertically moving random pattern with a spatial frequency distribution of natural images [97], i.e. the amplitudes of the Fourier coeffcients scale with $1/f^\alpha$ , where f is the spatial frequency and the exponent alpha a small number, typically ranging between 1 and 2 (in this study, we used $\alpha = 1.5$).

Fig. 3.2B displays the scatter of the detector responses for a constant velocity of the moving image. The scatter is symmetrical around the true optical flow. The distribution of the scatter resembles a Gaussian.

Fig. 3.2A verifies that the means of the detector estimates are linear in the presented optic flow within a window of five pixels per frame. Since the optic flow vectors induced by a self-rotation and a self-translation are added linearly to the overall optic flow vector, linearity in the estimates is a precondition for separating these two components of self-motion in later operation steps. In the same figure, the standard deviations are shown. The response scatter of a Lukas-Kanade-detector deviates from a perfect Gaussian detector: The standard deviation increases with an increasing optic flow and is, therefore, not constant.

The conditions for optic flow estimation are slightly more demanding, if the detector is used in a 3D model environment, because the optic flow differs in different viewing directions. However, neighboring viewing directions are characterized by similar optic flow values for a self-motion-induced optic flow field. Hence the spatial Gaussian windows of the Lukas-Kanade-detector play a subordinated role under these conditions. The discontinuities in the used 3D environmental models can be neglected, because the number of flow values at these positions is much smaller than the overall number of flow values.

### 3.4.2 Learning of the depth model

The actual depth model consisting of the nine parameters is used to estimate the current self-motion values. In the adaptive MFA, the depth model has to be updated from time
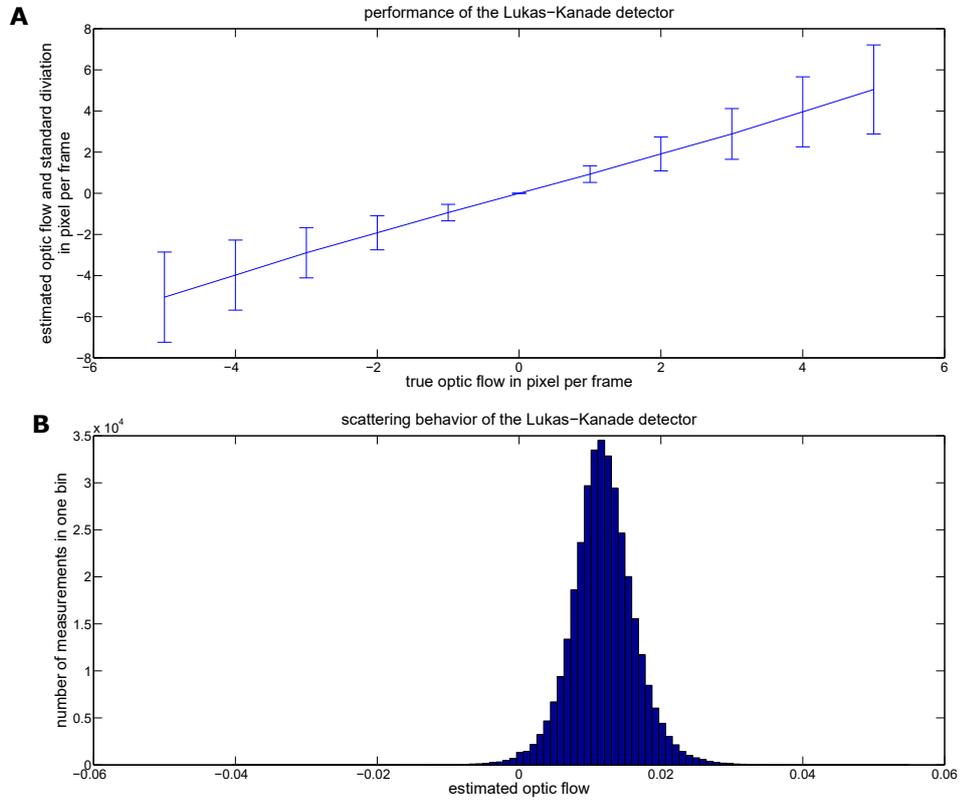
**Figure 3.2:** Response scatter of the Lukas-Kanade-detector is analyzed with the help of a random pattern with a frequency distribution resembling natural images. The pattern is shifted in the x-direction at a constant velocity; hence the optical flow is the same in the whole image. To approximate the limes of infinite flow estimates a large number of $600 \times 600$ single detectors are arranged over the image. Subfigure A shows the means of the detector outputs together with the standard deviations at different velocities. The means coincide with the true velocity which is expected for an unbiased estimator. Whereas the standard deviations show a certain characteristic of the Lukas-Kanade-detector: They are proportional to the magnitude of the detected optical flow. Subfigure B is a histogram which counts the numbers of estimated flow values within small bins for a given velocity. The bins have their maximum at the true velocity with symmetrical scattering around the maximum. The scattering behavior resembles a Gaussian bell curve.

to time. Given the current estimated self-motion values and the current optical flow values the nearnesses $\mu_i$ can be computed:

$$\mu_i = -\frac{\vec{t} \cdot \left( \vec{p_i} - \vec{d_i} \times \vec{r} \right)}{1 - \left( \vec{t} \cdot \vec{d_i} \right)^2},$$

where $\vec{p_i}$ is the optical flow in viewing direction $\vec{d_i}$ estimated by the motion detector and $\vec{t}$ and $\vec{r}$ the current estimates of self-translation and self-rotation, respectively. (For approximations to this equation, which are biologically easier to compute, see the discussion.) Consider that $\vec{t}$ and $\vec{r}$ are values that are obtained by averaging over the error-prone flow vectors $\vec{p_i}$. The error of the single nearness estimate $\mu_i$ depends on the error of the corresponding flow vector $\vec{p_i}$ (with a small non-linearity due to the term $1 - \left( \vec{t} \cdot \vec{d_i} \right)^2$). However, the estimated nearnesses $\mu_i$ are linearly combined to the nine parameters of the depth model, which corresponds to an averaging over the error-prone flow vectors $\vec{p_i}$. In section 3.4.3 it will be shown that the errors of $\vec{p_i}$ are more crucial when estimating the values of $\vec{t}$ and $\vec{r}$ than when estimating the adapting depth model, although the depth model depends on more error-prone values. The adapting depth model is thus more robust. It can slightly deviates from the correct value, but gives nonetheless good results.

**Depth model adaptation at one point**

In this section, the adaptation behavior of the three dipoles and five quadrupoles is analyzed. It was mentioned before (section 3.3.4) that these nine parameters contain all depth information required for self-motion estimation as long as the field of view is the whole sphere. We choose an arbitrary trajectory point in the 3D model environment with the constriction. It is only important that the depth distribution is asymmerical at this point, to which the algorithm has to adapt. (To obtain the subfigures 3.3 the true optic flow values and the true nearnesses are computed with the help of the depth-buffer of the rendering engine.)

The adaptation behavior is shown in Figure 3.3 at one single trajectory point so that the true depth model does not change during adaptation. Adaptation reduces the error of the dipoles and quadrupoles exponentially, after an onset time where the error may increase (see figure 3.3). This onset time plays no role, if one assumes that the algorithm has already adapted to a certain degree before the agent is exposed to the optical flow field corresponding to the next trajectory point.

It is assumed that the true depth parameters change only slightly from one trajectory point to the next rather than exponentially. Hence, the adaptation can follow the changes of the depth distribution.
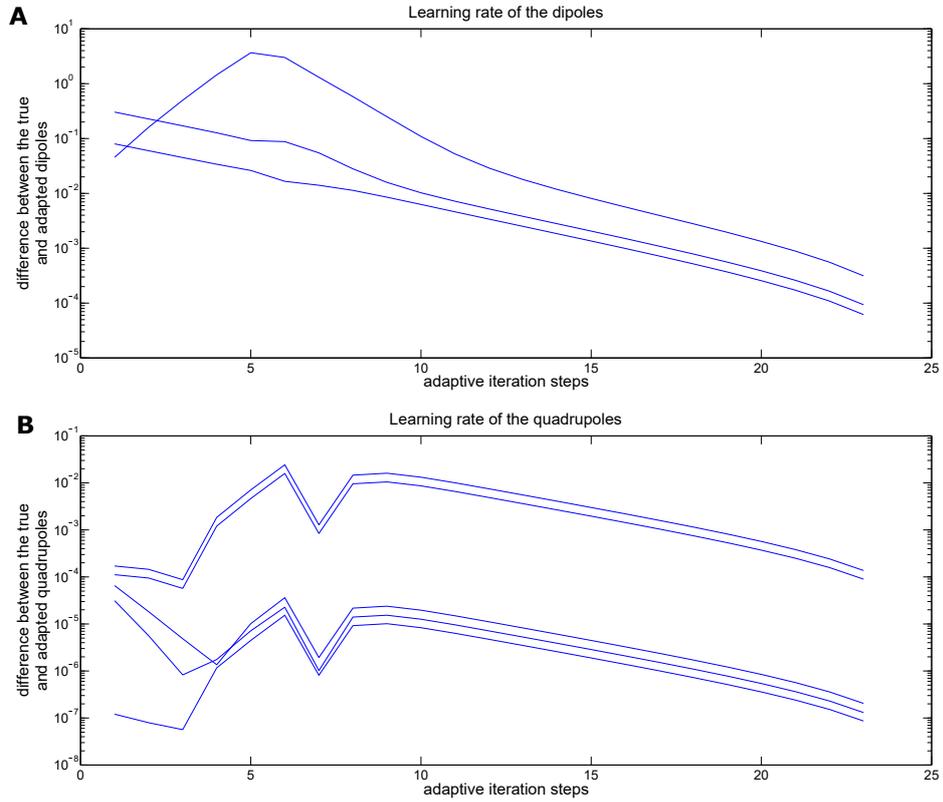
**Figure 3.3:** Subfigures A and B show the adaptation behavior of the estimated dipoles and quadrupoles at one arbitrary trajectory point in the constriction model environment. Note that the scale of the error on the y-axis is logarithmic, and the linear decrease after an onset indicates an exponential error reduction during the adaptation. The dipoles (subfigure A) are more pronounced than the quadrupoles (subfigure B). This is true for both tested environments. Whereas the dipoles are intuitive, they are simply the asymmetry in x, y and z-direction, the quadrupoles represent more complex depth characteristics.

