On the relationship between neuronal codes and mental models

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Gerrit Ecke
aus Wickede (Ruhr)

Tübingen
2021

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	24.06.2021
Stellvertretender Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Hanspeter Mallot
2. Berichterstatter:	Prof. Dr. Felix Wichmann

Dedication and acknowledgment

I have decided to write the dedication and the acknowledgment of my thesis in german, my mother-tongue. I hope that I have found the words that express my gratitude as vivid as I perceive it.

Meiner Familie möchte ich diese Doktorarbeit widmen. Meiner Frau Georgi und meinen Kindern Amelie und Elisa habe ich große Opfer abverlangt. Sie haben mehr meiner Zeit und meine Energie verdient, als ich ihnen in den letzten Jahren geben konnte. Die fehlende Zeit kann ich nicht nachholen und nicht wiedergutmachen. Ich verspreche Euch in Zukunft die Aufmerksamkeit zu geben, die Euch gebührt.

Herrn Professor Mallot, meinem Doktorvater und Mentor, danke ich für seine Menschlichkeit, für das vertrauensvolle Verhältnis, und für seine große, fundierte, interdisziplinäre Expertise, die er bereitwillig und geduldig mit seinem Umfeld teilt. Danke für die Geduld mit mir und meinem Eigensinn; danke für die Freiheit, die ich genießen durfte.

Die offene, positive und konstruktive Atmosphäre, die wir am Lehrstuhl genießen durften, ist nicht selbstverständlich und sie ist unendlich wertvoll. Ich danke allen, mit denen ich zusammengearbeitet habe, für die tiefen Inhaltlichen Diskussionen. Ohne Anspruch auf Vollständigkeit liegt es mir am Herzen einige Namen von Mitarbeitern, mit denen ich besonders eng zusammengearbeitet habe, hier zu verewigen: Gregor Hardiess, Hansjürgen Dahmen, Heinz Bendele, Tristan Baumann, Harald Papp, Fabian Mikulasch, Thede Witschel. Danken möchte ich auch Michaela Mohr und Martina Schmöe-Selich für ihr offenes Ohr und ihre Herzlichkeit, womit beide dazu beigetragen haben, dass ich mich am Lehrstuhl so wohl gefühlt habe.

Herrn Professor Wichmann danke ich für die Begutachtung meiner Thesis und die engagierte Übernahme der Berichterstattung nach dem Abschluss meiner Arbeit. Es war mir eine Freude, tiefsinnige Fragen und wertvolles Feedback mit Ihnen zu diskutieren. Ein herzliches Dankeschön auch meinen beiden anderen mündlichen Prüfern, Herrn Professor Benda und Herrn Professor Sinz.

Contents

Summary

The superordinate aim of my work towards this thesis was a better understanding of the relationship between mental models and the underlying principles that lead to the self-organization of neuronal circuitry. The thesis consists of four individual publications, which approach this goal from differing perspectives.

While the formation of sparse coding representations in neuronal substrate has been investigated extensively, many research questions on how sparse coding may be exploited for higher cognitive processing are still open. The first two studies, included as chapter 2 and chapter 3, asked to what extend representations obtained with sparse coding match mental models. We identified the following selectivities in sparse coding representations: with stereo images as input, the representation was selective for the disparity of image structures, which can be used to infer the distance of structures to the observer. Furthermore, it was selective to the predominant orientation in textures, which can be used to infer the orientation of surfaces. With optic flow from egomotion as input, the representation was selective to the direction of egomotion in 6 degrees of freedom. Due to the direct relation between selectivity and physical properties, these representations, obtained with sparse coding, can serve as early sensory models of the environment.

The cognitive processes behind spatial knowledge rest on mental models that represent the environment. We presented a topological model for wayfinding in the third study, included as chapter 4. It describes a dual population code, where the first population code encodes places by means of place fields, and the second population code encodes motion instructions based on links between place fields. We did not focus on an implementation in biological substrate or on an exact fit to physiological findings. The model is a biologically plausible, parsimonious method for wayfinding, which may be close to an intermediate step of emergent skills in an evolutionary navigational hierarchy.

Our automated testing for visual performance in mice, included in chapter 5,

is an example of behavioral testing in the perception-action cycle. The goal of this study was to quantify the optokinetic reflex. Due to the rich behavioral repertoire of mice, quantification required many elaborate steps of computational analyses. Animals and humans are embodied living systems, and therefore composed of strongly enmeshed modules or entities, which are also enmeshed with the environment. In order to study living systems as a whole, it is necessary to test hypothesis, for example on the nature of mental models, in the perception-action cycle.

In summary, the studies included in this thesis extend our view on the character of early sensory representations as mental models, as well as on high-level mental models for spatial navigation. Additionally it contains an example for the evaluation of hypotheses in the perception-action cycle.

Zusammenfassung

Das übergeordnete Ziel meiner Arbeit an dieser Dissertation war ein besseres Verständnis des Zusammenhangs von mentalen Modellen und den zugrundeliegenden Prinzipien, die zur Selbstorganisation neuronaler Verschaltung führen. Die Dissertation besteht aus vier individuellen Publikationen, die dieses Ziel aus unterschiedlichen Perspektiven angehen.

Während die Selbstorganisation von Sparse-Coding-Repräsentationen in neuronalem Substrat bereits ausgiebig untersucht worden ist, sind viele Forschungsfragen dazu, wie Sparse-Coding für höhere, kognitive Prozesse genutzt werden könnte noch offen. Die ersten zwei Studien, die in Kapitel 2 und Kapitel 3 enthalten sind, behandeln die Frage, inwieweit Repräsentationen, die mit Sparse-Coding entstehen, mentalen Modellen entsprechen. Wir haben folgende Selektivitäten in Sparse-Coding-Repräsentationen identifiziert: mit Stereo-Bildern als Eingangsdaten war die Repräsentation selektiv für die Disparitäten von Bildstrukturen, welche für das Abschätzen der Entfernung der Strukturen zum Beobachter genutzt werden können. Außerdem war die Repräsentation selektiv für die die vorherrschende Orientierung in Texturen, was für das Abschätzen der Neigung von Oberflächen genutzt werden kann. Mit optischem Fluss von Eigenbewegung als Eingangsdaten war die Repräsentation selektiv für die Richtung der Eigenbewegung in den sechs Freiheitsgraden. Wegen des direkten Zusammenhangs der Selektivitäten mit physikalischen Eigenschaften können Repräsentationen, die mit Sparse-Coding entstehen, als frühe sensorische Modelle der Umgebung dienen.

Die kognitiven Prozesse hinter räumlichem Wissen ruhen auf mentalen Modellen, welche die Umgebung representieren. Wir haben in der dritten Studie, welche in Kapitel 4 enthalten ist, ein topologisches Modell zur Navigation präsentiert, Es beschreibt einen dualen Populations-Code, bei dem der erste Populations-Code Orte anhand von Orts-Feldern (Place-Fields) kodiert und der zweite Populations-Code Bewegungs-Instruktionen, basierend auf der Verknüpfung von Orts-Feldern,

kodiert. Der Fokus lag nicht auf der Implementation in biologischem Substrat oder auf einer exakten Modellierung physiologischer Ergebnisse. Das Modell ist eine biologisch plausible, einfache Methode zur Navigation, welche sich an einen Zwischenschritt emergenter Navigations-Fähigkeiten in einer evolutiven Navigations-Hierarchie annähert.

Unser automatisierter Test der Sehleistungen von Mäusen, welcher in Kapitel 5 beschrieben wird, ist ein Beispiel von Verhaltens-Tests im Wahrnehmungs-Handlungs-Zyklus (Perception-Action-Cycle). Das Ziel dieser Studie war die Quantifizierung des optokinetischen Reflexes. Wegen des reichhaltigen Verhaltensrepertoires von Mäusen sind für die Quantifizierung viele umfangreiche Analyseschritte erforderlich. Tiere und Menschen sind verkörperte (embodied) lebende Systeme und daher aus stark miteinander verwobenen Modulen oder Entitäten zusammengesetzt, welche außerdem auch mit der Umgebung verwoben sind. Um lebende Systeme als Ganzes zu studieren ist es notwendig Hypothesen, zum Beispiel zur Natur mentaler Modelle, im Wahrnehmungs-Handlungs-Zyklus zu testen.

Zusammengefasst erweitern die Studien dieser Dissertation unser Verständnis des Charakters früher sensorischer Repräsentationen als mentale Modelle, sowie unser Verständnis höherer, mentalen Modellen für die räumliche Navigation. Darüber hinaus enthält es ein Beispiel für das Evaluieren von Hypothesn im Wahrnehmungs-Handlungs-Zyklus.

Chapter 1

Introduction

The brain has frequently been called the most complex structure in the universe. Its function has been assessed with countless approaches and from so many perspectives that it seems hard to find a common ground. Fields of study include, but are not limited to, neuroanatomy, cellular biology, genetics, neuronal circuitry, control theory, perception, cognitive neuroscience, ethology, psychology, and philosophy. All fields interact with each other and describe aspects of the brain as a whole. If it is the goal to gain an understanding of the complex structure of the brain that is as comprehensive as possible, it is challenging, yet crucial, to identify the interplay between all these aspects.

This thesis results from my work at the cognitive neuroscience lab at the University of Tübingen. Cognitive neuroscience studies the biological processes that underlie cognition (Gazzaniga, 2009). With my thesis, I want to shed light on the relationship between mental models and the simple neuronal coding principles from which they emerge. Emergence, a term from systems theory, describes properties that only arise due to the interaction between the individual parts of a system. The brain is such a system, because cognitive operations can only rest on local rules between individual neurons. In the following, I shortly introduce some associated concepts: living systems, emergence and mental models.

1.1 Living systems

The definition of the term *system* is quite simple. Miller (1973) defines a system as "[...] a set of interacting units with relationships among them". However, many phenomena fall into this definition, so that systems constitute a very wide field

of study. Ludwig von Bertalanffy established the "general systems theory", which integrates general properties common to all kinds of systems (Von Bertalanffy, 1956). These include concepts like entity boundaries, homeostasis and adaptation. The study of living systems is an important and large subfield of these studies. Miller (1973) described living systems as composed of a hierarchy of nested subsystems. He classified the entities as: the cell, the organ, the organism, the group, the organization, the society and the supranational system.

The interaction between the entities of living systems is manifold. Cybernetics examines systems subject to control and communication, in the animal and in the machine (N. Wiener, 1948). One of the most important concepts in cybernetics is the feedback loop. It can be followed back to Jakob von Uexküll, who described the "Funktionskreis" (functional circuit) between "Wirkwelt" (world of effects) and "Merkwelt" (world of realization). Uexküll coined these elements the basis elements of controlled interaction between an animal and its environment (von Uexküll, 1928). In today's theories, the "Funktionskreis" lives on as the "perception-action cycle" (Fuster, 2004). The feedback principle constitutes multiple, recurrent interactions between the environment and the living system, or more accurately, between the environment and all nested manifestations of the subsystems. The whole of these interactions constitutes a complex, adaptive system with emergent properties (Ahmed et al., 2005).

Interactions at any level can influence any other level, which can be denoted as upward causation and downward causation. Evolutionary forces have evolved living systems, so that they interact with a complex world and survive in it. Therefore, we can expect that the brain as a system is thoroughly enmeshed with the body, as well as with the surrounding world, at any level of the living system hierarchy; a concept which has been termed "embodiment" (Thompson & Varela, 2001) If we want to acquire an understanding of neuronal functioning, it is therefore crucial to establish a holistic perspective on the various aspects of the system.

1.2 Emergence

Systems are subject to emergent properties, which describes the phenomenon that "regularities in system behavior [...] are not revealed by direct inspection of the laws satisfied by the components". Typically, systems with emergent properties

"are composed of copies of a relatively small number of components that obey simple laws" (Holland, 2000).

Examples of systems with emergent behavior are manifold. The ant colony may serve as an example for emergent system behavior. Hofstadter (1979) used ants in his book *Gödel, Escher, Bach*, to illustrate emergence of cognition. Single ants follow quite simple, reflex driven rules. For example, they follow pheromone trails and visual or chemical gradients, a behavior which can be reproduced with the very simplistic circuitry of "Braitenberg vehicles" (Braitenberg, 1986). Another example is nest homing, which relies on path integration. Desert ants (*Cataglyphis fortis*) follow a straight route back to their nest after a period of exploration of their environment. The homing vector can simply be calculated by counting steps, weighted by the direction of locomotion with respect to the compass direction (M. Muller & Wehner, 1988). With independent and similar agents that only differ by a small number of variants—e.g. workers, soldiers, drones, queens—the colony adjusts to external events in a very specific and organized manner.

The minimalist setting of cellular automata makes emergence of complex system properties from simple rules most apparent. A cellular automaton consist of uniform cells which carry states; these states have defined dynamic impact on the states of cells in a local neighborhood. The complexity that emerges from such simple settings motivated Stephen Wolfram to postulate a "new kind of science" (Wolfram, 2002). The most famous cellular automaton is Conway's Game of Life (Gardner, 1970), in which cells on a two dimensional checkerboard only accept one of two states at a time: being alive or being empty. Furthermore, the state of each cell, from one generation to the next, only depends on the state of the eight cells in direct neighborhood. If two or three cells in the neighborhood of a living cell are alive, the cell survives. If four or more cells are alive, the cell dies from overpopulation; if one or none cell is alive, the cell dies from isolation. Only if exactly three cells in the neighborhood of an empty cell are alive, the cell is a birth cell. These few rules are sufficient for the emergence of regularities of various degrees, like patterns that are stable in time, oscillating patterns, moving patterns, and self replicating patterns. The Game of Life is turing complete, and implementations of turing machines in the game exist (Rendell, 2016).

Similarly, neurons in the brain constitute a system, with a finite number of

neuron types that follow simple rules and interact locally[1]. Governed by signals from their surrounding, neuron predecessor cells migrate along glial scaffolds, organizing themselves into the ordered, layered architecture of the cortex. They differentiate into a set of types, like the Pyramidal or the Purkinje type. Neurites grow out to connect neurons, guided locally by extracellular cues of attraction and repellence, emitted by other cells. The basic layout is refined with learning, which again can only be based on local rules of interaction between cells. For example, neurons alter the strength of their interconnection following Hebb's associative rule: if a neuron repeatedly takes part in exciting another neuron, the connection, and therefore the efficiency in exciting the neuron, will be strengthened (Hebb, 1949).

The simple rules give rise to the emergence of the cognitive processes at various levels of granularity. Research on the link between cognition and the interaction between neurons is called connectionism. The earliest examples of connectionism approaches go back to Frank Rosenblatt, who invented the perceptron, a simplified model of neuronal circuitry, which was able to perform classification tasks (Rosenblatt, 1958). Connectionism is often discussed in contrast to symbol manipulation approaches, which were long thought to be the strongest candidates for the root of intelligent action (Newell & Simon, 1976). Especially in language processing, symbol manipulation capabilities in human cognition are evident. However, due to the principle of emergence, connectionism has strong advantages explaining the evolution of cognition. Both approaches are still a matter of debate and there are also attempts to reconcile both approaches (Marcus, 2001). Further example of research topics include memory, inference, discrimination and identification, knowledge representation, and even consciousness (Clark & Lutz, 2012). Recently, connectionism approaches have gained scientific attention, due to advances in artificial intelligence, following Alex Krizhevsky's perceptron-like neural network for image classification (Krizhevsky et al., 2012).

[1]For a good overview, see Kandel (2009), part VIII: development and the emergence of behavior.

1.3 Mental models

Characteristics of mental models

Kenneth Craik expressed the concept of mental models as a thought process that represents the surrounding world (Craik, 1943). He reasoned that a "small-scale model" of how the world works might be used to try out alternatives of action and to conclude which is the best of them. The mental models of the brain do not necessarily mirror the complexity of the external events they represent, but are often rather simplistic and heuristic in nature. They are constructed by means of perception and characterized as being context dependent, constantly evolving and adapting to the experiences of the individual (N. A. Jones et al., 2011).

It is compelling how humans translate their experiences of physical relations into simplistic models, which are often referred to as "naïve physics". One common scheme is to infer explanations for phenomena from analogy, especially in the case that the domain is unfamiliar for a person (Collins & Gentner, 1987). Analogies are often drawn in an anthropomorphic manner. For example, the increased motor speed of a vacuum cleaner with blocked nozzle may be explained with increased efforts to work against the resistance, while in fact the motor spins faster due to reduced resistance (DiSessa, 1983). DiSessa identified the common experience of an "impetus", or effort, that a person needs to apply against resistance of various degree in order to achieve a desired result, as the deeper, underlying explanatory cognitive pattern. An extension to the "naïve impetus theory" of motion accounts for errors people make when predicting trajectories. It assumes a qualitative difference between objects in motion and objects at rest, so that a permanent force is needed to maintain the motion of an object (McCloskey, 1983).

While the false assumption of an impetus leads to wrong assumptions about the trajectories of moving objects, it still accounts for many phenomena experienced in everyday life, like the common slow deceleration of moving objects (which is due to friction in classical mechanics), or the forces experienced when an object hits another object (which is due to the momentum of the moving object in classical mechanics). It is common that mental models fail to explain phenomena of the surrounding world accurately but suit their purpose in everyday life. Limitations become apparent in stereotyped thinking, when models contain reduced, arbitrary

categories that fall short in adequately modelling rich relationships.

Neuronal representations of mental models

Mental models are often described system-like, as associations and networks of mental objects. For example, the motion of an object can be described as a network of qualitatively different types of motion, linked by descriptions of object states before and after the motion (Forbus, 1983). Such mental modeling must somehow be accomplished by the set of local rules between the individual neurons of the brain.

Exploring the relation between mental models and individual neurons, H. Barlow (1987) observed that the structure of the cerebral cortex is similar throughout, even though it accomplishes a vast variety of very diverse tasks. He therefore concluded that the cortex performs uniform operation everywhere. Barlow proposed that this basic operation is the detection of "suspicious coincidences" of features from the environment. If features frequently occur at the same time, more often than on chance level, it is likely that they are causally related. Finding these relations in hyperdimensional sensory space is a challenging combinatorial task, which might be mitigated by the genetically determined connectivity between cortical areas. Signals that are prone for causal relationships are often layed out in proximity, which is most apparent in the topological maps of early sensory processing. In visual processing for example, close-by receptors receive correlated signals, as chances are high that they stem from the same object in space (Srinivasan et al., 1982).

In Barlow's view, individual nerve cells or small populations of nerve cells serve as incidence detectors and therefore represent a hypothesis about the associative structure in its inputs. These hypotheses are then constantly tested by means of inductive inference; that is, hypotheses are confronted by facts in order to find out whether they are violated. Since cortical areas feed other cortical areas in a sequential manner, each area discovers valid hypotheses of this kind on varying levels of granularity. The repeated operation of finding suspicious coincidences leads to hierarchically organized, integrated representations of the sensory world.

Thagard (2010) built upon this notion and extended it with the proposal that mental models explain the cause for effects by means of abduction. Abductive reasoning starts from observations and searches for causes from which the ob-

servations would follow. intuitively, it follows the scheme "If p then q; why q? Maybe p" (Pierce, 1958). This notion supports the observation that people might be satisfied with a given explanation as long as it fits the result reasonably well. The concepts p that humans create could be described by the term "embodied abduction". They reflect sensations or patterns of visual-motor neuronal activity that cluster with similar experiences, which is reminiscent to the analogy thinking described in the previous subsection. Thagard proposed that new models are developed by creative conceptual combination, therefore creating more sophisticated, more abstract and more general mental models over time. In neuronal notion, this could be accomplished by the convolution of a number of neuronal ensembles that partially fit the sensation for which the brain seeks an explanation.

The approaches to neuronal representations of mental models are strongly related to the predictive coding principle. Inspired by telecommunication signal transmission techniques, Srinivasan et al. (1982) applied this efficient coding technique to image data. The processing only transmits statistically unexpected deviations from the sensory input. Optimizing for this objective, Srinivasan found a local center-surround organization of weights that fits physiological findings for retinal processing. Applying this principle to the hierarchical organization of the brain, each area contains a model of the expected errors it receives from previous areas and transmits unexpected errors to the next area. Higher level areas then send back suppressive signals to lower areas, trying to explain away prediction errors (Friston, 2008, 2010).

Recently, computational models with the premise of predictive coding have proven very successful. A predictive autoencoder can build representations that can be used as a pre-processing step for inference tasks; for example for speech recognition (Baevski et al., 2020; Schneider et al., 2019). Such studies reveal interesting aspects of underlying mental models. For example, a comprehensive computational model of the visual stream reproduced many physiological findings, including high-level invariant object representations and top-down effects that complete original full patterns (O'Reilly et al., 2017). Another study combined a predictive model network with a reinforcement control network to solve spatial computer games. The model could then be used to "dream" or "hallucinate" the games in order to further train the control network (Ha & Schmidhuber, 2018). A very interesting example of spatial cognition is the generative query net-

work, which is capable of scene representation and rendering, with the position and orientation of an agent as input parameters (Eslami et al., 2018). It was trained by comparing input images of a scene with predictions calculated from images taken from other perspectives. The errors from the predictions served as the learning signals for scene rendering from arbitrary positions. Interestingly, the representation could be used to render semantic aspects of a scene that were never presented to the network in that combination, like the particular combination of object shape and color as well as lighting conditions. It was also possible to retrieve an accurate top-down view map of the maze from the scene. The generative query network is an impressive demonstration of a neural network which establishes a general model of the environment solely learned from sensory example inputs by means of predictive coding.

1.4 Aim of the thesis

Throughout my thesis, I have approached the understanding of neuronal functioning at the level of neural coding principles. However, approaches from very different perspectives have contributed to a better understanding of the brain. One is the notion of the cortex as a model builder: Craik (1943) first hypothesized, that humans translate external events into mental models. With a representation of the surrounding world, the brain can make predictions about behavioral outcomes, therefore guiding decision making in a complex world.

How are models of the surrounding world represented at the level of neural coding? It is astonishing how we can assign meaning to individual cell activity, like edge detectors in the visual cortex (Hubel & Wiesel, 1959), face detectors in medial temporal lobe (Quiroga et al., 2005), or place cells in the hippocampus (O'Keefe, 1976). It appears that neural codes are, mostly, constituted by a population code, with *cardinal cells* of intermediate selectivity, which hold aspects of the stimulus they represent (H. B. Barlow, 1972). However, "meaning" is nothing that is clearly defined, but some measure of categorization that might reflect our models of the world that surrounds us.

Which neural coding principles account for the formation of mental models? The premise for biologically plausible models of learning is plasticity, based on information available at the synapse, like Hebbian learning (Hebb, 1949) or spike-

timing-dependent plasticity (Taylor, 1973). Meaningful pattern selectivity of single cells can emerge from such simple learning rules. For example, H. Barlow (1987) hypothesized, that the brain builds working models of the environment by forming associations between *suspicious coincidences*, a motif which could manifest by Hebbian plasticity.

The superordinate aim of my studies was to gain a better understanding on the relationship between the codes that serve as models of the environment and the simple rules for the formation of the neural connectivity. I compiled this thesis from four individual studies that are included in the following chapters:

- In the first study, the representation obtained from sparse coding of stereo visual input constituted a model for spatial layout and shape.
- In the second study, the representation obtained from sparse coding of optic flow input constituted a model for egomotion.
- The characterization of a novel dual population coding scheme was the focus of the third study. The population code served as a model of the environment for spacial cognition.
- The fourth study was an automated system for testing visual performance in mice. It is an example for the quantification of behavior in the perception-action cycle.

References

Ahmed, E., Elgazzar, A. S., & Hegazi, A. S. (2005). An overview of complex adaptive systems. *arXiv preprint nlin/0506059*.

Baevski, A., Schneider, S., & Auli, M. (2020). Vq-wav2vec: Self-Supervised Learning of Discrete Speech Representations [arXiv: 1910.05453]. *arXiv:1910.05453 [cs]*.

Barlow, H. (1987). Cerebral Cortex as Model Builder [Reporter: Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience]. In L. M. Vaina (Ed.), *Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience* (pp. 395–406). Springer Netherlands. https://doi.org/10.1007/978-94-009-3833-5_18

Barlow, H. B. (1972). Single Units and Sensation: A Neuron Doctrine for Perceptual Psychology? [Number: 4 Reporter: Perception]. *Perception*, *1*(4), 371–394. https://doi.org/10.1068/p010371

Braitenberg, V. (1986). *Vehicles: Experiments in synthetic psychology*. MIT press.

Clark, A., & Lutz, R. (2012). *Connectionism in context*. Springer Science & Business Media.

Collins, A., & Gentner, D. (1987). How people construct mental models. In D. Holland & N. Quinn (Eds.), *Cultural Models in Language and Thought* (1st ed., pp. 243–266). Cambridge University Press. https://doi.org/10.1017/CBO9780511607660.011

Craik, K. (1943). *The Nature of Explanation*. Cambridge University Press.

DiSessa, A. (1983). Phenomenology and the evolution of intuition [Publisher: Lawrence Erlbaum Press]. *Mental models*.

Eslami, S. M. A., Jimenez Rezende, D., Besse, F., Viola, F., Morcos, A. S., Garnelo, M., Ruderman, A., Rusu, A. A., Danihelka, I., Gregor, K., Reichert, D. P., Buesing, L., Weber, T., Vinyals, O., Rosenbaum, D., Rabinowitz, N., King, H., Hillier, C., Botvinick, M., ... Hassabis, D. (2018). Neural scene representation and rendering. *Science*, *360*(6394), 1204–1210. https://doi.org/10.1126/science.aar6170

Forbus, K. D. (1983). Qualitative reasoning about space and motion [Publisher: Hillsdale, NJ: Erlbaum]. *Mental models*, 53–73.

Friston, K. (2008). Hierarchical Models in the Brain (O. Sporns, Ed.). *PLoS Computational Biology*, *4*(11), e1000211. https://doi.org/10.1371/journal.pcbi.1000211

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, *11*(2), 127–138. https://doi.org/10.1038/nrn2787

Fuster, J. M. (2004). Upper processing stages of the perception–action cycle. *Trends in Cognitive Sciences*, *8*(4), 143–145. https://doi.org/10.1016/j.tics.2004.02.004

Gardner, M. (1970). Mathematical Games [Publisher: Springer Science and Business Media LLC]. *Scientific American*, *223*(4), 120–123. https://doi.org/10.1038/scientificamerican1070-120

Gazzaniga, M. S. (Ed.). (2009). *The cognitive neurosciences* (4th ed) [OCLC: ocn297494728]. MIT Press.