**Following the depth changes with the adaptive MFA**

To test the adaptation behavior of the adaptive MFA the agent flies through the 3D model environment with the constriction. The constriction ensures that the depth changes much over the trajectory. The rotation of the agent was set to zero. Because we want to show how the algorithm adapts to the agent centered depth structure, when the position is changed in the 3D environment. A rotation of the agent does not change the depth structure, it is only rotated from the perpective of the agent. Rotation is important when the self-motion parameters shall be determined in parallel (see section 3.4.3).

In this section, the adaptation is shown by the help of perfect optical flow vectors (computed with the true self-motion parameters and the depth buffer of the rendering engine) to see the adaptation effects in pure form at different adaptation rates. An update of the depth model takes place once at every trajectory point and once at every 15th trajectory point, respectively. Figure 3.4 A and B show a certain dipole and a certain quadrupole ($b_2$ and $c_1$ in equation 3.6 and 3.3). The choosen dipole and quadrupole exhibit the largest changes over the trajectory. (The shown dipole in figure 3.4A is the asymmetry in z-direction. The quadrupoles are more difficult to interpret. The quadrupole shown in figure 3.4B is the quadrupole which reacts mostly to the funnel parts.)

The graphs C until F in figure 3.4 show an exponential adaptation behavior to the changing depth values of the environment. The adaptation error is large when the slope of the depth value change is high. Whereas the values of the dipole and quadrupole (subfigures 3.4A and 3.4B) directly reflect the error for a non-adaptive spherical depth model (where all assumed dipoles and quadrupoles are zero), the adaptation error is smaller by roughly two orders of magnitude, if adaptation takes place at every trajectory point, and smaller by roughly one order of magnitude, if adaptation takes place at only every 15th trajectory point.

### 3.4.3 Comparison with fixed depth model

In this section, the usefulness of adaptation to the depth structure is compared with a fixed-depth model. The error in estimating the self-motion parameters can be subdivided into three parts:

$$
\begin{pmatrix} M^{tt} & M^{tr} \\ M^{rt} & M^{rr} \end{pmatrix} \cdot \begin{pmatrix} \vec{t} \\ \vec{r} \end{pmatrix} = \begin{pmatrix} \langle -\vec{p} \rangle \\ \langle \vec{p} \times \vec{d} \rangle \end{pmatrix}
$$
$$
\hat{M} \cdot \vec{s} = \vec{f}
$$
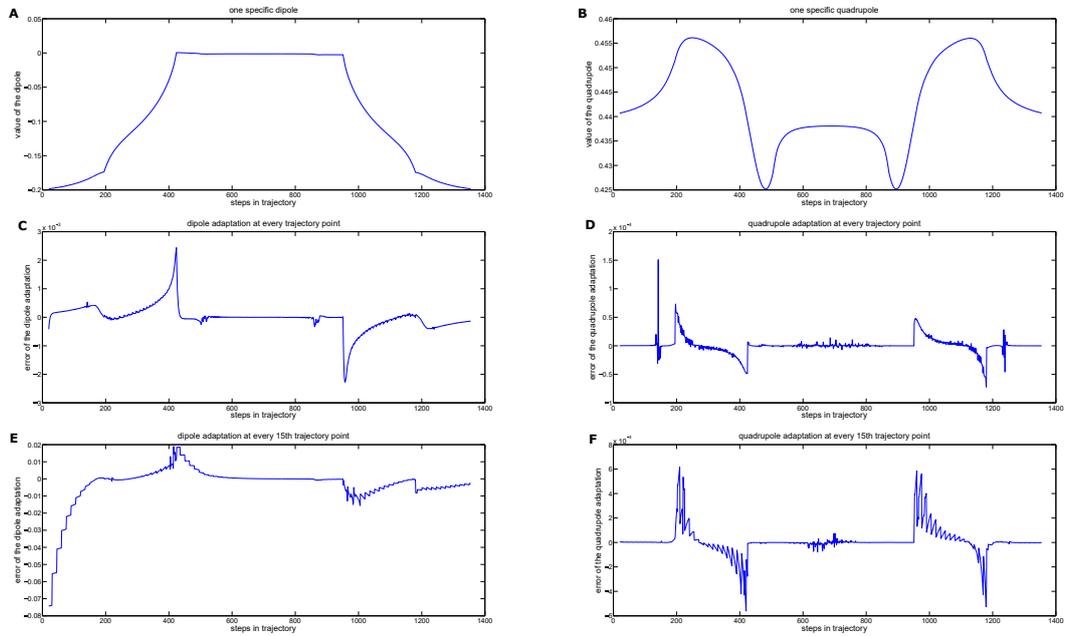
**Figure 3.4:** Subfigures A and B show the dipole and quadrupole which have the largest changes along the trajectory chosen for the constriction model environment. The interpretation of the dipole (subfigure A) is intuitive: it is the asymmetry in the z-direction, which is large at the beginning and end of the trajectory and falls to zero within the constriction. The changes of the quadrupole are much smaller than the changes of the dipole and occur almost exclusively in the narrowing parts of the 3d model. The quadrupoles can be understood with an analogy to physics: Particle physicists use quadrupole magnets to focus charged particle streams. Subfigures C until F show how the depth model adapts to the changing dipole and quadrupole. The adaptation follows the changes more rapidly if the depth model is updated each trajectory step (subfigures C and D) than if its updated every 15th trajectory step (subfigures E and F). The first few adaptation steps are not shown, since they are characterized by much larger errors (see Figure 3.3). Only in the case of the adaptation to the dipole at every 15th trajectory step (Subfigure E) the depth model needs a longer initiation phase until it can follow the changes.

where $\hat{M}$ combines all matrices, $\vec{s}$ stands for the self-motion values and $\vec{f}$ combines the right side of the equation. The above equation is reformulated to:

$$\vec{s} = \hat{M}^{-1} \cdot \vec{f}$$
$$\vec{s}_r + \vec{s}_e = \left( \hat{M}_r^{-1} + \hat{M}_e^{-1} \right) \cdot \left( \vec{f}_r + \vec{f}_e \right)$$
$$\vec{s}_r + \vec{s}_e = \hat{M}_r^{-1} \cdot \vec{f}_r + \hat{M}_r^{-1} \cdot \vec{f}_e + \hat{M}_e^{-1} \cdot \vec{f}_r + \hat{M}_e^{-1} \cdot \vec{f}_e$$
$$\vec{s}_e = \hat{M}_r^{-1} \cdot \vec{f}_e + \hat{M}_e^{-1} \cdot \vec{f}_r + \hat{M}_e^{-1} \cdot \vec{f}_e \tag{3.9}$$

The index $r$ represents the true value and the index $e$ represents the error of the indexed values. The term $\hat{M}_r^{-1} \cdot \vec{f}_e$ with the perfect depth model reflects the direct error of the flow values $\vec{p}$, whereas the term $\hat{M}_e^{-1} \cdot \vec{f}_r$ with the correct optic flow values reflects the error caused by a (slightly) incorrect depth model, the term $\hat{M}_e^{-1} \cdot \vec{f}_e$ is a second-order error term, which can be neglected in all analyzed cases. In the following this differentation makes the error sources clearer.

**A cube as simple 3d model environment**

The first 3D model environment in which self-motion estimation is tested is a simple cube. The cube has a side length of 300 units. The agent flies on a straight trajectory near the bottom of the cube at an altitude of 25 units. This causes a large dipole moment in the z-direction. The trajectory follows a line from $x = 50$ units to $x = -50$ units. The translation of the agent along the x-direction is superimposed by a rotation with a random axis (see section 3.3.2). A rotation is countered by a rotation in the opposite direction in the next trajectory step. We choose a simple kind of translation and rotation of the agent, but we argue (in the discussion) that the adaptive MFA can operate with arbitrary self-motion. To show the advantage of an adaptive model it is important that translation and rotation occur simultaneously. Otherwise, the dipole moments would play no role because the matrix $M^{rt}$ in equation 3.6 has no effect.

The magnitude of the translation and rotation is restricted by the operating range of the Lukas-Kanade detector, which is implemented in a non-pyramidal way. The maximal magnitude of the optical flow caused by the translation matches the magnitude of the optical flow caused by the rotation. In principle, the magnitudes of the translation and rotation can be chosen arbitrarily, if the caused optic flow lies in the operating range of the used optic flow detector. If the magnitudes differ much, one has to consider that the used motion detector leads to an estimation error. Hence, the smaller self-motion component would be estimated worse than the larger component. Here we choose the same magnitude for the self-motion components, because we analyze the relative error caused by the deviation of the depth model ($\hat{M}_e^{-1} \cdot \vec{f}_r$) and the deviation of the flow estimates ($\hat{M}_r^{-1} \cdot \vec{f}_e$), and not the relative error between the self-motion components.

The trajectory chosen for the 3D cube model environment leads to a large dipole moment, whereas the small quadrupole moments play no significant role. Hence, the
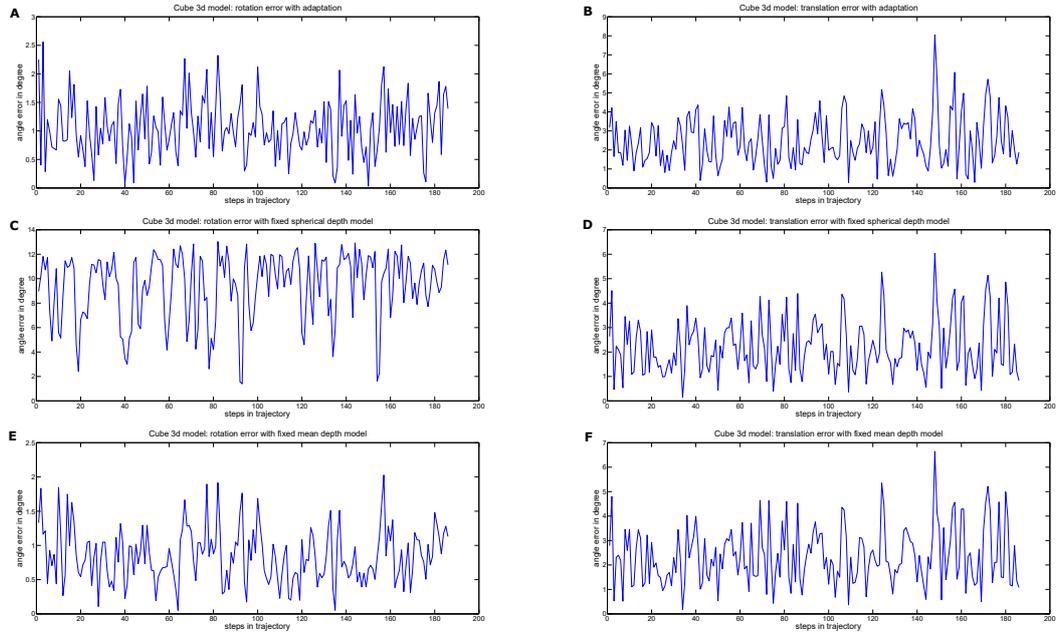
**Figure 3.5:** The self-motion estimation error along a trajectory is given here and later as an angle error for both the translation and rotation axis. The variations in the error of the estimates are the consequence of the detector variance. The subfigures show three different configurations. For each configuration, the rotation error (subfigures A, C, E) and the translation error (subfigures B, D, F) are presented. The first configuration (subfigures A and B) is the adaptive MFA. The subfigures C and D show the error for a fixed spherical depth model and the subfigures E and F show the error for a fixed but not symmetrical depth model. The asymmetrical depth model is the best fit of mean depth model. Because the quadrupoles are negligibly small, the depth model only affects the rotation estimates. Whereas the rotation estimates for the fixed mean depth model are as good as the estimates for the adapting depth model, the rotation estimates for the spherical depth model are significantly worse.

difference in the self-motion estimates appears in the rotation estimates. Figure 3.5 shows the translation and rotation estimates for three different configurations. As a measure of the error the angle between the correct self-motion axis and the estimated self-motion axis is chosen. For a better comparison, the error dimension is the same for translation and rotation estimates. The magnitude of the rotation is not kept in perspective. This error measurement allows an intuitive interpretation of how precise the estimates are.