Ha, D., & Schmidhuber, J. (2018). World Models [arXiv: 1803.10122]. *arXiv:1803.10122 [cs, stat]*. https://doi.org/10.5281/zenodo.1207631

Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory*. J. Wiley; Chapman & Hall.

Hofstadter, D. R. (1979). *Gödel, escher, bach*. Harvester press Hassocks, Sussex.

Holland, J. H. (2000). *Emergence: From chaos to order*. OUP Oxford.

Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex [Publisher: Wiley Online Library]. *The Journal of physiology*, *148*(3), 574–591. https://doi.org/10.1113/jphysiol.1959.sp006308

Jones, N. A., Ross, H., Lynam, T., Perez, P., & Leitch, A. (2011). Mental Models: An Interdisciplinary Synthesis of Theory and Methods. *Ecology and Society*, *16*(1). https://doi.org/10.5751/ES-03802-160146

Kandel, E. R. (2009). The Biology of Memory: A Forty-Year Perspective. *Journal of Neuroscience*, *29*(41), 12748–12756. https://doi.org/10.1523/JNEUROSCI.3958-09.2009

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 1097–1105.

Marcus, G. (2001). *The algebraic mind*. Cambridge, MA: MIT Press.

McCloskey, M. (1983). Naive theories of motion. *Mental models*, 299–324.

Miller, J. G. (1973). *Living systems* (Vol. 1378). McGraw-Hill New York.

Muller, M., & Wehner, R. (1988). Path integration in desert ants, Cataglyphis fortis [Publisher: Proceedings of the National Academy of Sciences]. *Proceedings of the National Academy of Sciences*, *85*(14), 5287–5290. https://doi.org/10.1073/pnas.85.14.5287

Newell, A., & Simon, H. A. (1976). Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, *19*(3), 113–126. https://doi.org/10.1145/360018.360022

O'Keefe, J. (1976). Place units in the hippocampus of the freely moving rat [Publisher: Elsevier BV]. *Experimental Neurology*, *51*(1), 78–109. https://doi.org/10.1016/0014-4886(76)90055-8

O'Reilly, R. C., Wyatte, D. R., & Rohrlich, J. (2017). Deep Predictive Learning: A Comprehensive Model of Three Visual Streams [arXiv: 1709.04654]. *arXiv:1709.04654 [q-bio]*.

Pierce, C. S. (1958). *Collected papers Vol. 5, eds.* P Weiss, C Hartshorne & A. Burks, Harvard: Harvard University Press.

Quiroga, R. Q., Reddy, L., Kreiman, G., Koch, C., & Fried, I. (2005). Invariant visual representation by single neurons in the human brain [Publisher: Springer Science and Business Media LLC]. *Nature*, *435*(7045), 1102–1107. https://doi.org/10.1038/nature03687

Rendell, P. (2016). *Turing Machine Universality of the Game of Life* (Vol. 18). Springer International Publishing. https://doi.org/10.1007/978-3-319-19842-2

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. [Publisher: American Psychological Association (APA)]. *Psychological Review*, *65*(6), 386–408. https://doi.org/10.1037/h0042519

Schneider, S., Baevski, A., Collobert, R., & Auli, M. (2019). Wav2vec: Unsupervised Pre-training for Speech Recognition [arXiv: 1904.05862]. *arXiv:1904.05862 [cs]*.

Srinivasan, M. V., Laughlin, S. B., & Dubs, A. (1982). Predictive coding: A fresh view of inhibition in the retina [Publisher: The Royal Society London]. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, *216*(1205), 427–459. https://doi.org/10.1098/rspb.1982.0085

Taylor, M. (1973). The problem of stimulus structure in the behavioural theory of perception. *South African Journal of Psychology*, *3*, 23–45.

Thagard, P. (2010). How Brains Make Mental Models. In L. Magnani, W. Carnielli, & C. Pizzi (Eds.), *Model-Based Reasoning in Science and Technology: Abduction, Logic, and Computational Discovery* (pp. 447–461). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-15223-8_25

Thompson, E., & Varela, F. J. (2001). Radical embodiment: Neural dynamics and consciousness. *Trends in cognitive sciences*, *5*(10), 418–425. https://doi.org/10.1016/S1364-6613(00)01750-2

Von Bertalanffy, L. (1956). *General system theory* (Vol. 1).

von Uexküll, J. (1928). *Theoretische Biologie*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-36634-9

Wiener, N. (1948). *Cybernetics: Or the Control and Communication in the Animal and the Machine. Paris, France: Librairie Hermann & Cie, and Cambridge, MA*. Mrr Press. Cambridge, MA: MIT Prcss.

Wolfram, S. (2002). *A new kind of science* (Vol. 5). Wolfram media Champaign, IL.

List of publications

The following publications constitute the main body of this thesis, included as chapters 2–5.

1. Ecke, G. A., Papp, H. M., & Mallot, H. A. (2021). Exploitation of image statistics with sparse coding in the case of stereo vision. *Neural Networks*, 135, 158–176. Available from: http://dx.doi.org/10.1016/j.neunet.2020.12.016

2. Ecke, G. A., Bruijns, S. A., Hölscher, J., Mikulasch, F. A., Witschel, T., Arrenberg, A. B., & Mallot, H. A. (2019). Sparse coding predicts optic flow specificities of zebrafish pretectal neurons. *Neural Computing and Applications*, 32(11), 6745–6754. Reprinted by permission from Springer Nature: Springer Neural Computing and Applications, Copyright © 2020. Available from: http://dx.doi.org/10.1007/s00521-019-04500-6

3. Mallot, H. A., Ecke, G. A., & Baumann, T. (2020). Dual Population Coding for Path Planning in Graphs with Overlapping Place Representations. *Spatial Cognition XII*, 3–17. Reprinted/adapted by permission from Springer Nature: German Conference on Spatial Cognition, Copyright © 2020. Available from: http://dx.doi.org/10.1007/978-3-030-57983-8_1

4. Benkner, B., Mutter, M., Ecke, G., & Münch, T. A. (2013). Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex. *Behavioral Neuroscience*, 127(5), 788–796. Copyright © 2013, American Psychological Association. Reproduced with permission. Available from: http://dx.doi.org/10.1037/a0033944

Authorship

Exploitation of image statistics with sparse coding in the case of stereo vision[2]

Literature research and the concept of the study was my contribution, under the influence of discussions with lab colleagues and under the supervision of Professor Mallot. Most of the data analysis was carried out on my own, with the following exceptions. The generation of the stereo image data base was carried out within the scope of a Bachelor thesis by Kevin Reich under my supervision (Binokulare Bildstatistik mit virtueller Vergenz, 2017). The data analysis on tuning maps of surface orientation, was derived from a Master thesis by Harald M. Papp under my supervision (A likelihood approach for first order depth estimation from sparse stereo-representation, 2018). The article was written by my own, influenced by valuable feedback from colleagues from the lab and especially from Professor Mallot. The work of many Bachelor- and Master students was an important source for my understanding of the sparse coding algorithm, and in that sense incorporated into this publication.

[2]Ecke, G. A., Papp, H. M., & Mallot, H. A. (2021). Exploitation of image statistics with sparse coding in the case of stereo vision. *Neural Networks*, 135, 158–176. Available from: http://dx.doi.org/10.1016/j.neunet.2020.12.016

Sparse coding predicts optic flow specificities of zebrafish pretectal neurons[3]

The concept of the study was developed by Professor Mallot and by Professor Arrenberg, who also supported us with valuable input of background regarding zebrafish larvae. The virtual fish tank, the projection to a virtual retina, and the extraction of optic flow, as well as a fist comparison to the physiological reference data, was carried out within the scope of the Bachelor thesis from Fabian Mikulasch under my supervision (Self-organization of motion-sensitive receptive fields in the zebrafish optokinetic system, 2017). The analysis of optic flow selectivity with feed-forward networks, trained with backpropagation, was carried out within the scope of the Bachelor thesis from Sebastian Bruijns under my supervision (Self-motion estimation from optic flow with neural networks in simulated zebrafish, 2017). The analysis of ego-motion selectivity of the sparse representation was carried out within the scope of the Bachelor thesis from Johannes Hölscher under my supervision (A sparse coding model of zebrafish ego-motion detection from optic flow, 2018). Thede Witschel reanalyzed and improved the extraction of optic flow. I contributed with the supervision of the students and reanalysis and consolidation of the data, with focus on the LCA sparse coding. The article was written to equal parts by Professor Mallot and me. The work of many Bachelor- and Master students was an important source for my understanding of the sparse coding algorithm, and in that sense incorporated into this publication.

[3]Ecke, G. A., Bruijns, S. A., Hölscher, J., Mikulasch, F. A., Witschel, T., Arrenberg, A. B., & Mallot, H. A. (2019). Sparse coding predicts optic flow specificities of zebrafish pretectal neurons. *Neural Computing and Applications*, 32(11), 6745–6754. Available from: http://dx.doi.org/10.1007/s00521-019-04500-6

Dual population coding for topological navigation: Combining discrete state-action-graphs with distributed spatial knowledge[4]

The basic idea of the algorithm was jointly invented in the lab seminar. The concept of the study was developed to equal parts by me, Professor Mallot and Tristan Baumann, within the scope of Tristan Baumann's Bachelor- and Master theses. Software development and data analysis was carried out by Tristan Baumann under my supervision. The article was written to equal parts by Tristan Baumann and Professor Mallot. Note that the chapter included in this thesis is an extended version of the publication. It contains an additional landmark replacement experiment.

Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex[5]

The setup was developed within the scope of Boris Benkner's PhD thesis, under the supervision of Thomas Münch. My contribution was the collaborative development and implementation of algorithms and other software functions. I contributed to the software development of the presentation of stripe patterns, the mouse tracking method, the behavioral scoring method, and the graphical user interface.

[4]Mallot, H. A., Ecke, G. A., & Baumann, T. (2020). Dual Population Coding for Path Planning in Graphs with Overlapping Place Representations. *Spatial Cognition XII*, 3–17. Available from: http://dx.doi.org/10.1007/978-3-030-57983-8_1

[5]Benkner, B., Mutter, M., Ecke, G., & Münch, T. A. (2013). Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex. *Behavioral Neuroscience*, 127(5), 788–796. Available from: http://dx.doi.org/10.1037/a0033944

Contributions of this thesis

This chapter contains a summary of the contributions of the individual publications from which I compiled this thesis.

Exploitation of image statistics with sparse coding in the case of stereo vision[6]

We started from the assumption, that the brain uses sparse coding to exploit statistical properties of the sensory stream. With the hypothesis that a set of some patterns from the external world is embedded in a representation obtained with sparse coding, we assumed that these patterns can be retrieved with simple readout. We assessed our hypothesis with stereo visual data, using the Locally Competitive Algorithm (LCA, Rozell et al. (2008)), followed by a naive Bayes classifier for simple readout.

We examined individual LCA units for their selectivity to stereo disparity and surface orientation, as well as for statistical properties of their receptive field shapes. These findings were then compared to physiological findings of neurons from the visual cortex. We evaluated how the error of inference depends on the parameters overcompleteness and sparsity, and found that the size of the set of detectable patterns grows with expanded, redundant representations. Furthermore, we found a correlation between the inference error and the number of active LCA units, a relation that can be used to predict the inference error.

[6]Ecke, G. A., Papp, H. M., & Mallot, H. A. (2021). Exploitation of image statistics with sparse coding in the case of stereo vision. *Neural Networks*, 135, 158–176. Available from: http://dx.doi.org/10.1016/j.neunet.2020.12.016

Sparse coding predicts optic flow specificities of zebrafish pretectal neurons[7]

In this study (Ecke et al., 2020), we hypothesized that specificities of zebrafish pretectal neurons reflect the input statistics of natural optic flow. The assumption was tested with a virtual reality setup of a fish tank, from which we generated visual input akin to the wide-field visual system of zebrafish larvae. Optic flow was extracted using Flownet 2.0 (Ilg et al., 2017), followed by the formation of a sparse representation with the locally competitive algorithm (LCA, Rozell et al. (2008)).

Individual LCA units were assessed for their receptive field shapes and for their selectivity to egomotion (rotation and translation). We revealed substantial differences between learned units with and without preliminary whitening of the optic flow input. We carried out experiments according to the protocol of Kubo et al. (2014), who examined the egomotion selectivity of zebrafish pretectal neurons. Results from the LCA model, from the whitened LCA model, and from an additional simple feed forward network, were compared against the results from Kubo et al. (2014). Results from the LCA model were in good general agreement with the fish data, therefore indicating that sparse coding is a good candidate for formation of neural circuitry in zebrafish pretectal neurons.