The subfigures A and B in Figure 3.5 show the estimates of the adaptive MFA. The estimated error for the rotation is, on average, roughly one degree. The estimated translation error is larger (2-3 degrees), because the optical flow of the translation is very small in the upper hemisphere. The second configuration (subfigures C and D) uses the non-adaptive MFA with a spherical depth assumption (all dipoles and quadrupoles are zero). Whereas the translation estimates are not observably affected, the errors of the rotation estimates are one order of magnitude larger (10 degrees in the mean). The third configuration uses a fixed mean depth model. The best fit for a depth model, which does not change, is the depth distribution in the middle of the trajectory. For this mean depth model, the non-adaptive MFA is not significantly worse than the adaptive MFA.

**Constriction 3D model environment**

The 3D constriction model environment is more complex than the cube model environment analyzed before. When flying along the trajectory the depth distribution of the environment changes significantly. The 3D environment consists of five cylindrical elements. The agent starts in a large cylinder (300 units in diameter) flying near the bottom (flight altitude 25 units). This starting configuration resembles the configuration in the 3D cube. After flying 85 units in x-direction the agent reaches a cylindrical section which tapers in its diameter to 50 units. The agent follows the ascent at a constant altitude of 25 units. After the taper, the agent flies through a cylindrical section of 50 units in diameter. Hence, in this section, the ceiling is as far away as the floor (25 units). The whole 3D environment and the trajectory are mirror symmetric in the x-direction, so the agent flies through sections of the same shape after the constriction in reverse order (see figure 3.1).

The rotation of the agent is implemented in the same way as in the cube model environment. It is a random rotation which is followed by a counter-rotation. The magnitude of the optical flow caused by the rotation is in the same range as the maximum magnitude of the optical flow caused by the translation.

Figure 3.6 shows the estimation results for the adaptive MFA. Subfigures A to D show the estimation errors for the terms $\hat{M}_e^{-1} \cdot \vec{f_r}$ and $\hat{M}_r^{-1} \cdot \vec{f_e}$ (equation 3.9) separately. The estimation error caused directly by the error-prone flow estimates ($\hat{M}_r^{-1} \cdot \vec{f_e}$) lies in the range of 1.5 degrees. In the first, second, fourth and fifth section of the 3D environment, the estimation error of the translation is larger than the rotation error, because in these sections the upper hemisphere is further away.
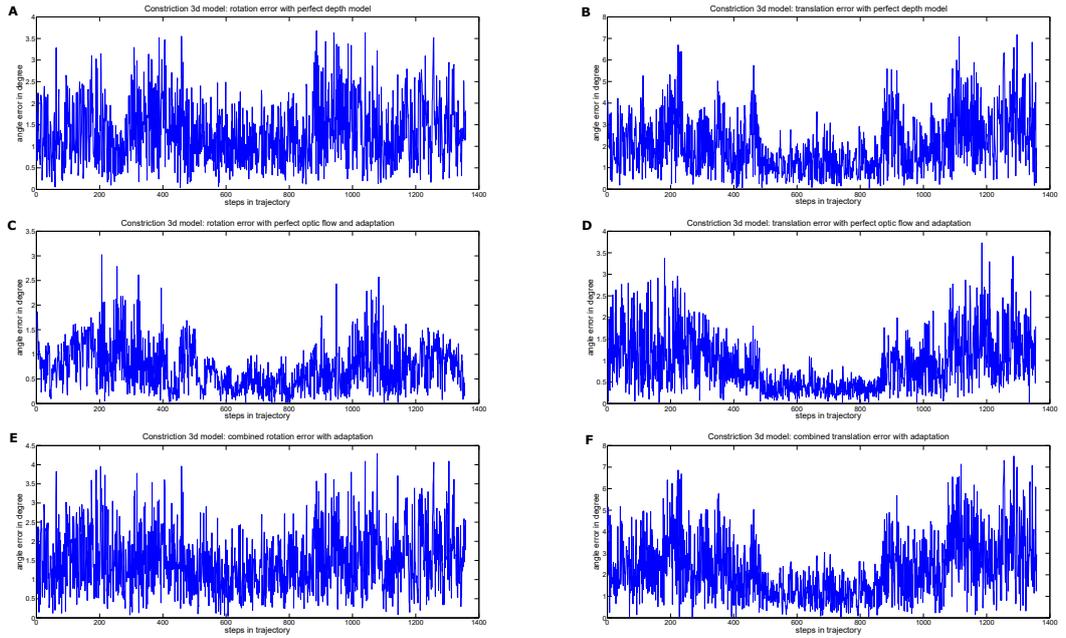
**Figure 3.6:** The plotted self-motion estimation errors show the performance of the adaptive MFA in the constriction 3d environment. Subfigures A and B show the error term $\hat{M}_r^{-1} \cdot \vec{f_e}$ separately. It uses a perfect depth model and presents the contribution of the direct flow error. The variations in this error are caused by the detector variance. In contrast, subfigures C and D show the error term $\hat{M}_e^{-1} \cdot \vec{f_r}$. In this term, the optical flow values are exact, whereas the adapting depth model is error prone. The variations are caused by the superimposed random rotation and indirectly over the equation 3.8 by the detector variance. The combination of the two terms together with the negligible second error term $\hat{M}_e^{-1} \cdot \vec{f_e}$ are shown in the subfigures E and F.

For computing the error of $\hat{M}_e^{-1} \cdot \vec{f_r}$ the correct optical flow values are used (obtained with the help of the trajectory parameters and the depth buffer of the rendering engine). The depth model $\hat{M}_e^{-1}$ is generated by a simulation run with error-prone flow values under the same conditions. Hence, the depth model includes by equation 3.8 the errors of the optical flow $\vec{p_i}$ and the errors of the self-motion $\vec{t}$ and $\vec{r}$ estimated each time one trajectory step before. The estimation of $\vec{t}$ and $\vec{r}$ is computed with the current depth model, so the error in $\hat{M}_e^{-1}$ includes the exponential adaptation rate (see subsection 3.4.2).

The error term $\hat{M}_e^{-1} \cdot \vec{f_r}$ is smaller by roughly a factor of two than the error term $\hat{M}_r^{-1} \cdot \vec{f_e}$ in this configuration, which shows the robustness of the depth model. The subfigures E and F show the overall estimation error $\vec{s_e} = \hat{M}_r^{-1} \cdot \vec{f_e} + \hat{M}_e^{-1} \cdot \vec{f_r} + \hat{M}_e^{-1} \cdot \vec{f_e}$. It is the sum of $\hat{M}_r^{-1} \cdot \vec{f_e}$ and $\hat{M}_e^{-1} \cdot \vec{f_r}$, the quadratic error term $\hat{M}_e^{-1} \cdot \vec{f_e}$ has no significant contribution.

The non-adaptive MFA is tested in two configurations on the same trajectory and optic flow inputs as the adaptive MFA. The first configuration uses a fixed depth model obtained at the beginning of the trajectory. Figure 3.7 A and B show the estimation error in this configuration with perfect flow values ($\hat{M}_e^{-1} \cdot \vec{f_r}$). Again, as in the cube 3D environment, the dipoles play the major role. The rotation error is small at the beginning and end, where the fixed model is tuned at. Within the constriction, the rotation error increases up to 35 degrees due to the false dipole assumption in this section of the 3D environment, whereas the quadrupoles play an observable role only in the narrowing part of the 3d environment.

The results obtained with the second configuration is shown in subfigures C and D. Here a fixed depth model is used which is obtained in the middle of the constriction part. With this depth model, the dipole assumption is worse at the beginning and end of the trajectory, and the rotation error raises up to 18 degrees in these sections.