Dual population coding for topological navigation: Combining discrete state-action-graphs with distributed spatial knowledge[8]

We developed a new topological navigation scheme that can be classified as a view-graph model of topological navigation (e.g. Franz et al. (1998c)). In these models,

[7]Ecke, G. A., Bruijns, S. A., Hölscher, J., Mikulasch, F. A., Witschel, T., Arrenberg, A. B., & Mallot, H. A. (2019). Sparse coding predicts optic flow specificities of zebrafish pretectal neurons. *Neural Computing and Applications*, 32(11), 6745–6754. Available from: http://dx.doi.org/10.1007/s00521-019-04500-6

[8]Mallot, H. A., Ecke, G. A., & Baumann, T. (2020). Dual Population Coding for Path Planning in Graphs with Overlapping Place Representations. *Spatial Cognition XII*, 3–17. Available from: http://dx.doi.org/10.1007/978-3-030-57983-8_1

agents reach their goal by the association of a recognized place with a movement command towards the next place. In our model, a large set of overlapping place fields (first population code) were constituted using SURF-features (Bay et al., 2008b). Place fields were associated with movement commands from one place field to the next (second population code). The commands were recorded as the directions of movement of the agent, when it first crossed the boundary between the two place fields. For navigation towards a goal position, the agent performed a graph search from all place fields at the current position to all place fields at the goal position. The movement direction was then calculated by voting over the directions from the first edges of all graph searches.

The model performed well with an agent in a virtual environment, it is therefore a good starting point for new, realistic models of spatial cognition. Our results show that recognition-response based navigation is possible with vague position information and with simple image processing. Furthermore, population coding with a distributed and quasi-continuous representation of space avoids the problem to select optimal snapshot positions. We also showed that our model can partially reproduce results from a human psychophysical navigation experiment reported by H. A. Mallot and Gillner (2000b) (Results not published, but included in chapter 4).

Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex[9]

With our setup (Benkner et al., 2013) it is possible to characterize acuity and contrast sensitivity of the mice visual system. The setup consists of four monitors, placed around a platform. Mice receive stimulation with a moving stripe pattern, which triggers the optokinetic reflex: a head movement that follows the movement of the stripe pattern. The process is completely automated, including the adaptation of the stripe pattern relative to the head of the mouse, and including an automated scoring procedure that rates the tracking behavior.

[9]Benkner, B., Mutter, M., Ecke, G., & Münch, T. A. (2013). Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex. *Behavioral Neuroscience*, 127(5), 788–796. Available from: http://dx.doi.org/10.1037/a0033944

Due to the strong focus on automatization, visual performance of mice can be assessed with high throughput. The development of the setup was the basis for the a startup company, which makes it commercially available as OptoDrum[10].

References

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008b). Speeded-up robust features (SURF). *Computer vision and image understanding*, *110*(3), 346–359. https://doi.org/10.1016/j.cviu.2007.09.014

Benkner, B., Mutter, M., Ecke, G., & Münch, T. A. (2013). Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex. *Behavioral Neuroscience*, *127*(5), 788–796. https://doi.org/10.1037/a0033944

Ecke, G. A., Bruijns, S. A., Hölscher, J., Mikulasch, F. A., Witschel, T., Arrenberg, A. B., & Mallot, H. A. (2020). Sparse coding predicts optic flow specificities of zebrafish pretectal neurons. *Neural Computing and Applications*, *32*(11), 6745–6754. https://doi.org/10.1007/s00521-019-04500-6

Franz, M. O., Schölkopf, B., Mallot, H. A., & Bülthoff, H. H. (1998c). Learning view graphs for robot navigation. *Autonomous agents*, 111–125. https://doi.org/10.1007/978-1-4615-5735-7_9

Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2017). FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2017.179

Kubo, F., Hablitzel, B., Dal Maschio, M., Driever, W., Baier, H., & Arrenberg, A. B. (2014). Functional Architecture of an Optic Flow-Responsive Area that Drives Horizontal Eye Movements in Zebrafish. *Neuron*, *81*(6), 1344–1359. https://doi.org/10.1016/j.neuron.2014.02.043

Mallot, H. A., & Gillner, S. (2000b). Route Navigating without Place Recognition: What is Recognised in Recognition-Triggered Responses? *Perception*, *29*(1), 43–55. https://doi.org/10.1068/p2865

[10] https://stria.tech/optodrum/

Rozell, C. J., Johnson, D. H., Baraniuk, R. G., & Olshausen, B. A. (2008). Sparse Coding via Thresholding and Local Competition in Neural Circuits [Number: 10 Reporter: Neural Computation]. *Neural Computation*, *20*(10), 2526–2563. https://doi.org/10.1162/neco.2008.03-07-486

Chapter 2

Exploitation of image statistics with sparse coding in the case of stereo vision[1]

Abstract

The sparse coding algorithm has served as a model for early processing in mammalian vision. It has been assumed that the brain uses sparse coding to exploit statistical properties of the sensory stream. We hypothesize that sparse coding discovers patterns from the data set, which can be used to estimate a set of stimulus parameters by simple readout. In this study, we chose a model of stereo vision to test our hypothesis. We used the Locally Competitive Algorithm (LCA), followed by a naïve Bayes classifier, to infer stereo disparity. From the results we report three observations. First, disparity inference was successful with this naturalistic processing pipeline. Second, an expanded, highly redundant representation is required to robustly identify the input patterns. Third, the inference error can be predicted from the number of active coefficients in the LCA representation. We conclude that sparse coding can generate a suitable general representation for subsequent inference tasks.

[1]Ecke, G. A., Papp, H. M., & Mallot, H. A. (2021). Exploitation of image statistics with sparse coding in the case of stereo vision. *Neural Networks*, 135, 158–176. Available from: http://dx.doi.org/10.1016/j.neunet.2020.12.016

2.1 Introduction

Among neural coding principles that have been proposed over time, sparse coding has a long-standing and successful history in explaining properties of neuronal circuitry. The firing rates of visual cortical neurons follow a sparse regime (R. Baddeley et al., 1997; Froudarakis et al., 2014; Rolls & Tovee, 1995) and several algorithms that model this premise predict receptive fields of visual cortex neurons quite accurately (see Hunter and Hibbard (2015), Hyvärinen et al. (1998), B. A. Olshausen and Field (1996), Rehn and Sommer (2007), and Ringach (2002)).

It is not straight forward to understand why sparse representations evolved in the brain. A possible explanation is based on the assumption that the neuronal code exploits statistical properties of the sensory input (H. Barlow, 2001; Simoncelli & Olshausen, 2001). Sparse coding represents the sensory input with a low number of specialized units that make the higher order, redundant components of a signal explicit (Bethge, 2006; Eichhorn et al., 2009; Field, 1987). This specialization is reminiscent of Barlow's concept of specialist units, or *cardinal cells*, with a selectivity intermediate between that of concrete *pontifical neurons* or *grandmother cells* and that of a typical distributed representation (H. B. Barlow, 1972, 2001). Cardinal cells could represent faces, objects, or, as Barlow puts it, *"a pattern of external events of the order of complexity of the events symbolized by a word"* (H. B. Barlow, 1972).

The sensory visual stream contains evidence for external events of various degrees of abstraction that are relevant for an animal to detect. Examples are the occurrence of a specific texture, an object that can be assigned to a category, or subtle cues, like signs of social interaction. We hypothesize that sparse coding supports the exploitation of sensory statistics by the formation of *cardinal cells* that make a subset of these external events accessible with a *simple readout* method.

Sparse coding transforms the sensory stream $\boldsymbol{x}$ into a representation $\boldsymbol{h} = H(\boldsymbol{x})$. $\boldsymbol{h}$ is the vector of activities of a set of *cardinal cells*, with an intermediate selectivity to external events $\{y_i\}$. We further assume that the selectivity of cardinal cells allows us to detect the occurrence of elements of $\{y_i\}$ with a simple processing step $\hat{y} = Y(\boldsymbol{h})$. For this *simple readout* we chose a naïve Bayes classifier

$$\hat{y} = \arg\max_i \; P(y_i) \prod_{k=1}^{K} P\left(h_k \,|\, y_i\right) \tag{2.1}$$

with uniform prior $P(y_i)$. It selects the external event y_i that most likely occurred in the sensory stream, based on evidence from the K elements h_k in $\boldsymbol{h}$.

The readout $Y(\boldsymbol{h})$ assumes independence of the elements of $\boldsymbol{h}$. Sparse coding belongs to the class of independent component analysis algorithms (ICA) that aim to extract basis vectors which are statistically independent (Hyvärinen & Oja, 2000). Note that, in the case of image data, the independence of basis vectors obtained by standard ICA algorithms is known to be strongly violated (Bethge, 2006; Eichhorn et al., 2009). Interestingly, classifiers that assume independence often yield surprisingly good results, even though existing dependencies between variables are omitted (Hand & Yu, 2001; Kuncheva, 2006; Kupervasser, 2014; Zhang, 2005).

It is unclear how to identify the set of external events that is accessible with this simple readout. However, assuming that the striate cortex forms a representation akin to sparse coding, we can use physiological evidence to identify candidates. For our evaluation we therefore adopt stereo vision, which is an early detection task. Indeed, we can compare our results with a large body of literature that is concerned with stereo vision in biological systems.

The contributions of this paper are: (*i.*) a characterization of stereo kernels learned with the Locally Competitive Algorithm (LCA), and their associated tuning to disparity and surface orientation in comparison to physiological findings, (*ii.*) an evaluation of disparity inference with simple readout from the LCA representation, subject to sparsity load and overcompleteness, (*iii.*) a method to predict the inference error, based on the number of active coefficients in the LCA representation.

2.2 Related work

2.2.1 Compact vs. expanded codes

Barlow reasoned in his efficient coding hypothesis that an efficient code, stripped by its redundancies, makes information more accessible, just as reducing the size of a haystack simplifies the task of finding needles (H. B. Barlow, 1959). He later extended this view by arguing that, in such a compact representation, interference between several, simultaneously occurring events might impair their separability

(H. B. Barlow, 2001). Gardner-Medwin and Barlow (2001) hypothesized that event retrieval from a population code is optimal when overlap between the neurons that correspond to each event is minimal. They tested their assumption by linking each of a number of hypothetical events to a fixed, random subset of binary neurons within a population. Overlap then was subject to two degrees of freedom: the number of neurons that, on average, corresponded to an event, and the total number of neurons in the population. Results indicated minor (but evident) impact of mean neural activity, but strong impact of population size on the readout error. Their findings suggest that an expanded, exceedingly redundant representation provides an optimal basis for event retrieval. An encoding with the sparse coding algorithm transforms the sensory stream into such an expanded, redundant representation (Field, 1994). Moreover, Gardner-Medwin and Barlow varied mean activity and population size, which are also parameters of the sparse coding algorithm. The population size corresponds to the dimensionality, which is usually several times overcomplete, and activity can be adjusted by the sparsity load of the optimization. We report how varying these parameters effects disparity inference in Sec. 2.5.2 and Sec. 2.5.2.

2.2.2 Sparse coding and pattern recognition

Rigamonti et al. (2011) examined the sparse coding algorithm as the first processing step in an image classification pipeline. They found that the features extracted by sparse coding were superior to handcrafted features even when they were used as a simple convolutional filter bank. They also evaluated the classification error as a function of sparsity penalty. No substantial improvement over convolutional processing was found. Better classification performance was monotonically linked to *lower* sparsity penalty. Bhatt and Ganguly (2018) found the opposite: better classification performance with larger sparsity penalty, however with the very specialized MNIST dataset. Also employing the MNIST dataset for evaluation, Lopez-Hazas et al. (2018) imposed sparsity on a perceptron-like feed forward network by adjusting neural thresholds, and similarly obtained a positive correlation between high sparsity penalty and classification performance.

2.2.3 Sparse coding and stereo vision

A considerable amount of work on stereo vision with unsupervised learning methods was carried out in the context of independent component analysis (ICA). Hoyer and Hyvärinen (2000) applied ICA to color- and stereo images and received disparity tuned Gabor-like basis vectors. Left and right basis vectors were similar, but varied in position and phase, as well as in the the degree of ocular dominance. Hunter and Hibbard (2015) performed a thorough analysis of ICA stereo basis vectors, obtained from a database carefully adjusted to the human visual system. The most notable difference to physiological data was two modes in the difference of phase between left and right basis vectors. The two modes were at zero and at π radians phase-shift, i.e., with opposite polarity. This finding might be related to a model from Li and Atick (1994), who derived kernels for correlated and anticorrelated left and right stereo half-images.

Lonini et al. (2013) found that a sparse representation can be learned altogether with vergence control. They reasoned that the angular orientation of both eyes has significant impact on achievable optimality of the representation. In their model, vergence control was a function of the global distribution of disparities. This is in line with psychophysical experiments with humans, which fits a population coding model that minimizes overall disparity energy in the two half-images (H. A. Mallot et al., 1996).

Lundquist et al. (2017), Lundquist et al. (2016) used stereo sparse coding, followed by a classifier, for depth inference, as well as for object detection. Their model outperformed others in the case of limited labeled training data. They concluded that the competition inherent in sparse coding requires elements to match associated contextual cues. Timofte and Van Gool (2015) tackled the associated problem of optic flow detection with a model which performed competitive to state of the art algorithms.

2.2.4 Stereo vision in biological systems

In the visual cortex of mammals, most cells in foveal striate and prestriate cortex show binocular interaction (Guillemot et al., 1993; Hubel & Wiesel, 1970; Hubel & Wiesel, 1962; Hubel et al., 2015; Levay et al., 1978; G. F. Poggio & Fischer, 1977; Tanabe et al., 2011). Binocular simple cells are similar to kernels obtained with

sparse coding or ICA. They best respond to Gabor-like binocular stimuli, with slight differences in position and phase (Anzai et al., 1999). V1 receptive fields are, however, more variant, with a tendency to appear more blob-like, with fewer sinusoidal sub-fields (Ringach, 2002). Binocular complex cells are more generally tuned to disparity than binocular simple cells, irrespective of position and polarity of the stimulus within the receptive field. In the standard model, complex cells are driven by a quadrature pair of Gabor-like monocular simple cells (Ohzawa et al., 1990).

Robust disparity inference requires further processing. Two constraints are crucial for the recovery of depth. First, each location in one stereo half-image corresponds to at most one location in the other half-image. Second, depth varies smoothly in general Marr and Poggio (1976, 1979). The constraints hold well, with the exception of strong local violation at the edges of objects. Optimization for both constraints yields the disparity of corresponding image locations. With epipolar geometry and with known distance of the eyes, disparity can be used to calculate depth (Hartley & Zisserman, 2004). Read and Cumming (2007) presented a model which relates the correspondence problem to differences in position- and phase of the receptive fields of binocular simple cells. Equally shaped Gabor filters that only vary in position are the best match for the corresponding structures in both half-images, whereas phase-shift Gabor filters carry the information to detect false matches. Goncalves and Welchman (2017) showed that a simple readout of disparity from simple cells incorporates this information.

We assume that a representation built by sparse coding provides a generalizing, yet limited basis for a range of pattern detection tasks. In order to test this assumption, we experimented with the detection of other characteristics of spatial layout than disparity. Psychophysical findings indicate that many more cues than point disparities contribute to a complete understanding of spatial layout. Examples include orientations of lines, light intensity differences, disparate specular highlights, and monocular occlusions. For an overview of geometrical and global aspects of stereopsis see H. A. Mallot (1999). Neurons in caudal intraparietal area were shown to be selective for first order depth, i.e., for specific surface tilt- and slant angles, and neurons in the temporal sulcus were shown to be selective for second order depth, i.e., for concave and convex curvature (Orban, 2011). Responses of such neurons were highly specific and robust against texturing and other orders

of depth. We therefore decided to test the sparse coding representation for first order depth selectivity. For an overview of physiological aspects of higher order visual processing of 3D-shape in the brain see Orban (2008).

2.3 Databases

Analyses of this paper rely on four databases. The virtual vergence database was used for LCA optimization, the disparity database and the naturalistic scene database for disparity inference, and the surface orientation database was used to characterize LCA selectivity to surface orientation.

2.3.1 Virtual vergence database

We captured images around Tübingen, Germany, with a ZED stereo camera[2]. The camera was equipped with two 1/3″ sensors, fixed at 120 mm distance, with parallel principal axes. The fields of view covered 76°(H) × 47°(V), with a resolution of 2208 × 1242 px. With $f_{x/y} = 1400$ px, the central angular resolution was ∼0.04 degrees. Note that the angular resolution of the final images we used in the subsequent processing steps was ∼0.08 degrees, as described in detail below. Image data were stored lossless as 24 bit png-files after automated brightness and gamma correction. In total, 1081 pairs of pictures were taken, from which 222 were captured inside rooms, 480 showed man made outdoors structures and the remaining 379 comprised natural scenes.

[2]https://www.stereolabs.com/

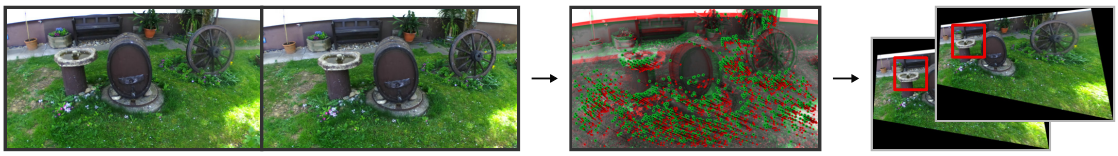


Figure 2.1. The virtual vergence database was created from images captured with a ZED stereo camera with parallel principal axes. **a)** Example image (view cross-eyed). **b)** Corresponding image points (SURF features) in the left and right half-images (red and green respectively, anaglyph image) were automatically matched and selected. **c)** Example stereo image with virtual vergence, created from **a** by correcting for radial distortion and applying homography transformation. Red frames indicate the extend of the final database images.

In order to obtain a database with vergence towards corresponding locations, images with several virtual fixations were created from each captured stereo image (see Fig. 2.1). SURF-features (Bay et al., 2008b) from left and right half-images were brute-force matched by the metric distance between the feature vectors. Only sufficiently similar matches below a threshold were selected and outliers with respect to the epipolar constraint were excluded.

For a given stereo image pair, a virtual fixation of any point in the image can be calculated by homographic transformation (Hartley & Zisserman, 2004). Because the transformation assumes a pinhole camera, the images were first corrected for radial distortion. The pixel positions $\boldsymbol{x}$ of these rectified images were then shifted to their new positions $\boldsymbol{x}'$. Assuming a rotation around the camera nodal point, the shift of each pixel was calculated with

$$\boldsymbol{x}' = KRK^{-1}\boldsymbol{x}\,, \tag{2.2}$$

where K is the camera matrix and R is the matrix that describes the rotation of the camera. We used the camera calibration app from the MATLAB computer vision toolbox to estimate the camera matrix K. The virtual rotation of each camera was determined, according to Listing's law, as a rotation around the axis $\boldsymbol{u}$ that is parallel to the image plane and perpendicular to the vector $\boldsymbol{s}-\boldsymbol{p}$ between the matched SURF-feature location $\boldsymbol{s}$ and the principle point of the camera $\boldsymbol{p}$ in the image. Therefore, the rotation axis was calculated as

$$\boldsymbol{u} = \begin{pmatrix} -(s_x - p_x) \\ s_y - p_y \\ 0 \end{pmatrix}\,. \tag{2.3}$$

The value of the rotation angle was calculated as

$$\Theta = \arctan \frac{\|\boldsymbol{s}-\boldsymbol{p}\|}{f_{x/y}}\,. \tag{2.4}$$

With the normalized vector $\hat{\boldsymbol{u}} = \boldsymbol{u}/\|\boldsymbol{u}\|$, the rotation matrix was then obtained

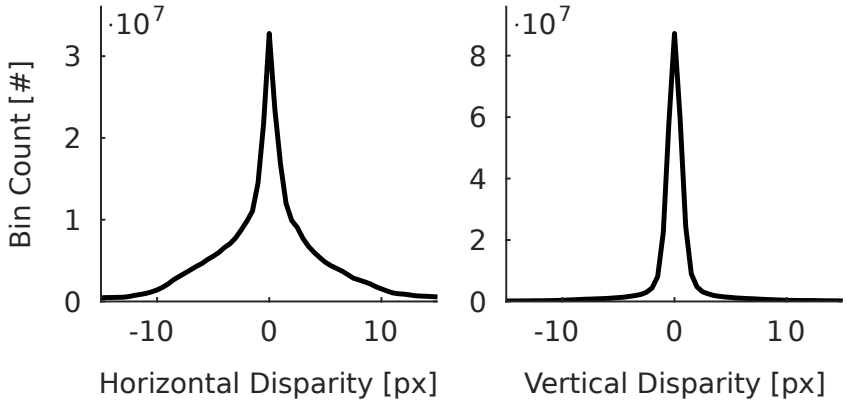


Figure 2.2. Distribution of horizontal and vertical disparities in the virtual vergence database. The histograms show the distribution of 3.3×10^8 randomly drawn data points (bin widths 0.8 px). Disparities were clustered around zero with high kurtosis ($k_x = 29.9$, $k_y = 118.1$).

by calculating

$$R = \begin{bmatrix} \cos\Theta + \hat{u}_x^2(1-\cos\Theta) & \hat{u}_x\hat{u}_y(1-\cos\Theta) & \hat{u}_y \sin\Theta \\ \hat{u}_x\hat{u}_y(1-\cos\Theta) & \cos\Theta + \hat{u}_y^2(1-\cos\Theta) & -\hat{u}_x \sin\Theta \\ -\hat{u}_y \sin\Theta & \hat{u}_x \sin\Theta & \cos\Theta \end{bmatrix}. \tag{2.5}$$

In order to keep local image statistics intact, we discarded images in which the virtual camera rotation angles exceeded 20 degrees. Pixel values were mapped to the new pixel raster and downscaled to half the original resolution with bicubic interpolation. The angular resolution of the final images was therefore ∼0.08 degrees. They were cropped to 256×256 px, centered at the respective principal points. In total, the virtual vergence database consisted of 72 991 images. We extracted the distribution of disparities contained in the database with FlowNet 2.0 (Ilg et al., 2017); results are shown in Fig. 2.2.

2.3.2 Disparity database

The disparity database contained stereo images, where each right stereo half-image was a shifted version of the left half-image. Images were collected from the same set that was used to create the virtual vergence database. It consisted of disparities in the range of d_x, $d_y = -6, \ldots, 6$ px, rasterized by 0.5 px in both dimensions. These were processed by cropping out 512×512 px sized pairs, randomly positioned in

the original images, with $2\,d_x$ and $2\,d_y$ px left-to-right offset. Next, they were downscaled to half the original resolution with bilinear filtering. We obtained $500 \times 25 \times 25$ image pairs, each with a resolution of 256×256 px. From these 500 images per stimulus, 490 were used for training and the remaining 10 were used for testing. We used convolutional LCA from Schultz et al. (2014) and accumulated data points over the feature maps (see Sec. 2.4.2 and Sec. 2.4.3). After discarding the margins, each feature map yielded 784 data points, which amounts to a total number of 384 160 data points per disparity and kernel for training and a total number of 7840 data points per disparity and kernel for testing.

2.3.3 Naturalistic scene database

We used the publicly available Genua Pesto database (Canessa et al., 2017), which contains two rendered 3d-scenes with vergence towards common fixation points. We used one of these scenes, the ground truth disparity and the right half-image of which are shown in Fig. 2.13a and b.

2.3.4 Surface orientation database

The surface orientation database contained stereo images of surfaces, textured with images from the same set that was used to create the virtual vergence database. With Blender[3], two virtual cameras, with a 11.8° field of view, were placed 7 cm apart and oriented towards the surface. The distance from the mid point between the two camera nodes to the central point of the surface was 1 m. The cameras were oriented so that the principal axes pierced the center of the surface, mimicking ocular vergence. We created stereo half-images for every combination of 36 tilt angles φ and 6 slant angles α with respect to a fronto-parallel plane. Tilt angles φ were equally spaced by 10°, and slant angles were set to $\alpha = 6°$, 24.3°, 38.2°, 48.2°, 55.2°. They were chosen so that each step increased disparity of a horizontally slanted surface by 1 px, assessed at 10 px horizontal distance from the center. We additionally included the images with a fronto-parallel plane $\alpha = 0°$. Per stimulus, we generated a training set with $L = 10050$ images, and an additional test set with 1000 images, both with 256×256 px resolution.

[3]https://blender.org

2.4 Modeling the visual processing pipeline

We built a simplified, naturalistic processing pipeline that mimics the mammalian visual system. For an illustration, see Fig. 2.3. Processing started from two horizontally separated eyes, with vergence towards a common fixation point. Visual sensory data underwent retinal pre-processing and were propagated to the model's sub-unit resembling V1, where a sparse representation was established. Finally, a naïve Bayes classifier was used for simple readout. If not acknowledged otherwise, implementation was carried out in MATLAB[4]. The retina model and the LCA sparse coding were implemented in PetaVision[5].

2.4.1 Retinal processing

Retinal processing was modeled in two steps. First, each image was smoothed by Gaussian filtering ($\sigma = 0.5\,\mathrm{px}$). Then, mimicking receptive fields with center-surround organization, images were convolved with a difference-of-Gaussians filter (DoG, inner Gaussian: $\sigma = 1\,\mathrm{px}$, outer Gaussian: $\sigma = 5.5)\,\mathrm{px}$. For each Gaussian kernel, weights were normalized so that the integral was equal to 1. Before propagation to the LCA sparse coding layer, each image was mean-centered and rescaled to a common ℓ_2-norm.