The error term $\hat{M}_r^{-1} \cdot \vec{f_e}$ with the perfect depth model is the same as in the adaptive case (Figure 3.6 A and B). Figure 3.7 E and F show the combined errors $\vec{s_e} = \hat{M}_r^{-1} \cdot \vec{f_e} + \hat{M}_e^{-1} \cdot \vec{f_r} + \hat{M}_e^{-1} \cdot \vec{f_e}$ for the second configuration. Again the quadratic error $\hat{M}_e^{-1} \cdot \vec{f_e}$ can be neglected due to the small $\vec{f_e}$.
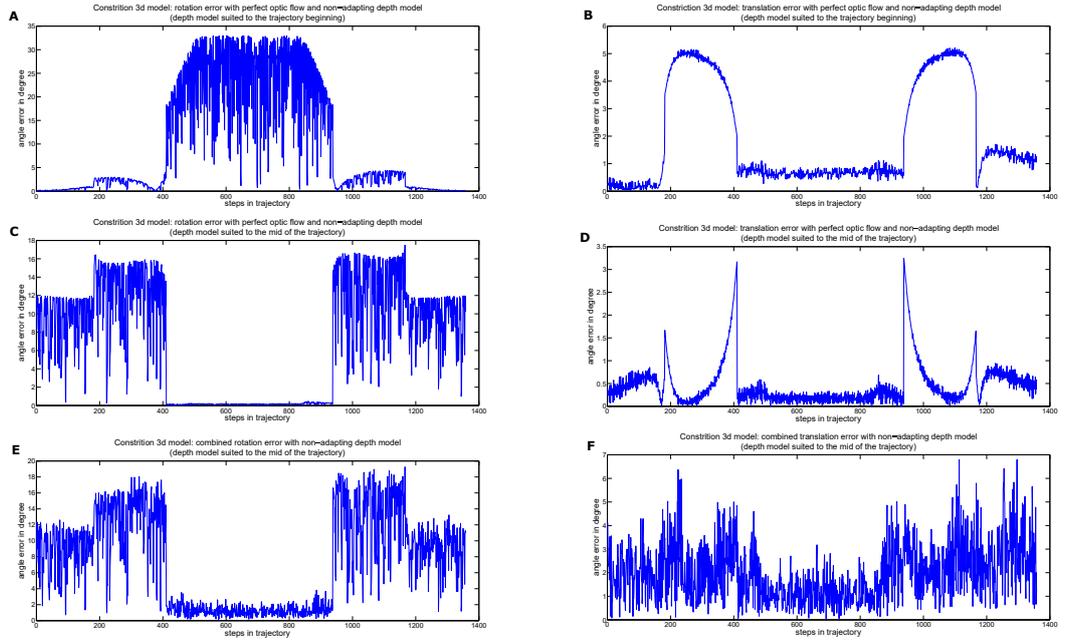
**Figure 3.7:** In the constriction 3d environment the non-adaptive MFA is tested with two different fixed depth models. One is fitted to the depth distribution at the beginning (subfigures A and B) and one uses the depth distribution in the middle of the constriction (subfigures C and D). In parts where the depth model is not tuned to the rotation error large errors my occur (subfigures A and C). In contrast to the cube 3d environment the constriction environment includes parts where the quadrupoles lead to errors which are not totally overlaid by the error caused by the detector variance. These parts are the narrowing parts of the 3d environment (subfigures B and D). Subfigures A until D show the self-motion estimation error with perfect flow values $(\hat{M}_e^{-1} \cdot \vec{f}_r)$. The overall error $\vec{s}_e$ is shown in subfigures E and F for the second fixed depth model.

## 3.5 Discussion

This study analyzes the matched filter approach of self-motion estimation as proposed in our previous paper [89] in two variants: one with a fixed depth model and other with an adapting depth model. It has been shown mathematically that the self-motion estimation error can be divided into three parts (see equation 3.9). An error that is caused directly by the motion detector variance ($\hat{M}_r^{-1} \cdot \vec{f}_e$), an error that is caused by a (slightly) incorrect depth model ($\hat{M}_e^{-1} \cdot \vec{f}_r$) and a second-order error term ($\hat{M}_e^{-1} \cdot \vec{f}_e$), which combines these two errors. The second-order error term can be neglected in all analysed cases, because both error contributions are smaller than one. This leads to a second-error term, which is much smaller than the first-order error terms. This decomposition of the error contributions shows that the performance of the non-adapting and the adapting matched filter approach can be analyzed independly from the used motion detector. However, the variance of the used motion detector should be small to ensure that the contribution of the depth model error is not fully overlaid by the error resulting from the motion detector variance.

It has been shown that by using the Lukas-Kanade motion detector the error contribution of the adapting depth model ($\hat{M}_e^{-1} \cdot \vec{f}_r$) is much smaller than the direct detector variance error ($\hat{M}_r^{-1} \cdot \vec{f}_e$). This is astonishing, because the adapting depth model error is not only affected by the adaptation rate, but also by the motion detector variance and the slightly incorrectly estimated self-motion components. This observation is likely to hold also for other motion detector models with a larger variance, because both first-order errors, the direct motion detector error and the depth model error, are affected similarly by the motion detector variance. Both first-order error terms do not rely on a single estimated motion vector, but on a summation over all estimated motion vectors.

Whereas the adaptive MFA adapts in every tested 3D environment, the non-adapting MFA uses a fixed depth model in all 3D environments. The non-adapting depth model is the best fit to the 3D layout of the respective environment at a certain point on the trajectory. In the tested cube 3D environment a non-adapting depth model can be found, which gives similar results as the adapting depth model. In the 3D environment with the constriction we use two non-adapting depth models. Both are suitable for a certain section of the trajectory, but get worse for other sections. It has been shown that the error of the non-adapting depth model is very large in those sections of the environment where the non-adapting depth model is not tuned to. These errors reach dimensions far above the direct motion detector variance error.

The chosen trajectories for the cube environment and the environment with the constriction look very special and one can ask, whether the results hold also for arbitrary trajectories. In our model simulations the trajectories were chosen as simple as possible in order for the induced optic flow to lie in the detectable range of the used motion detector (see section 3.3.3). The floor of the 3D environment has the smallest distance to the agent for the chosen trajectories. Hence, we made sure by controlling the fight

altitude that the translatory optic flow in the corresponding part of the field of view was in the detectable range of the motion detector. The flight altitude was set to a value to ensure that the optic flow induced by the floor is about half the maximal optic flow that can be detected. Hence, a rotational optic flow of the same magnitude and direction can be added to this translational flow component without exceeding the detectable range. Both the magnitude of the rotation as well as the rotation axes are random. The magnitude of the random rotation never exceeds a value that the caused rotational optic flow exceed the maximal translational optic flow. Moreover, the minimal magnitude of the rotation is never so small, that the motion detector cannot detect this self-motion component.

Theoretically, the used trajectories can be more complex if the caused optic flow lies nonetheless in the detectable range of the motion detector. The simplicity of the chosen trajectories -the translation is almost a straight line in world coordinates- does not simplify the self-motion estimation, because the agent does not remember the previous self-motion components. At every point on the trajectory, the agent does not have an initial guess for the current self-motion based on previous self-motion. We rely only on the assumption that the depth model obtained on preceding trajectory points is still a sufficiently good estimate for the depth model at the current trajectory point. Hence, we can assume that the analyzed adaptive MFA functions with arbitrary trajectories, if the caused optic flow is in the detectable range of the used motion detector.

The range of the detectable optic flow for the Lukas-Kanade motion detector can be extended to arbitrary values by implementing it pyramidally [93]. It is very likely that the analyzed adaptive MFA can operate with arbitrary optical flow fields caused by self-motion, because the algorithm does not rely on assumptions about the optical flow field. A problem arises only, if one motion component is much smaller than the other. Because every motion detector is error prone, the smaller self-motion component can be totally lost due to this error.

The restricted operating range of the motion detector is reminiscent of contraints imposed on biological motion detection. In flying insects, like bees and flies, the optic flow is detected by comparing the light intensities projected on neighboring ommatidia [4]. Besides this space constraint, the processing speed of processing of changing light intensities is limited depending on the time constants in the visual pathway [28]. These constraints lead to a specific range of detectable optic flow for biological motion detectors.

Flying insects have been concluded to actively control the detected optic flow during flight by adjusting their flight speed [2, 54]. In environments where the distances to structures in the environment are small the insect flies slower than in environments where the distances are larger, thereby keeping the optic flow in a range where the movement detector responses increase with increasing velocity.. The detected optic flow is a good indicator for the time to contact to objects [40], because the optic flow is roughly the product between flight speed and environmental distance.

Another aspect concerning the generalizability of the shown results are the artificial

environments used in our simulations. These environments exhibit no depth discontinuities as natural cluttered environments do. Natural cluttered environments have depth discontinuities at the edges of objects and contain regions that are visible at some points on the trajectory, but are hidden behind objects at others. However, natural cluttered environments are not arbitrarily discontinuous. Concerning the entire field of view, there are more continuous parts than discontinuous parts, which leads to more reliable optic flow vectors than unreliable ones. Hence, when summating linearly over all flow vectors the discontinuoties may play an only minor role. How does the depth model of the adaptive MFA behave in cluttered environments? The depth model only takes into account the lowest frequency spherical harmonics up to the order of two. Whereas higher orders of the spherical harmonics represent more and more details of the depth structure of the environment. The used lowest frequencies represent only global characteristics like the asymmetry between the upper and lower hemisphere. Hence, we can assume, that these low frequency harmonics are not strongly affected by a natural cluttered environment, so that the assumption may still hold that the depth model is a good enough approximation over several trajectory steps.

A field of view covering the whole sphere, was chosen due to the simple depth model that is obtained on this basis. Whereas, the coupling matrix (matrix $M$ in equation 3.1) only differs from the unit matrix by the dipole and quadrupoles of the nearnesses of the environment in this case, the coupling matrix will be more complex for a restricted field of view. The higher frequencies of the spherical harmonics only drop completely, if one summates over the whole sphere. Besides higher frequencies of spherical harmonics, the coupling matrix would also involve correction terms, which are not caused by the depth structure of the environment. These correction terms arise from the coupling between the self-motion components in a restricted field of view. With respect to a small field of view, one cannot distinguish, whether the detected optic flow is caused by a rotation or a translation. This ambiguity increases for a more and more restricted field of view, up to a point where the coupling matrix is no longer invertible. Hence, the magnitude of the determinant of the coupling matrix is a good measurement how strong the coupling between the self-motion components is. We tested the adaptive MFA with a spherical field of view because this comes close to the visual field of most flying insects. A spherical field of view is not only relevant for the biology of flying insects, but also for technical systems for self-motion estimation. These could benefit from the optimal case of a spherical field view, realized, for example, by omnidirectional cameras [90].

In our quantitative analysis of the adaptive MFA, a gradient detector, i.e. the Lukas-Kanade detector, was used to obtain an optical flow field from rendered image sequences. In flying insects, like flies and bees, a different detector has been concluded to estimate local motion. This detector is a correlation detector and was first introduced by Reichardt and Hassenstein [44]. It correlates the spatiotemporal image intensities at two neighboring points A and B in the retinal image. This detector conists of two mirror-symmetrical subunits, the half-detectors, which contribute to its output signal with an opposite sign.

In the simplest version of this detector, one half-detector low-pass filters the signal obtained at position A and multiplies this filtered signal with the signal at position B that has not been low-pass filtered. In a certain velocity range the multiplied response increases the faster the local image intensities move from A to B. The other half-detector inverts the role of the positions A and B to determine motion in the opposite direction. The local overall output of the Reichardt-Hassenstein detector is given by the difference between the two half-detector outputs.

The output of the Reichardt Hassenstein detector depends not only on the velocity of the image, but it depends also on the pattern of the image. This pattern dependence may be large and may predominate the dependence on velocity in the standard implementations of the detector. Usually, this pattern dependence is seen as pattern noise and several elaborations of the Reichardt Hassenstein detector were proposed which try to reduce this dependence [25, 76]. The classification of the pattern dependence as pattern noise is not the only interpretation. Some authors argue that the pattern dependence of the Reichardt-Hassenstein detector may be functional and to provide important information about the environment [73, 49].