2.4.2 Establishing a sparse representation

In order to model V1, we used the locally competitive algorithm (LCA), introduced by Rozell et al. (2008), extended to convolutional LCA by Schultz et al. (2014) (see also Zeiler et al. (2010) and Lundquist et al. (2016)). Here, we provide a short summary of the algorithm. Sparse coding is the optimization of an error function that consists of two terms: a reconstruction term for reversibility, and a penalty which encourages sparsity (B. A. Olshausen & Field, 1996). In the case of stereo sparse coding, where the inputs were left and right stereo half-images $\boldsymbol{I}_L$ and $\boldsymbol{I}_R$, reconstruction was approximated by the convolutions

$$\boldsymbol{I}_L \approx \sum_{k=1}^{K} \boldsymbol{\Phi}_{L,k} * \boldsymbol{A}_k \quad \text{and} \quad \boldsymbol{I}_R \approx \sum_{k=1}^{K} \boldsymbol{\Phi}_{R,k} * \boldsymbol{A}_k \,. \tag{2.6}$$

[4]MATLAB Release 2018a, The MathWorks, Inc., Natick, Massachusetts, United States.
[5]https://petavision.github.io/

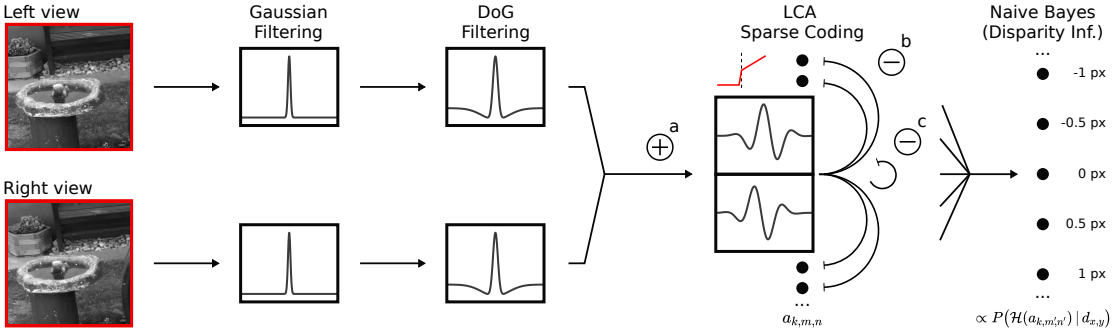


Figure 2.3. Schematic processing pipeline. Left and right half-images from the virtual vergence database were first pre-processed by a convolution with a Gaussian and subsequent difference-of-Gaussians filtering. In neural network notion, processing with the Locally Competitive Algorithm (LCA) is equivalent to a recurrent network. It is driven by excitatory feed forward connections, with learned weights that are usually Gabor-like (labeled a), competition through mutual lateral inhibition, with connection strengths proportional to the pairwise similarity of the feed-forward weights (b) and self inhibition or leaky integration (c). A naïve Bayes classifier was used for simple readout. It is equivalent to a simple feed-forward network, with weights proportional to the LCA neurons' log-probability of being active in the presence of a stimulus, and an additional winner-take-all mechanism.

$\Phi_L = \{\boldsymbol{\Phi}_{L,k}\}_{k=1}^{K}$ and $\Phi_R = \{\boldsymbol{\Phi}_{R,k}\}_{k=1}^{K}$ were sets of left and right half-kernels. Both half-kernels were convolved with a common corresponding feature map from the set $A = \{\boldsymbol{A}_k\}_{k=1}^{K}$, so that the reconstruction of the left and the right stereo half-image was coupled. We jointly normalized the left and right half-image by their ℓ_2-norm, which enabled learning of monocular dominant kernels. The particular error function for stereo sparse coding was

$$E = \frac{1}{2}\left(\left\|R(\boldsymbol{I}_L, \Phi_L, A)\right\|_2^2 + \left\|R(\boldsymbol{I}_R, \Phi_R, A)\right\|_2^2\right) + S(A)\,, \tag{2.7}$$

with the reconstruction term

$$R(\boldsymbol{I}_{L/R}, \Phi_{L/R}, A) = \boldsymbol{I}_{L/R} - \sum_{k=1}^{K} \boldsymbol{\Phi}_{L/R,k} * \boldsymbol{A}_k\,. \tag{2.8}$$

For standard sparse coding, the sparsity penalty $S(A)$ is the ℓ_1-norm of the coefficients of A (B. A. Olshausen & Field, 1996; Tibshirani, 1996). The LCA penalizes the number of super-threshold coefficients, given a threshold λ. With convolutional feature map dimensions $M \times N$, and with coefficients $a_{k,m,n}$ of $\boldsymbol{A}_k$, the sparsity

penalty was

$$S(A) = \sum_{k,m,n} \mathcal{H}(a_{k,m,n} - \lambda)\,, \tag{2.9}$$

where $\mathcal{H}(x) = 1$ if $x > 0$ and $\mathcal{H}(x) = 0$ otherwise. Note that this formulation requires the activity to be restricted to $a_{k,m,n} \geq 0$, which is convenient in neural network notion. Optimization of Eq. 2.7 for kernels $\Phi_{L/R}$, as well as activity in A, was obtained by the gradient descent procedure described by Rozell et al. (2008) and Schultz et al. (2014).

We set the kernel size to 16×16 px and the stride of the convolutions to 8 px, so that $k \times 2 \times 2$ elements k, m, n contributed to the reconstruction of single image pixels. We obtained five sets with $K = 85$, 128, 384, 1024 and 2048 kernels respectively, which constituted 0.66, 1, 3, 8 and 16 times overcomplete representations. λ was set to 0.1 for all models at learning time and was only varied at test time.

2.4.3 Simple readout

The readout was based on representations obtained by running the LCA procedure on stereo images, but with learning of $\boldsymbol{\Phi}_{L/R}$ turned off. The kernels were obtained from the previous LCA optimization on the virtual vergence database. An example of disparity readout is visualized in Fig. 2.4. With each image presentation, the set of feature maps A was used for inference. After settled optimization, the coefficients $a_{k,m,n}$ of all feature maps $\boldsymbol{A}_k$ were set to binary states by applying $\mathcal{H}(a_{k,m,n})$, with $\mathcal{H}(x) = 1$ if $x > 0$ and $\mathcal{H}(x) = 0$ otherwise. Disparity was inferred based on the 2×2 coefficients $a_{k,m',n'}$ of feature maps $\boldsymbol{A}_k$, which is the extend of all coefficients that include a single pixel in their receptive fields. Surface orientation tuning was examined based on a larger 7×7 region around the central fixation point.

Each category y_i in $\boldsymbol{y}$ is represented by a unique two-dimensional parameter combination: horizontal and vertical disparity d_x and d_y, and surface tilt- and slant angles φ and α. At each image location, they can be estimated by selecting $\hat{y} = y_i$ for some i that is most probable. Assuming independence of the coefficients,

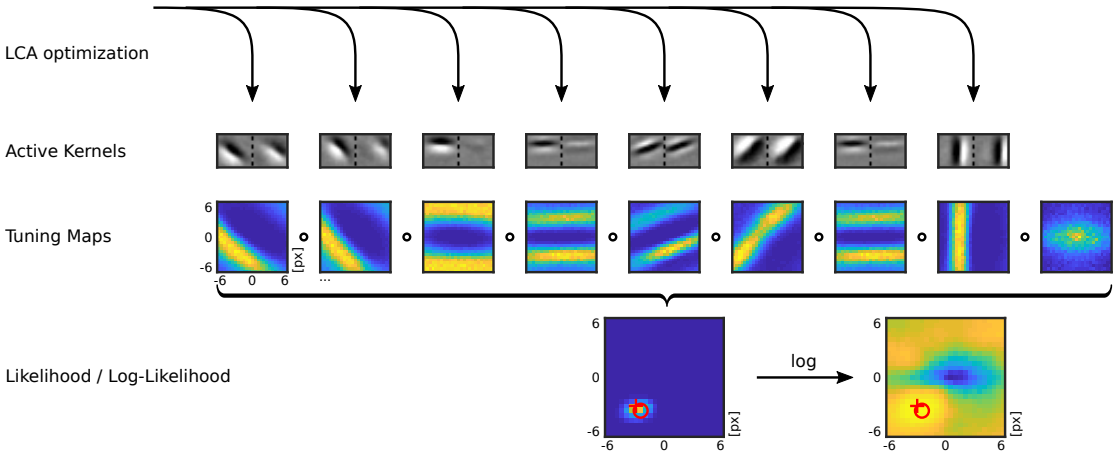


Figure 2.4. Example of disparity inference with the naïve Bayes classifier from a single image presentation with disparity $d_x = -2.5\,\mathrm{px}$, $d_y = -4\,\mathrm{px}$. LCA optimization results in a sparse set of active coefficients. Each binocular kernel $\mathbf{\Phi}_{L/R,k}$ (grayscale) is associated with a tuning map (arbitrary units). The tuning maps display the probability of the corresponding coefficient $a_{k,m',n'}$ to be in an active state, as a function of the evaluated range of x- and y-disparities. Bottom row: disparity likelihood / log-likelihood (prior omitted). The likelihood map is the Hadamard product of all 8 tuning maps associated with active coefficients, and all inverted tuning maps of non-active units (accumulated, outmost right). The true disparity is indicated by a red circle and the mode of the distribution is indicated by a red cross.

estimates were calculated by applying a naïve Bayes classifier with[6]

$$\hat{y} = \arg\max_i \; P(y_i) \prod_{k,m',n'} P\left(\mathcal{H}(a_{k,m',n'}) \,|\, y_i\right) \,. \tag{2.10}$$

We omitted the priors $P(y_i)$, even though a strongly non-uniform distribution of disparities is apparent in natural image data, as can be seen in Fig. 2.2.

Because elements $\mathcal{H}(a_{k,m',n'})$ were restricted to two states, the probabilities of being in one of these states, $P\left(\mathcal{H}(a_{k,m',n'}) = 1 \,|\, y_i\right)$ and $1 - P\left(\mathcal{H}(a_{k,m',n'}) = 1 \,|\, y_i\right)$ were determined experimentally by calculating the arithmetic mean. In the case of stereo disparity, we assumed that the probabilities were linked to each of the kernels $\mathbf{\Phi}_{L/R,k}$ and invariant with respect to image feature location. Therefore, probes were accumulated over the feature maps of size $M \times N$, as well as over the

[6] In practice, inference was calculated equivalently with the logarithmic form $\hat{y} = \arg\max_i \; \log P(y_i) + \sum_{k,m',n'} \; \log \; P\left(\mathcal{H}(a_{k,m',n'}) \,|\, y_i\right)$.

whole training set of size L by calculating

$$P\left(\mathcal{H}(a_{k,m',n'}) = 1 \,|\, y_i\right) \approx \frac{1}{LMN} \sum_l \sum_{m,n} (\mathcal{H}(a_{k,m,n}))_l \,. \tag{2.11}$$

For inference with Eq. 2.10, the same probability estimate of one kernel was used for all 2×2 locations m', n'.

In contrast, we assumed that probabilities differ with respect to image location in the case of surface orientation. We reasoned that kernels are mainly disparity tuned and that the orientation of a surface may be detected by the pattern of disparities within a local range. We therefore calculated probability estimates independently for all 7×7 locations m, n with

$$P\left(\mathcal{H}(a_{k,m,n}) = 1 \,|\, y_i\right) \approx \frac{1}{L} \sum_l (\mathcal{H}(a_{k,m,n}))_l \,. \tag{2.12}$$

Probability estimates vary smoothly with respect to the parameter combinations d_x and d_y as well as for φ and α and therefore constitute "tuning maps". In the case of surface orientation estimation we exploited this local continuity and smoothed out noise with two dimensional Savitzky-Golay filtering (Savitzky & Golay, 1964), with a polynomial of degree 3, and with 5 px width in both dimensions.

2.4.4 Linking the processing pipeline to biological vision

With this study, we hope to contribute to a better understanding of biological vision. We chose the aspects of our processing model so that we could study our hypothesis adequately. Here, we motivate some aspects of the simplified naturalistic processing stream, both with respect to biological findings, as well as to their functional role.

Vergence. As a first processing step, we incorporated vergence in our model, the rotation of the two eyes towards each other. The visual system controls gaze, so that the image of objects or any structure of interest is moved to the fovea, the location on the retina with the best spatial resolution. Vergence is not used in common technical solutions. State-of-the-art algorithms work with images obtained with parallel camera axes. They calculate depth by applying epipolar stereo geometry to corresponding locations in both half-images (Hartley & Zisserman, 2004). However, if the goal is to understand vision based on statistical processing,

vergence has crucial impact.

Sparse coding is an optimization that builds on statistical regularities of the underlying data. As a first approximation of stereo vision, each half-image is a locally shifted version of the other. The extend of these image shifts, or disparities, differs broadly over the whole scene. In contrast, if both eyes are oriented towards the same location, the distribution of disparities in the vicinity of the fixation point is very narrowly distributed around zero, as shown in Fig. 2.2. This finding is due to the local smoothness of disparities, disrupted only by discontinuities at object boundaries. Only through vergence the sparse coding algorithm can find statistical dependencies between the two half-images, because corresponding image locations are close-by. Statistical dependencies then manifest in similarly shaped left and right half-kernels, which are often slightly shifted versions of one another. Indeed, Hunt et al. (2013) have shown that sparse coding with simulated strabismus only extracts monocular dominant kernels.

Retinal pre-processing. Receptive fields of the retina are characterized by a center surround organization, with weights in central location that are opposed in polarity to the weights around the center. They are often modeled with the Mexican hat shaped Laplacian of a Gaussian, or the simpler approximation with the difference of two Gaussians, like in our case. Reasons discussed for this kind of retinal processing include mechanisms of efficient coding, compression, response equalization, sparseness and others (Graham et al., 2006). Convolving the visual input with a difference-of-Gaussians decorrelates the overall pink-noise spectrum of natural scenes, transforming it into a representation with equalized power spectrum (Atick & Redlich, 1992). Removing these first order correlations, also called whitening, is a common pre-processing step before applying an ICA procedure, because it affects the algorithms' search for higher order statistical dependencies (Hyvärinen & Oja, 2000).

Sparse coding in biological substrate. The sparse coding algorithm serves as a model for the formation of neuronal circuitry in V1. It has been proposed that the gradient descent on the error function, with respect to the coefficients, could be implemented directly in neural network topology (B. A. Olshausen, 2003; B. A. Olshausen & Field, 1997; Rozell et al., 2008). In the following, we consider single neurons for simplicity. The gradient descent on coefficients a, which is the activity of the neurons in neural network notion, follows a differential equation. In each

time step, the activity a of a neuron k changes proportionally to the sum of three terms (see also Fig. 2.3): (*i.*) a feed forward term $\boldsymbol{\varphi}_k^\intercal \boldsymbol{x}$, where the vectorized kernels $\{\boldsymbol{\varphi}_k\}_{k=1}^K$ serve as receptive fields for the input $\boldsymbol{x}$, (*ii.*) a competition term $-\sum_{c \neq k} \boldsymbol{\varphi}_k^\intercal \boldsymbol{\varphi}_c \, a_c$ that introduces lateral inhibition proportional to the activity a_c from all other neurons of the LCA layer, with weights proportional to the similarity of the receptive fields, and (*iii.*) a self-inhibition term $-a_k$. In LCA optimization, the sparse coding algorithm is extended by deriving a "leaky integrator" neuron (see Abbott (1999)). The main difference is the introduction of an inner state u, which is coupled to the output of the neuron with a thresholding function $a = T(u)$. The three terms stay the same with LCA sparse coding, except that they drive the inner state u of the neuron and that the self inhibition in term *iii.* is replaced by $-u$, the leak of the neuron.

This network is reminiscent to the Hopfield network (Hopfield, 1982; Hopfield, 1984; Little, 1974). With equivalent topology, the weights of networks derived from the Hopfield network are in many cases trained by applying biologically more plausible learning rules that rely on information available at the synapse. For example, Földiak presented an artificial neural network in which feed-forward weights were learned by Hebb's rule and lateral inhibition was subject to anti-Hebbian learning. Anti-Hebbian learning means that inhibitory connections between neurons were enhanced if they were active at the same time (Földiák, 1990). Therefore, the network learned competition between neurons that were driven by similar patterns, akin to term *ii.* It was shown that a network with these learning rules, applied to natural images, developes Gabor-like kernels (Falconbridge et al., 2006). Applying the same Hebbian, anti-Hebbian learning to spiking neural networks yields similar results, drawing even closer to a biologically accurate model (King et al., 2013; Zylberberg et al., 2011). Chauhan et al. (2018) applied such a network to stereo images and reported successful disparity readout of the neural population with a simple classifier. Physiological studies indeed provide evidence for lateral inhibition between neurons with similar receptive fields in V1: orientation selectivity of neurons might benefit from lateral inhibition between neurons with similar orientation tuning (Blakemore & Tobin, 1972) or from other types of cross-orientation inhibition (Ringach, 2003; Shapley et al., 2003).

Probabilistic inference. Hypotheses on properties of the world are subject to uncertainty. Bayesian inference provides a framework that allows to account for

ambiguity and a broad range of brain functions, like multimodal perception, decision making or motor control, have been modeled following Bayesian approaches (Doya et al., 2006; Knill & Pouget, 2004). Training a perceptron-like neural network with backpropagation is linked to probabilistic inference. With respect to stereo vision, Goncalves and Welchman (2017) analyzed the relationship between a binocular likelihood model and a two layer feed-forward neural network. The first layer represented simple cells, preset with Gabor-like receptive fields, and the second layer represented complex cells tuned for disparities. The weights of both layers were trained with back-propagation. The learned weights from simple to complex cells werer proportional to the log-probability of the simple cell being active, given the preferred disparity represented by the complex cell. Because neural networks of this kind compute the weighed sum of the individual units' activities, each complex cell calculated the log-likelihood of its preferred disparity. This is equivalent to the naïve Bayes classifier we used for inferring disparity, the logarithmic form of which can be implemented similarly in a neural network.

2.5 Results

Our analysis of the stereo-vision processing pipeline followed the main hypothesis of this paper: that patterns from the external world can be accessed with simple readout from a representation obtained with sparse coding. We chose disparity and surface orientation as candidates for such patterns. In Sec. 2.5.1, we first describe qualitative and quantitative properties of the learned LCA representation. Sec. 2.5.2 addresses the main hypothesis of the paper by evaluating the errors of simple readout of stereo disparity. In Sec. 2.5.2 and 2.5.2 we discuss the extend of errors subject to overcompleteness and sparsity of the LCA representation. The findings are expanded with Sec. 2.5.2, where we describe how the accuracy of the inference can be predicted by the overall activity in the LCA layer. The mechanism holds implications for possible attention mechanisms. Results from these subsections culminate in the evaluation of disparity maps of naturalistic scenes in Sec. 2.5.2. In Sec. 2.5.3 we then evaluate the orientation tuning of Kernels obtained by LCA optimization.

2.5.1 Characteristics of the LCA representation

In the following, we focus on results specific to disparity selectivity and compare them to physiological findings. As outlined in Sec. 2.2.3, the kernels obtained by applying ICA methods to stereo image data have been well described elsewhere. We therefore limit our report to results specific to LCA sparse coding.

Selectivity for disparity

For simple probabilistic readout, individual neurons need to exhibit some degree of specificity for the pattern of interest. Indeed, all tuning maps of kernels obtained with Eq. 2.11 yielded clear, smoothly varying selectivity as a function of disparity. This was true throughout all kernels obtained by optimizing Eq. 2.7, irrespective of the level of overcompleteness. Therefore, all kernels potentially contribute to disparity inference with the simple readout scheme. For representative examples, see Figs. 2.4, 2.5 and 2.7. We include all learned kernels in the supplementary material, Figs. S01–S05. The shape of the kernels was in most cases well described by the Gabor function (see Sec. 2.5.1 and Fig. 2.8a, c). Kernels which were not Gabor-shaped, and which were therefore not classical in terms of physiologically described receptive fields, did only emerge with higher levels of overcompleteness.

We identified three main types of kernel shapes: "Matched Gabor", "Tuned Inhibitory" and "Blob-like". A significant number of the "Matched Gabor" and the "Tuned Inhibitory" type were evident at all levels of overcompleteness. However, the share of the "Tuned Inhibitory" type was decreasing the lscarger the overcompleteness of the model. The "Blob-like" type only emerged in models which were

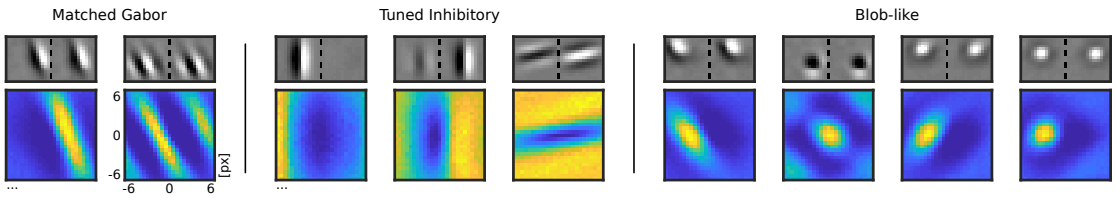


Figure 2.5. Typical kernels obtained by LCA optimization. *Matched Gabor* left and right half-kernels were very similarly shaped, but shifted in position. *Tuned Inhibitory* kernels were selective for large disparity values. They were mostly ocular dominant and left vs. right half-kernels were shifted in phase by about π radians. *Blob-like* kernels' weights consisted of a central spot and an outer lobe with opposed polarity. Disparity selectivity was more localized than for the other types.

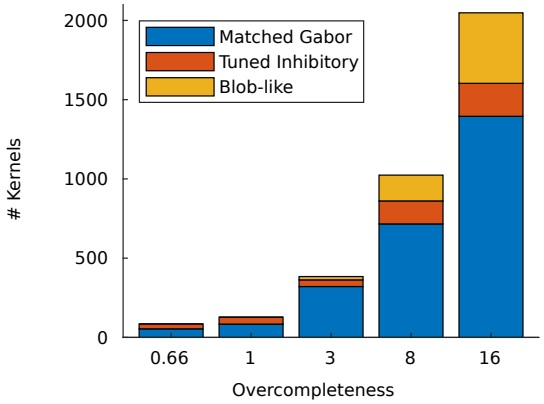


Figure 2.6. Proportion of the three kernel types on the total number of kernels, plotted for each of the five trained values of overcompleteness. "Matched Gabor" and "Tuned Inhibitory" types were evident on all levels of overcompleteness, with a decreasing fraction of "Tuned Inhibitory" types for larger models. "Blob-like" kernels only emerged in the models that were at least 3× overcomplete.

at least 3× overcomplete, with increasing share the larger the overcompleteness of the model. When presented with natural stereo images, the average number of kernels from each type that contributed to the reconstruction of the stimulus was proportional to their share in the set of kernels. This finding was slightly violated if the sparsity penalty λ was very high. In these cases, "Matched Gabor" kernels had an up to 10 % larger share on the number of active kernels than on average. The three types will be described in more detail in the following paragraphs. For examples of each type see Fig. 2.5, for the share of each type on the total number of kernels see Fig. 2.6.

Matched Gabor The majority of kernels were Gabor-like, with very similar left and right shapes. Differences between the two half-kernels were best described by a shift in position and almost no shift in phase. Tsao et al. (2003) reported that most receptive field shapes in V1 are also characterized by only a small amount of phase-shifts. Such kernels are well suited to represent corresponding (or matching) structures in the two half-images that originate from the same object in the world (see also Sec. 2.2.4). Conversely, the mode of the tuning maps was equal to the position-shift of the two half-kernels. Note that the mode was sharply peaked perpendicular to the orientation of the kernel shape, but wide in direction of the orientation. These kernels were therefore only selective for disparity perpendicular to their orientation.

Tuned Inhibitory The probability of these kernels being active increased with the absolute value of disparity. Typically, they were monocular or monocular dominant, i.e., most of the weight energy was in either the left or the right half-kernel.

If they were binocular, the lobes were usually shifted by about $\pi/2$ or by about π radians (see also Sec. 2.5.1). Such kernels were also reported by Hunter and Hibbard (2015), see Sec. 2.2.3. Note that phase-shift kernels might serve as "what not"-detectors when used for stereo inference, as described in Sec. 2.2.4 (Goncalves & Welchman, 2017; Read & Cumming, 2007). The weights of monocular kernels process information from only one stereo half-image. An explanation for the disparity selectivity based on feed forward processing is therefore unlikely. With sparse optimization on the other hand, matching structures can be reconstructed more sparsely with a single binocular kernel, where otherwise two monocular kernels would be needed. In neural network notion, monocular and binocular kernels compete against each other through lateral inhibition. The probability that a binocular kernel exists that can jointly represent both half-images decreases with larger disparities (see Fig. 2.10). Therefore, the likelihood that monocular kernels are active increases with disparity.

Blob-like The shapes of this type were not Gabor-like, but had in common a center-surround organization with a central spot of one polarity and a surrounding structure with opposed polarity. The surrounding lobe, however, varied in its extend, not always completely enclosing the central spot. The resulting shapes described a continuum, with a partial opening resembling an end-stopped ridge, an opening of approximately half extend matching a corner and an even further opening describing slightly curved edges. Kernels of this type were selective for disparity in both dimensions, as opposed to Gabor-like kernels, which were prone to the aperture problem: the displacement of an oriented structure can only be measured perpendicular to its orientation. Our results reflect image statistics and therefore show that natural images consist of a substantial amount of structures, which are best described as corners, ridges and blobs. Such elements may be used to reconstruct two-dimensionally displaced structures directly rather than with a combination of local spatial frequency elements, i.e. Gabor-like kernels. Note that Ringach (2002) reported a substantial amount of blob-like receptive field shapes in V1.

The kernel shapes we obtained with LCA sparse coding fit well to physiological findings. G. F. Poggio et al. (1988) recorded neuron responses from rhesus macaque monkey visual cortex and classified disparity tuned cells in six categories.

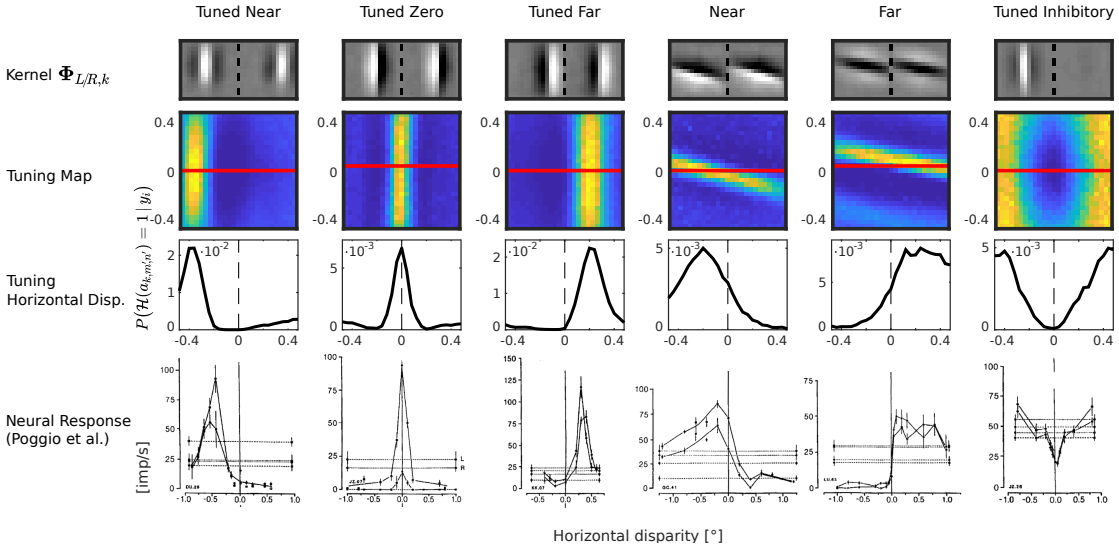


Figure 2.7. Comparison of example representatives from our data that match the six disparity response types defined by G. F. Poggio et al. (1988). Each column shows a single Kernel, its disparity tuning map and its horizontal cross section along the red line (horizontal disparity tuning). They fit the examples from the bottom row, which consists of the disparity tuning curves of physiological single neuron recordings from monkey visual cortex. All examples were drawn from the 16× overcomplete model, $\lambda = 0.04$. For details see Sec. 2.5.1.