The Reichardt-Hassenstein detector was tested as we tested the Lukas-Kanade detector in both 3D environments, but we show only the results for the Lukas-Kanade detector. We used a standard implementation of the Reichardt-Hassenstein detector which led to the problem that the direct detector variance error ($\hat{M}_r^{-1} \cdot \vec{f_e}$) was much larger than the depth model error ($\hat{M}_e^{-1} \cdot \vec{f_r}$) for both, the adaptive and non-adaptive MFA. Hence, when combining these first-order error terms the advantage of the adaptive version was not visible. Because it was beyond the scope of this study to test different variants of the Reichardt Hassenstein detector to find a variant, which is suitible for self-motion estimation, we decided to use only the Lukas-Kanade detector. The possibility to separate the error contributions in additive terms (see equation 3.9) allows to compare the adaptive and non-adaptive MFA solely in view of their depth model error. This comparison can be done with every motion detector, particularly with the Lukas-Kanade detector. Nonetheless, the Lukas-Kanade detector shares some properties with the Reichardt-Hassenstein detector. Both can be applied to images where the information of every pixel is used to construct a dense optic flow field and both have a restricted window to detect pixel velocities.

Flying insects like bees and flies perform an active flight strategy to separate translatory and rotatory flight components. During so-called intersaccades, they try to avoid any rotation. These phases are intersected by quick rotations, the saccades [81, 7]. This flight strategy simplifies the computational effort of the insect. Only the translational optic flow depends on the depth structure of the environment. Hence the insect can obtain depth information about the environment during the intersaccades without an overlaid rotational optic flow.

The separation of the flight components is not perfect. Due to the inertia, there is a small translatory component during the saccades and rotations cannot be avoided totally

during the intersaccades. If the separation were perfect the largest error term of the MFA would drop. The matrix $M^{tr}$ would then play no role and the dipoles of the depth distribution do not contribute to the depth model error (see equation 3.6).

The adaptive MFA is bio-inspired and solves the self-motion estimation problem for arbitrary self-motion, if the flow field is given by a motion detector with sufficient accuracy. The adaptive model can, thus, be transferred to artificial agents with arbitrary flight strategies. The results show that despite of the large number of optic flow vectors an error of a few degrees remains for the self-motion components, mainly due to the detector variance. Nonetheless, the estimation results are robust. The few optic flow vectors which deviate much from the correct vector, for example at depth discontinuities, do not contribute much to the overall estimate. In any case, the estimates of self-motion based on the approach presented here can be used as a first approximation to more exact self-motion estimation methods like feature tracking (see [43]). With a robust first approximation, it might then be easier to sort out false tracked features.

Potters and Bialek [77] have shown that the estimation of optic flow vectors needs a non-linearity within the computation: With solely linear operations one cannot obtain a motion dependent signal from successive images. The non-linearity within the Reichardt-Hassenstein detector is the multiplicative stage. It has been shown that this multiplication is the lowest order of possible non-linearities which is optimal for motion detection under a low signal to noise ratio. In the simulation of the 3D environments presented here there is no noise on the rendered images, hence, the Reichardt-Hassenstein detector could not benefit from this advantage. The non-linearity of the Lukas-Kanade detector orientates on the definition of motion: Velocity formally corresponds to the distance moved per time. So, expressed in time and space derivatives it devides the time derivative of the image through the space derivative.

Whereas the non-adaptive MFA is not a good estimator of self-motion in changing environments it principally shows that self-motion estimation is possible with solely linear operations. Hence, we discuss linearizations of the adaptive MFA. The adaptive MFA include two non-linearities: The inversion of the coupling matrix and the denominator in the estimation of the nearnesses. The non-adaptive MFA has also to invert the coupling matrix, but the matrix is constant in this approach. This means that this matrix needs to be inverted only once and not during the run-time of the algorithm. In the adaptive MFA, the coupling matrix has to be inverted after every update of the depth model. As shown in [89] the inverse of the coupling matrix M (equation 3.1) can be simplified. As seen in section 3.3.4 the matrix $M^{tr}$ is zero for a spherical field of view, hence, only the matrix $M^{tt}$ must be effectively inverted. The matrix $M^{tt}$ is equivalent to the sum of the unit matrix $I$ and a matrix $A$ which contains only quadrupoles which are usually small. This kind of matrix inverse can be linearized: $(I - A)^{-1} \approx I + A$.

It can be assumed that this linearization of the inversion is a good linearization because the precondition that $A$ is much smaller than the unit matrix $I$ is fulfilled. This is not obvious for a linearization of the nearness estimation:

$$\mu_i = -\frac{\vec{t} \cdot \left( \vec{p}_i - \vec{d}_i \times \vec{r} \right)}{1 - \left( \vec{t} \cdot \vec{d}_i \right)^2} \tag{3.10}$$

$$\frac{1}{1 - \left( \vec{t} \cdot \vec{d}_i \right)^2} \approx 1 + \left( \vec{t} \cdot \vec{d}_i \right)^2 \approx 1 \tag{3.11}$$

Because $\left( \vec{t} \cdot \vec{d}_i \right)^2$ is not necessarily much smaller than one. When the viewing direction $\vec{d}_i$ approaches the direction of translation $\vec{t}$ the denominator can get arbitrarily small and then leads to a large value for the nearness. Theoretically the optic flow $\vec{p}_i$ approaches zero when the denominator gets smaller, but the estimated optic flow is error prone. Hence, when the unknown true optic flow gets smaller the impact of this error increases [6]. When a flying insect tries to learn a goal location it shows flight segments where it reveals considerable sideways movements relative to the goal [23, 24]. Hence, the insect uses the most reliable and largest optic flow to learn a goal location.

It was tested that adding a small constant to the denominator gives the optic flow vectors less weight, when the denominator gets smaller (data not shown). A constant of up to a value of 0.1 can be added without worsening the overall result. Hence, one can assume that even the linearization given by equation 3.11 still leads to good results. Overall, this short discussion of possible linearizations show that the adaptive MFA could be realized by solely linear operations.

The adaptive MFA determines coefficients of spherical harmonic functions based on the nearnesses, which might appear to be complicated. However, nine artificial neurons can be defined, one for each spherical harmonic up to the order of two, which compute these coefficients. Nine artificial neurons are the minimum for this task. Each artificial neuron has to cover the whole viewing field as the spherical harmonics do. Since the determination of each coefficient is linear one can divide the whole viewing field in smaller parts and summate the results. This provides the possibility that each of the hypothetical nine artificial neurons with spherical receptive fields might be represented by a number of neurons each with a smaller receptive field.

The determination of each coefficient of the spherical harmonics is a weighted sum over the nearness values. This weighted sum has constant weights because the value of one specific spherical harmonic only depends on the viewing direction. Hence, each nearness value is multiplied by a different but constant value. This kind of weighted sum satisfies the definition of an artificial neuron. In view of the biological plausibility of the adaptive MFA, one can postulate specific neurons in the visual pathway of flying insects, which have a sensitivity structure over the receptive field, which resembles the values of the nine spherical harmonic functions.

# 4 Resume and Outlook

## 4.1 What is gained theoretically and practically by this doctoral study?

**Theoretically**  Two approaches to self-motion estimation, which are well known from the science literature, the Koenderink van Doorn (KvD) algorithm and the matched filter approach (MFA) are analyzed in this doctoral study. The two approaches originate from different fields. Whereas the MFA should explain the ability of flying insects to find back to a location which was visited before by estimating the self-motion during the whole flight with the help of specific filters, the KvD algorithm shows the relationship between optical flow vectors and self-motion and provides an algorithm to compute the former from the latter. However, both approaches try to estimate the same: The self-motion parameters from given flow vectors. So, the defined filters of the MFA must be comparable to the vector equations of the KvD algorithm. This doctoral study shows that these two approaches have not only something in common, but that they are identical under one specific condition. So, the progress which was made here, is, that from a theoretical viewpoint, one has not two methods but just one. This helps analyzing and modeling certain of tangential cells of insects, which are said to work as such self-motion filters by providing a more general view point on this problem.

The mentioned specific condition under which the two methods are identical pertain to the question of whether the self-motion estimation can be regarded as a linear or non-linear estimation problem. If the depth distribution of the environment is known, the problem is linear. Then the two approaches must be identical because both are optimal in some way and there exists only one optimal solution for a linear estimation problem. This study took the direct way to show the identity: The equations from the MFA are transformed to the equations of the KvD algorithm. Hence, one has not only a theoretical argument from linear estimation theory but additionally a concrete transformation of one approach into the other.

After identifying the common linear core of both approaches to self-motion estimation my mathematical analysis considers the non-linear estimation problem, where the depth distribution is not assumed to be known. This non-linear problem is only considered for the KvD algorithm. The original MFA uses a evolutionary learned and hard-wired depth distribution. I could show, that the KvD algorithm, even if dealing with a non-linear estimation problem, uses the methods of linear estimation theory to obtain self-motion

estimation equations. This might lead to non-optimal solutions, and this doctoral study shows that this is exactly the case. When inserting the solutions of the first iteration in the next a non-vanishing error term remains. This error term does not vanish even in the limes of infinite given optic flow vectors. Hence, the original KvD algorithm has a bias. In this doctoral study a slightly different version of the equations of the KvD is given, which seems to be optimal. This is shown by a numerical simulation in which the number of the error-prone flow vectors reaches higher and higher values. From a mathematical viewpoint, this kind of simulation cannot show the optimality of the alternative version of the KvD algorithm: One needs a mathematical proof. From this study, one can only say that the original KvD algorithm is not optimal and one has a good candidate for an optimal non-linear estimator.

The optical flow field can only be obtained within the field of view of a technical or biological agent. In a small region of the field of view, an optic flow field caused by a translation is indistinguishable from a flow field caused by a rotation of the agent, which can be seen by the invertibility of the coupling matrix (see Fig.4.1). The larger the field of view the better the self-motion components can be distinguished. Hence, the optimal case is a field of view which covers the whole sphere. For this optimal case, I show in this doctoral study that the depth distribution of the environment can be represented by nine specific spherical harmonic functions without losing information for self-motion estimation. Because a spherical field of view is not just a special case, but the optimal case, this finding shows a deep mathematical relationship between the characteristics of spherical harmonic functions and the problem of self-motion estimation.

Although, the biological reasons why flying insects have nearly a spherical field of view might be manifold, the field of view is optimal for self-motion estimation. And the depth dependence of the self-motion estimation problem simplifies for this condition to the dependence of nine depth parameters.

**Practically**   Most methods of self-motion or camera displacement estimation rely on only two images at two different points in time. This is true for the original KvD algorithm and also for the original MFA. Although the MFA was designed to estimate the self-motion along a trajectory of an agent, no information of the previous steps is used for the current estimation. More technical methods like the eight-point algorithm [43] do not work with optical flow vectors but rather with displacements of feature points from one image to the next. If it is the task to estimate the relative position between two cameras which are fixed in the setup it is the right way to consider only two images. However, if one tries to estimate the self-motion along a trajectory of an agent one has the possibility to use the obtained information during the self-motion estimation for several steps.

Which information can this be? We assume that in the general case at every point, the agent is allowed to move in an arbitrary way and independently from the previous self-motion step. Moreover, we assume small self-motion steps or a high rate of incomming
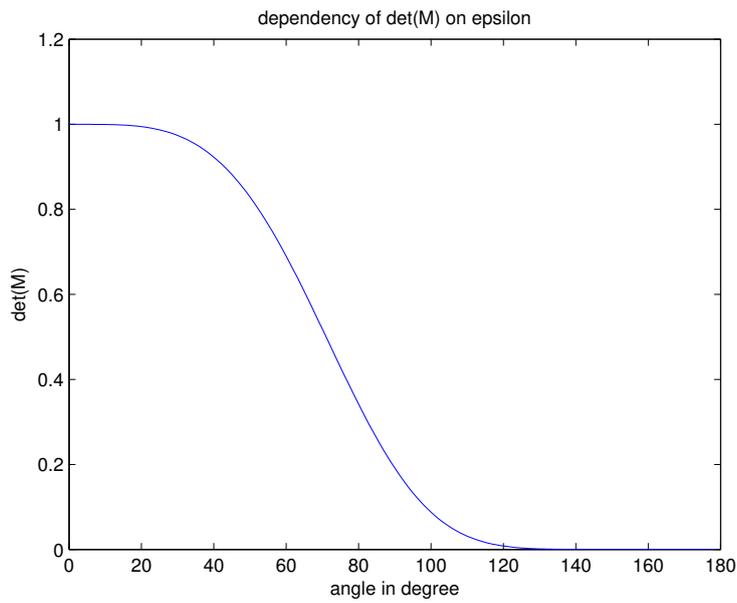
**Figure 4.1:** As a measurement of the singularity of the coupling matrix the value of the determinant is shown. The increasing restriction of the field of view is described by a single angle epsilon. The angle epsilon specifies a circular part which is excluded from the viewing field. Ninety degrees of epsilon describe the half viewing sphere. Up to thirty degrees the determinant is still nearly one.

images, the depth distribution of the environment does not change much between steps. In the strict sense, the environmental depth distribution relative to the agent changes only due to translations of the agent. Rotations of the agent lead only to a rotated depth distribution relative to the agent.

In this doctoral study, advantage is taken of the possibility of using and updating a depth distribution of the environment over several time steps. By doing so, the advantages of the original KvD algorithm and the original MFA are combined: The non-adaptiveness of the original MFA leads to large errors if the depth distribution of the environment changes much. The original KvD algorithm is no longer an iterative method when it is transformed to the adaptive MFA. Hence, the adaptive MFA is a logical enhancement of the idea, that some tangential cells work as self-motion filters.

Several methods, such as the eight-point algorithm [43], estimate self-motion or displacements of a camera by tracking features and finding point correspondences in this way. Point correspondences can be seen as optical flow vectors and vice versa if the distance between the points is not too large. These feature tracking methods could profit from the adaptive MFA by providing a first robust guess for the self-motion parameters. The adaptive MFA (as the non-adaptive MFA) uses the whole visible information to construct a dense optical flow field, which is converted to the estimation of self-motion parameters in a way that the errors in computing the flow vectors are minimized in an optimal way (see Fig. 2.1). Whereas the non-adaptive MFA could not handle changing environments, the adaptive MFA adapts to the environment in a way that the error contribution of the depth model is smaller than the error caused by slightly misjudged flow vectors.

This first robust guess from the adaptive MFA could help to sort out falsely tracked features. Whereas it is more precise to figure out point correspondences as computing the optical flow by gradient methods because the self-motion can only be estimated up to a confidence level of a few degrees in all angles with the Lukas-Kanade detectors as optical flow detector. However, one has a confidence level for the adaptive MFA, whereas it is hard to say how falsely tracked features contribute to the overall error of a method like the eight-point algorithm [72].

The computational effort of the adaptive MFA seems to be large. But in its core, it uses weighted sums, which can be represented by artificial neurons. For this kind of computing elements, there are special hardware in development [39] which are designed to accelerate the computation of artificial neuronal networks. With this kind of hardware on board, it is imaginable that even small flying artificial agents could profit from the computation underlying the adaptive MFA.

## 4.2 Open Questions

It was shown in this doctoral study that for a spherical field of view the depth distribution of the environment in relation to the agent can be expressed by nine spherical harmonic functions, which are the functions of the lowest order, without losing any information for self-motion estimation. Because a spherical field of view is not only a special case of a viewing field but it is the optimal case one can conclude that there is some relationship between self-motion estimation and the description of the environment by spherical harmonic functions. Do these functions play any role for a restricted field of view? This case was not analyzed in this study and is still an open question.

Spherical harmonic functions are defined on the whole sphere. Hence, a restricted field of view requires a more general definition of these functions on arbitrary sections of the sphere. This was done partly by Differt [22]. One open question in this context is: If one regards sections of the sphere as viewing fields, are the lowest orders of these functions sufficient to describe the environmental depth distributions, or are also higher orders relevant? This problem arises, if the spherical harmonic functions are no longer orthogonal to each other in the case that only parts of the sphere are taken into account.

Another open question is whether the given modified version of the Koenderink van Doorn algorithm is the/an optimal solution of the non-linear problem of self-motion estimation if the depth distribution is not given in any way? A mathematical proof to decide this question was not attempted in this study. Hence, it can not be assessed if this proof is trivial or hard to obtain. It can only be concluded, that this proof would be a topic of non-linear estimation theory, because there are products between parameters, which have to be estimated.

Ideally, the simulation results should have been obtained with both the Lukas-Kanade detector and the Reichardt-Hassenstein detector as optical flow detectors. However, the Reichardt-Hassenstein detector leads to problems when using it for self-motion estimation. Under the conditions the self-motion methods are tested here, the Reichardt-Hassenstein detector leads to errors in the self-motion estimation angle of several tenths of degrees (data not shown).

For this analysis I used a variant of the detector with a temporal high-pass filter in its input lines. The temporal high-pass filter increases the signal to pattern-noise ratio by filtering out the mean intensity of the input pattern. Apart from this, the basic design of the detector was not changed yet. The detector shows these large errors if (A) the depth distribution of the environment is highly asymmetrical and (B) the translation and rotation of the agent have to be estimated simultaneously. If one condition, A or B, is not fulfilled, the self-motion estimation error due to the Reichardt-Hassenstein detector does not show these high errors of several tenths of degrees. The self-motion estimation error is comparatively small if the depth distribution is symmetrical or if the translation and rotation is estimated separately.

Two aspects were tested to get better results with this detector. The first was to add a

contrast normalization stage to the detector, because the Reichardt-Hassenstein detector depends strongly on the pattern of the moving images and, especially, on the contrast. The output of the Reichardt-Hassenstein detector is nearly proportional to the local contrast [82], hence, the output was divided through the local contrast. But, the used random textures with their $1/f$ spectrum have nearly the same local contrast at every position in the image. Hence, it is not surprising that the contrast normalization does not improve the overall result.

The second test was to increase the number of Reichardt-Hassenstein detectors by the factor of nine by using a grid of nine detectors at every position where previously was one detector. This was done parallel with a resolution increase of the pattern. If the detector was a detector with Gaussian noise, the error has to improve by the factor of $\sqrt{9} = 3$, if the pattern-noise is assumed to be independent at neighboring positions in the pattern. But this procedure did not improve the performance of the Reichardt-Hassenstein detector for self-motion estimation. Therefore, I decided to use only the Lukas-Kanade detector to show the advantage of the adaptive MFA over the non-adaptive MFA, because the advantage does not depend on the motion detector (see section 3.4.3). The questions which are left are: Can the Reichardt-Hassenstein detector be elaborated in a way that it makes possible to estimate self-motion under the conditions A and B with a useful degree of estimation error? Does the Reichardt-Hassenstein detector turn over in a Gaussian detector for a certain elaboration?

Another open question concerns the existence of depth-sensitive neurons in motion computation of flying insects. This question was asked several times during this study and is listed here for completeness. One can predict nine artificial neurons one for each of the nine spherical harmonic functions. They are defined over the whole sphere and have constant weights, which are given by the value of a specific harmonic function in a specific viewing direction. Because the adaptive MFA consists of two subsequent linear operations, summing the optical flow values and multiplication of this result with the inverse of the coupling matrix (see equation 3.9), it is yet not clear at which point the adaptation should take place. It can take place (1) at the level of tangential neurons or (2) in a network postsynaptic to them. In case (2) one would search for neurons in the visual pathway, which are solely sensitive to depth values. These neurons would have a sensitivity structure for depth values, which resembles a specific spherical harmonic function. In case (1) the tangential neurons, which are assumed to act as self-motion filters, would adapt to the depth structure. In this case, one had to analyse the adaptive behavior of these neurons in relation to varying depth values.

# Bibliography

[1] Emily Baird, Torill Kornfeldt, and Marie Dacke. "Minimum viewing angle for visually guided ground speed control in bumblebees". In: *The Journal of experimental biology* 213.10 (2010), pp. 1625–1632.

[2] Emily Baird et al. "Visual control of flight speed in honeybees". In: *Journal of Experimental Biology* 208.20 (2005), pp. 3895–3905.

[3] Simon Baker and Iain Matthews. "Lucas-Kanade 20 Years On: A Unifying Framework". In: *International Journal of Computer Vision* 56 (2004), pp. 221–255. ISSN: 09205691. DOI: 10.1023/B:VISI.0000011205.11775.fd.

[4] DGM Beersma, DG Stavenga, and JW Kuiper. "Retinal lattice, visual field and binocularities in flies". In: *Journal of comparative physiology* 119.3 (1977), pp. 207–220.

[5] John A Bender and Michael H Dickinson. "Visual stimulation of saccades in magnetically tethered Drosophila". In: *Journal of Experimental Biology* 209.16 (2006), pp. 3170–3182.

[6] Olivier JN Bertrand, Jens P Lindemann, and Martin Egelhaaf. "A bio-inspired collision avoidance model based on spatial information derived from motion detectors leads to common routes". In: *PLoS computational biology* 11.11 (2015), e1004339.

[7] Norbert Boeddeker et al. "The fine structure of honeybee head and body yaw movements in a homing task". In: *Proceedings of the Royal Society B: Biological Sciences* 277.1689 (2010), pp. 1899–1906.

[8] A Borst and M Egelhaaf. "Detecting visual motion: theory and models." In: *Reviews of oculomotor research* 5 (Jan. 1993), pp. 3–27. ISSN: 0168-8375. URL: http://www.ncbi.nlm.nih.gov/pubmed/8420555.

[9] A Borst and J Haag. "Neural networks in the cockpit of the fly." In: *Journal of comparative physiology. A, Neuroethology, sensory, neural, and behavioral physiology* 188.6 (July 2002), pp. 419–37. ISSN: 0340-7594. DOI: 10.1007/s00359-002-0316-8. URL: http://www.ncbi.nlm.nih.gov/pubmed/12122462.

[10] Alexander Borst. "Fly visual course control: behaviour, algorithms and circuits". In: *Nature Reviews Neuroscience* 15.9 (2014), p. 590.

[11]   Alexander Borst, Martin Egelhaaf, and Jürgen Haag. "Mechanisms of dendritic integration underlying gain control in fly motion-sensitive interneurons". In: *Journal of computational neuroscience* 2.1 (1995), pp. 5–18.

[12]   Alexander Borst, Carolina Reisenman, and Juergen Haag. "Adaptation of response transients in fly motion vision. II: Model studies". In: *Vision research* 43.11 (2003), pp. 1311–1324.

[13]   Alexander Borst and Franz Weber. "Neural action fields for optic flow based navigation: a simulation study of the fly lobula plate network." In: *PloS one* 6.1 (Jan. 2011), e16303. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0016303`. URL: `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3031557\&tool=pmcentrez\&rendertype=abstract`.

[14]   Elke Braun, Bart Geurten, and Martin Egelhaaf. "Identifying prototypical components in behaviour using clustering algorithms". In: *PloS one* 5.2 (2010), e9361.

[15]   Elke Braun et al. "Prototypical components of honeybee homing flight behavior depend on the visual appearance of objects surrounding the goal". In: *Frontiers in behavioral neuroscience* 6 (2012).

[16]   Russell SA Brinkworth and David C O'Carroll. "Robust models for optic flow coding in natural scenes inspired by insect biology". In: *PLoS computational biology* 5.11 (2009), e1000555.

[17]   Kenneth H Britten. "Mechanisms of self-motion perception." In: *Annual Review of Neuroscience* 31 (2008), pp. 389–410.

[18]   Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. "Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods". In: *International journal of computer vision* 61.3 (2005), pp. 211–231.

[19]   Aihua Chen, Gregory C DeAngelis, and Dora E Angelaki. "Macaque parieto-insular vestibular cortex: responses to self-motion and optic flow." In: *Journal of Neuroscience* 30 (2010), pp. 3022–3042.

[20]   John S Chipman. "Gauss-Markov Theorem". In: *International Encyclopedia of Statistical Science*. Springer, 2011, pp. 577–582.

[21]   HJ Dahmen, MO Franz, and HG Krapp. "Extracting egomotion from optic flow: limits of accuracy and neural matched filters". In: *Motion vision* (2001). URL: `http://link.springer.com/chapter/10.1007/978-3-642-56550-2\_8`.

[22]   Dario Differt. "Holistic methods for visual navigation of mobile robots in outdoor environments". In: (2017).

[23]   Laura Dittmar et al. "Goal seeking in honeybees: matching of optic flow snapshots?" In: *Journal of Experimental Biology* 213.17 (2010), pp. 2913–2923.

[24] Laura Dittmar et al. "The behavioral relevance of landmark texture for honeybee homing". In: *Frontiers in behavioral neuroscience* 5 (2011).

[25] Ron O Dror, David C O'carroll, and Simon B Laughlin. "Accuracy of velocity estimation by Reichardt correlators". In: *JOSA A* 18.2 (2001), pp. 241–252.

[26] Martin Egelhaaf. "On the neuronal basis of figure-ground discrimination by relative motion in the visual system of the fly". In: *Biological Cybernetics* 52.2 (1985), pp. 123–140.

[27] Martin Egelhaaf. "The neural computation of visual motion information." English. In: *Invertebrate vision.* Ed. by E. Warrant and D. E. Nielsson. Cambridge University Press, 2006, pp. 399–461.

[28] Martin Egelhaaf, Roland Kern, and Jens Peter Lindemann. "Motion as a source of environmental information: a fresh view on biological motion computation by insect brains". In: *Frontiers in neural circuits* 8 (2014).

[29] Martin Egelhaaf and Werner Reichardt. "Dynamic response properties of movement detectors: theoretical analysis and electrophysiological investigation in the visual system of the fly". In: *Biological Cybernetics* 56.2 (1987), pp. 69–87.

[30] Martin Egelhaaf et al. "Neural encoding of behaviourally relevant visual-motion information in the fly". In: *Trends in neurosciences* 25.2 (2002), pp. 96–102.

[31] Martin Egelhaaf et al. "Spatial vision in insects is facilitated by shaping the dynamics of visual input through behavioral action." In: *Frontiers in neural circuits* 6.December (Jan. 2012), p. 108. ISSN: 1662-5110. DOI: 10.3389/fncir.2012.00108. URL: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3526811\&tool=pmcentrez\&rendertype=abstract.

[32] M O Franz and J S Chahl. "Insect-Inspired Estimation of Self-Motion". In: *Proc 2nd Workshop on Biologically Motivated Computer Vision 2002 BMCV 2002 2525.* 2002, pp. 171–180. DOI: 10.1007/3-540-36181-2\_17.

[33] M O Franz and H G Krapp. "Wide-field, motion-sensitive neurons and matched filters for optic flow fields." In: *Biological cybernetics* 83.3 (Sept. 2000), pp. 185–97. ISSN: 0340-1200. URL: http://www.ncbi.nlm.nih.gov/pubmed/11007295.

[34] Matthias O Franz and Javaan S Chahl. "Linear Combinations of Optic Flow Vectors for Estimating Self-Motion - a Real-World Test of a Neural Model". In: *Advances in Neural Information Processing Systems 15* 15 (2003), pp. 1–8.

[35] Matthias O Franz, Javaan S Chahl, and Holger G Krapp. "Insect-inspired estimation of egomotion." In: *Neural Computation* 16.11 (2004), pp. 2245–2260. URL: http://eprints.pascal-network.org/archive/00000371/.

[36]  Matthias O Franz et al. "Can fly tangential neurons be used to estimate self-motion?" In: *Proceedings of the 9th International Conference on Artificial Neural Networks ICANN 1999*. Vol. 2. 1999, pp. 994–999. ISBN: 0852967217. DOI: 10.1049/cp:19991242.

[37]  B J Frost, D R Wylie, and Y C Wang. "The processing of object and self-motion in the tectofugal and accessory optic pathways of birds." In: *Vision Research* 30 (1990), pp. 1677–1688.

[38]  Bart RH Geurten et al. "A syntax of hoverfly flight prototypes". In: *The Journal of experimental biology* 213.14 (2010), pp. 2461–2475.

[39]  Arfan Ghani et al. "Neuro inspired reconfigurable architecture for hardware/software co-design". In: *SOC Conference, 2009. SOCC 2009. IEEE International*. IEEE. 2009, pp. 287–290.

[40]  James J Gibson. "Visually controlled locomotion and visual orientation in animals". In: *British journal of psychology* 49.3 (1958), pp. 182–194.

[41]  Juergen Haag, Abhishek Mishra, and Alexander Borst. "A common directional tuning mechanism of Drosophila motion-sensing neurons in the ON and in the OFF pathway". In: *eLife* 6 (2017), e29044.

[42]  Robert A Harris, David C O'Carroll, and Simon B Laughlin. "Contrast gain reduction in fly motion adaptation". In: *Neuron* 28.2 (2000), pp. 595–606.

[43]  Richard I Hartley. "In defense of the eight-point algorithm". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 19.6 (1997), pp. 580–593.

[44]  B Hassenstein and W Reichardt. "Systemtheoretische Analyse der Zeit-, Reihenfolgen- und Vorzeichenauswertung bei der Bewegungsperzeption des Rüsselkäfers Chlorophanus". In: *ZNaturforsch* 11 (1956), pp. 513–524. ISSN: 10008047.

[45]  JH van Hateren and C Schilstra. "Blowfly flight and optic flow. II. Head movements during flight". In: *Journal of Experimental Biology* 202.11 (1999), pp. 1491–1500.

[46]  K Hausen and M Egelhaaf. "Neural mechanisms of visual course control in insects". In: *Facets of vision* (1989), pp. 391–424. URL: http://link.springer.com/chapter/10.1007/978-3-642-74082-4\_18.

[47]  Klaus Hausen. "Motion sensitive interneurons in the optomotor system of the fly". In: *Biological Cybernetics* 45.2 (1982), pp. 143–156.

[48]  Klaus Hausen. "The lobula-complex of the fly: structure, function and significance in visual behaviour". In: *Photoreception and vision in invertebrates*. Springer, 1984, pp. 523–559.

[49]  Patrick Hennig and Martin Egelhaaf. "Neuronal encoding of object and distance information: a model simulation study on naturalistic optic flow processing". In: *Frontiers in neural circuits* 6 (2012).

[50]  M Juusola et al. "Contrast gain, signal-to-noise ratio, and linearity in light-adapted blowfly photoreceptors." In: *The Journal of general physiology* 104.3 (1994), pp. 593–621.

[51]  Mikko Juusola and Gonzalo G de Polavieja. "The rate of information transfer of naturalistic stimulation by graded potentials". In: *The Journal of general physiology* 122.2 (2003), pp. 191–206.

[52]  Mikko Juusola et al. "Information processing by graded-potential transmission through tonically active synapses". In: *Trends in neurosciences* 19.7 (1996), pp. 292–297.

[53]  Katja Karmeier, Holger G Krapp, and Martin Egelhaaf. "Robustness of the tuning of fly visual interneurons to rotatory optic flow". In: *Journal of neurophysiology* 90.3 (2003), pp. 1626–1634.

[54]  Roland Kern et al. "Blowfly flight characteristics are shaped by environmental features and controlled by optic flow information". In: *The Journal of experimental biology* 215.14 (2012), pp. 2501–2514.

[55]  Roland Kern et al. "Function of a fly motion-sensitive neuron matches eye movements during free flight". In: *PLoS biology* 3.6 (2005), e171.

[56]  Bernd Kimmerle, A-K Warzecha, and Martin Egelhaaf. "Object detection in the fly during simulated translatory flight". In: *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology* 181.3 (1997), pp. 247–255.

[57]  J J Koenderink and A J Van Doorn. "Facts on optic flow". In: *Biological Cybernetics* 56 (1987), pp. 247–254. ISSN: 03401200.

[58]  H G Krapp, R Hengstenberg, and M Egelhaaf. "Binocular contributions to optic flow processing in the fly visual system." In: *Journal of neurophysiology* 85.2 (Mar. 2001), pp. 724–34. ISSN: 0022-3077. URL: http://www.ncbi.nlm.nih.gov/pubmed/11160507.

[59]  Holger G Krapp. "Neuronal matched filters for optic flow processing in flying insects". In: *International review of neurobiology* 44 (1999), pp. 93–120.

[60]  Holger G Krapp and Roland Hengstenberg. "A fast stimulus procedure to determine local receptive field properties of motion-sensitive visual interneurons". In: *Vision research* 37.2 (1997), pp. 225–234.

[61]  Rafael Kurtz. *Adaptive encoding of motion information in the fly visual system.* Springer, 2012.

[62]  Rafael Kurtz et al. "Adaptation accentuates responses of fly motion-sensitive visual neurons to sudden stimulus changes". In: *Proceedings of the Royal Society B: Biological Sciences* 276.1673 (2009), pp. 3711–3719.

[63]    Simon B Laughlin. "Matching coding, circuits, cells, and molecules to signals: general principles of retinal design in the fly's eye". In: *Progress in retinal and eye research* 13.1 (1994), pp. 165–196.

[64]    Aljoscha Leonhardt et al. "Asymmetry of Drosophila ON and OFF motion detectors enhances real-world velocity estimation". In: *Nature neuroscience* 19.5 (2016), p. 706.

[65]    Jinglin Li, Jens Lindemann, and Martin Egelhaaf. "Local motion adaptation enhances the representation of spatial structure at EMD arrays". In: *PLoS computational biology* 13.12 (2017), e1005919.

[66]    Jinglin Li, Jens P Lindemann, and Martin Egelhaaf. "Peripheral processing facilitates optic flow-based depth perception". In: *Frontiers in computational neuroscience* 10 (2016).

[67]    Pei Liang, Roland Kern, and Martin Egelhaaf. "Motion adaptation enhances object-induced neural activity in three-dimensional virtual environment". In: *The Journal of Neuroscience* 28.44 (2008), pp. 11328–11332.

[68]    Pei Liang et al. "Impact of visual motion adaptation on neural responses to objects and its dependence on the temporal characteristics of optic flow". In: *Journal of neurophysiology* 105.4 (2011), pp. 1825–1834.

[69]    Jens Peter Lindemann et al. "On the computations analyzing natural optic flow: quantitative model analysis of the blowfly motion vision pathway". In: *Journal of Neuroscience* 25.27 (2005), pp. 6435–6448.

[70]    Q-T Luong and Olivier D Faugeras. "Self-calibration of a moving camera from point correspondences and fundamental matrices". In: *International Journal of computer vision* 22.3 (1997), pp. 261–289.

[71]    T Maddess and SB Laughlin. "Adaptation of the motion-sensitive neuron H1 is generated locally and governed by contrast frequency". In: *Proceedings of the Royal society of London. Series B. Biological sciences* 225.1239 (1985), pp. 251–275.

[72]    Elmar Mair, Michael Suppa, and Darius Burschka. "Error Propagation in Monocular Navigation for Zinf Compared to Eightpoint Algorithm". In: *IEEE International Conference on Intelligent Robots and Systems*. Nov. 2013.

[73]    Hanno Gerd Meyer, Jens Peter Lindemann, and Martin Egelhaaf. "Pattern-Dependent response modulations in motion-sensitive visual interneurons, a model study". In: *PLoS One* 6.7 (2011), e21488.

[74]    Claus Müller. *Spherical harmonics, volume 17 of Lecture Notes in Mathematics.* 1966.

[75] David Nistér. "An efficient solution to the five-point relative pose problem". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.6 (2004), pp. 756–770.

[76] David C O'Carroll, Paul D Barnett, and Karin Nordström. "Local and global responses of insect motion detectors to the spatial structure of natural scenes". In: *Journal of vision* 11.14 (2011), pp. 20–20.

[77] Marc Potters and William Bialek. "Statistical mechanics and visual signal processing". In: *Journal de Physique I* 4.11 (1994), pp. 1755–1775.

[78] Werner Reichardt. "Autocorrelation, a principle for the evaluation of sensory information by the central nervous system". In: *Sensory communication* (1961), pp. 303–317.

[79] C Reisenman, J Haag, and A Borst. "Adaptation of response transients in fly motion vision. I: Experiments". In: *Vision research* 43.11 (2003), pp. 1293–1309.

[80] Richard Roberts, Christian Potthast, and Frank Dellaert. "Learning general optical flow subspaces for egomotion estimation and detection of motion anomalies". In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.* IEEE. 2009, pp. 57–64.

[81] C Schilstra and JH van Hateren. "Blowfly flight and optic flow. I. Thorax kinematics and flight dynamics". In: *Journal of Experimental Biology* 202.11 (1999), pp. 1481–1490.

[82] Alexander Schwegmann, Jens P Lindemann, and Martin Egelhaaf. "Depth information in natural environments derived from optic flow by insect motion detection system: a model analysis". In: *Frontiers in computational neuroscience* 8 (2014).

[83] Tobias Senst, Volker Eiselein, and Thomas Sikora. "Robust local optical flow for feature tracking". In: *IEEE Transactions on Circuits and Systems for Video Technology* 22.9 (2012), pp. 1377–1387.

[84] Marion Silies, Daryl M Gohl, and Thomas R Clandinin. "Motion-detecting circuits in flies: coming into view". In: *Annual review of neuroscience* 37 (2014), pp. 307–327.

[85] Sandra Single, Juergen Haag, and Alexander Borst. "Dendritic computation of direction selectivity and gain control in visual interneurons". In: *Journal of Neuroscience* 17.16 (1997), pp. 6023–6030.

[86] Mandyan V Srinivasan et al. "Honeybee Navigation: Nature and Calibration of the" Odometer"". In: *Science* 287.5454 (2000), pp. 851–853.

[87] MV Srinivasan et al. "Range perception through apparent image speed in freely flying honeybees". In: *Visual neuroscience* 6.05 (1991), pp. 519–535.

[88] NJ Strausfeld et al. "Parallel processing in the optic lobes of flies and the occurrence of motion computing circuits". In: *Invertebrate vision* 2006 (2006), pp. 349–398.

[89] Simon Strübbe, Wolfgang Stürzl, and Martin Egelhaaf. "Insect-Inspired Self-Motion Estimation with Dense Flow Fields—An Adaptive Matched Filter Approach". In: *PloS one* 10.8 (2015), e0128413. DOI: 10.1371/journal.pone.0128413.

[90] W Stürzl, Darius Burschka, and Michael Suppa. "Monocular ego-motion estimation with a compact omnidirectional camera". In: *Intelligent Robots and ...* (2010). URL: http://ieeexplore.ieee.org/xpls/abs\_all.jsp?arnumber=5649970.

[91] Wolfgang Stürzl and Jochen Zeil. "Depth, contrast and view-based homing in outdoor scenes". In: *Biological cybernetics* 96.5 (2007), pp. 519–531.

[92] Wolfgang Stürzl et al. "Mimicking honeybee eyes with a 280 degree field of view catadioptric imaging system". In: *Bioinspiration & biomimetics* 5.3 (2010), p. 036002.

[93] Sukeshni N Tamgade and Vibha R Bora. "Motion vector estimation of video image by pyramidal implementation of Lucas Kanade optical flow". In: *Emerging Trends in Engineering and Technology (ICETET), 2009 2nd International Conference on*. IEEE. 2009, pp. 914–917.

[94] Lance F Tammero and Michael H Dickinson. "Collision-avoidance and landing responses are mediated by separate pathways in the fruit fly, Drosophila melanogaster". In: *Journal of Experimental Biology* 205.18 (2002), pp. 2785–2798.

[95] Lance F Tammero, Mark A Frye, and Michael H Dickinson. "Spatial organization of visuomotor reflexes in Drosophila". In: *Journal of Experimental Biology* 207.1 (2004), pp. 113–122.

[96] GK Taylor and HG Krapp. "Sensory systems and flight stability: what do insects measure and why?" In: *Advances in insect physiology* 34.07 (2007). DOI: 10.1016/S0065-2806(07)34005-8. URL: http://www.sciencedirect.com/science/article/pii/S0065280607340058.

[97] DJ Tolhurst, Y Tadmor, and Tang Chao. "Amplitude spectra of natural images". In: *Ophthalmic and Physiological Optics* 12.2 (1992), pp. 229–232.

[98] G Turin. "An introduction to matched filters". In: *Information Theory, IRE Transactions on* 6.3 (1960), pp. 311–329.

[99] Thomas W Ullrich, Roland Kern, and Martin Egelhaaf. "Influence of environmental information in natural scenes and the effects of motion adaptation on a fly motion-sensitive neuron during simulated flight". In: *Biology open* 4.1 (2015), pp. 13–21.

[100]  JH Van Hateren. "Processing of natural time series of intensities by the visual system of the blowfly". In: *Vision research* 37.23 (1997), pp. 3407–3416.

[101]  JH Van Hateren. "Theoretical predictions of spatiotemporal receptive fields of fly LMCs, and experimental validation". In: *Journal of Comparative Physiology A* 171.2 (1992), pp. 157–170.

[102]  Anne-Kathrin Warzecha, Jutta Kretzberg, and Martin Egelhaaf. "Temporal precision of the encoding of motion information by visual interneurons". In: *Current Biology* 8.7 (1998), pp. 359–368.

[103]  Dirk Werner. *Funktionalanalysis, 5., erw.* Deutsch. Springer, 2005.

[104]  Harald Wolf. "Odometry and insect navigation". In: *Journal of Experimental Biology* 214.10 (2011), pp. 1629–1641.

[105]  D R Wylie. "Binocular neurons in the nucleus lentiformis mesencephali in pigeons: responses to translational and rotational optic flowfields." In: *Neuroscience Letters* 291 (2000), pp. 9–12. ISSN: 03043940.

[106]  Jochen Zeil. "Visual homing: an insect perspective". In: *Current opinion in neurobiology* 22.2 (2012), pp. 285–293.