Three of these, "Tuned Near", "Tuned Zero" and "Tuned Far" neurons, were characterized by sharply peaked response curves, tuned to negative, zero or positive horizontal disparities. The two categories "Near" and "Far" contained neurons that were similarly selective for negative or positive disparities. However, these neurons' responses were not as peaked as the responses of the "Tuned" neurons but rather broadly tuned. The last category was referred to as "Tuned inhibitory" and contained neurons that were more likely to fire the larger the disparity, irrespective of its sign. We can reproduce the physiological examples of all six categories with our kernel sets and present them in Fig. 2.7.

The three "Tuned" types describe the same response as our "Matched Gabor" kernels. They were sharply tuned to disparity, but only perpendicular to their orientation. If oriented vertically, they were therefore sharply tuned to horizontal disparity. In some cases they were tuned to more than one disparity, like in the second "Matched Gabor" example of Fig. 2.5. This was due to the repetitions of the sinusoids. However, most of our kernels had a single sinusoid lobe, like in the "Tuned Zero" and "Tuned Far" examples of Fig. 2.7, which resulted in a

single, elongated peak in the tuning maps. It seems that single lobed kernels are a specialty of LCA sparse coding, as compared to standard sparse coding. This finding will be discussed in more detail in Sec. 2.5.1 (see also Fig. 2.8h).

We also found tuning maps which reproduce the "Near"- and "Far" types. Oblique "Matched Gabor" kernels were more broadly tuned to horizontal disparity, which was due to the kernels' elongated response peaks in two-dimensional disparity space. However, we reproduced the horizontal tuning curves with shifted images, which was a very stable stimulus. We assume that horizontal disparity tuning of oblique kernels is very sensitive to small changes of vertical disparity. In addition, G. F. Poggio and Fischer (1977) found that most "Near"- and "Far" cells received unbalanced inputs from the two eyes, which is not true for our "Matched Gabor" kernels.

The "Tuned Inhibitory" type matches our own classification. In the respective paragraph we have offered an explanation for how the lateral inhibition of the sparse optimization leads to tuned inhibitory units. This finding has physiological support. G. F. Poggio and Fischer (1977) and G. F. Poggio and Talbot (1981) reported that "Tuned Inhibitory" neurons often showed "strong excitatory dominance of one eye (ocular unbalance), the 'silent' eye exercising only inhibitory functions and only over a restricted disparity range". They also reported bidirectional cells, "with balanced ocularity, from which stimulation of either eye alone evoked excitatory responses that of the two eyes together evident response suppression". These bidirectional cell's responses were similar to the response of kernels with about π radians shifted sinusoid. Further physiological evidence supports that suppressive mechanisms of this kind help to solve the stereo correspondence problem (Henriksen et al., 2016; Tanabe et al., 2011; Tanabe & Cumming, 2014). To the best of our knowledge, tuned inhibitory units in stereo vision have not been described in the context of sparse coding in the literature, yet.

Statistical analyses of the kernels

In order to characterize quantitative properties of the kernels from the LCA optimization, we fitted the Gabor-function

$$g(a, b, \phi, x, y, \theta, \sigma_x, \sigma_y) = a + b \exp(c) \cos(d) \tag{2.13}$$

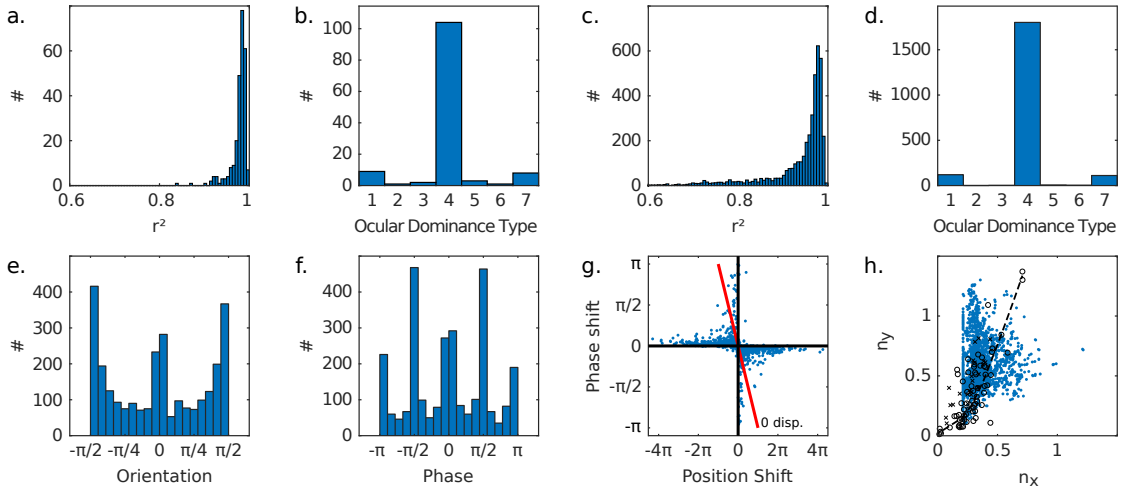


Figure 2.8. Statistics of the learned kernels $\mathbf{\Phi}_{L/R}$, for details see Sec. 2.5.1. Diagrams **a**, **b** contain data from the 1× overcomplete model (128 kernels), **c**–**h** contain data from the 16× overcomplete model (2048 kernels). **a**, **c)** Histogram of coefficients of determination as a measure for goodness of fit. **b**, **d)** Ocular dominance types on 7 point scale: 4 is binocular; 1 and 7 are left/right monocular, respectively. Row **e-h** is restricted to data with coefficient of determination $r^2 > 0.93$, in order to reject non-classical receptive field shapes. **e, f)** Distribution of orientation and phase. **g)** Interdependence between difference in position and phase of left vs. right Gabor fit. position-shift is expressed perpendicular to the orientation of the Gabor function and normalized by spacial frequency, calculated with $f \,\|(\Delta x, \Delta y)^\intercal\| \cos\phi$. The red line marks zero-disparity. **h)** Relationship $n_{x/y} = f\,\sigma_{x/y}$, with spatial frequency f and the width of the Gaussian envelope σ_x/σ_y (perpendicular/along orientation). Blue: our data. Black circles/crosses: data from macaque V1, reported by Ringach (2002)/J. P. Jones and Palmer (1987).

to each half-kernel $\mathbf{\Phi}_{L/R,k}$, with offset a, scale b, an elliptical Gaussian envelope $\exp(c) = \exp(\alpha x'^2 + 2\beta x'y' + \gamma y'^2)$ and a sinusoid $\cos(d) = \cos(2\pi f x' + \kappa)$ along x', with spatial frequency f and phase-shift κ. Orientation ϕ and position x, y in image space were free, with

$$\begin{aligned} x' &= \quad (x - x_0)\, cos(\phi) \;+\; (y - y_0)\, sin(\phi)\,, \\ y' &= -(x - x_0)\, sin(\phi) \;+\; (y - y_0)\, cos(\phi)\,. \end{aligned} \tag{2.14}$$

The elliptical envelope, with widths σ_x and σ_y, was allowed to rotate freely by the angle θ, relative to the orientation of the sinusoid, with

$$\begin{aligned} \alpha &= \frac{cos(\theta)^2}{2\sigma_x^2} + \frac{sin(\theta)^2}{2\sigma_y^2} \,, \\ \beta &= -\frac{sin(2\theta)}{4\sigma_x^2} + \frac{sin(2\theta)}{4\sigma_y^2} \,, \\ \gamma &= \frac{sin(\theta)^2}{2\sigma_x^2} + \frac{cos(\theta)^2}{2\sigma_y^2} \,. \end{aligned} \tag{2.15}$$

We used a custom implementation in MATLAB, which we made publicly available[7].

Most kernels were well described by the Gabor function, with the coefficient of determination r^2 close to 1 (Fig. 2.8a, c). Some of the lower values can be attributed to monocular kernels, in which the half-kernel with less weight energy has a lower signal-to-noise ratio. With higher levels of overcompleteness, more non-Gabor-like kernel shapes appeared, which is apparent with the heavy tail in the distribution of the 16× overcomplete model (2048 kernels) in Fig. 2.8c, as opposed to the distribution of the 1× overcomplete model (128 kernels) in Fig. 2.8a.

We analyzed the ocular dominance of the kernels by adapting the 7 point scale from Hubel and Wiesel (Hubel & Wiesel, 1962). They were calculated with

$$\arctan\left(\frac{\|\mathbf{\Phi}_{L,k}\|}{\|\mathbf{\Phi}_{R,k}\|}\right) \,, \tag{2.16}$$

with values in the range $[0, \pi/2]$ plotted in a histogram with 7 equally spaced bins. Kernels which were left or right monocular fell into category 1 and 7, respectively. If weight energy was equally distributed, kernels fell into category 4, the other categories were left or right dominant, respectively. We show the results in Fig. 2.8b and d. The majority of kernels were in category 4, i.e., binocular with balanced weight energy (see Fig. 2.8b, d). A substantial fraction of kernels was purely monocular (category 1 or 7). Only a small fraction was in the intermediate categories and the proportion of intermediate kernels was even lower with higher levels of overcompleteness. The shape of these kernels was usually characterized by a phase-shift of about π radians. Kernels that did not fall into category 4 were

[7]https://www.mathworks.com/matlabcentral/fileexchange/60700-fit2dgabor-data-options

usually of the "Tuned Inhibitory" type, described in Sec. 2.5.1. In physiological experiments, a similar three-mode distribution of weight energy between left and right receptive fields was also found in ferrets, albeit not as distinctly peaked as in our results. (Kalberlah et al., 2009). Other physiological studies on various animals report rather flat distributions (Guillemot et al., 1993; Hubel & Wiesel, 1962; Hubel et al., 2015; Levay et al., 1978; Schiller et al., 1976).

The following analyses were based on the 16× overcomplete model. Fits with a coefficient of determination of $r^2 < 0.93$ were excluded in order to exclude non-classical receptive field shapes. The distribution of orientations (Fig. 2.8e) had two peaks at 0 degrees and at ±90 degrees. Two possible explanations have been offered in the literature for this bias: the rasterization of the input images and the prevalence of orientations in human made structures (Hunt et al., 2013). The distribution of phases (Fig. 2.8f) had distinct peeks at 0 degrees, at ±90 degrees and at ±180 degrees, i.e., the kernel shapes were, in most cases, either sine-like or cosine-like. Ringach (2002) reported that physiological receptive fields similarly cluster in such even- and odd-symmetric shapes. Opposed to our findings with LCA sparse coding, he also reported that, with standard sparse coding, there is a tendency towards odd-symmetric receptive fields, but not towards even-symmetric receptive fields.

As described in Sec. 2.2.4, binocular Gabor-filters that are shifted in position from left to right half-kernel can serve as matched filters for corresponding image structures, whereas Gabor-filters shifted in phase can serve as "what not"-detectors for false matches (Goncalves & Welchman, 2017; Read & Cumming, 2007). We were therefore interested in the interrelationship between the shift in position and the shift in phase of the kernels in our data. Results are displayed in Fig.2.8g. Because the tuning maps were characterized by elongated peaks, we expressed the position-shift relative to the most sharply tuned axis. It was therefore calculated as the difference in horizontal and vertical position, projected on the axis perpendicular to the orientation of the Gabor function. For better comparability between position-shift and phase-shift, we also normalized the position-shift by the spatial frequency of the sinusoid. The position-shift was therefore calculated as $f \, \|(\Delta x, \Delta y)^\intercal\| \cos\phi$. Our data showed a transient separation between position-shift and phase-shift kernels. If a kernel had both, a substantial position- and phase-shift, they counteracted each other, so that almost all data points fell into

quadrant ii and iv. Lobes of the sinusoid match when data points are on the red line. The majority of the kernels was mainly shifted in position and therefore match the "Matched Gabor" type from Sec. 2.5.1.

Ringach (2002) reported that Gabor-like receptive field shapes of macaque V1 were more variable and often more blob-like than kernels from sparse coding and basis vectors from ICA. In his study, he related the spatial frequency f of the sinusoid to the extend of the Gaussian envelope σ_x, perpendicular to the orientation of the sinusoid, and σ_y, along the orientation of the sinusoid. The relationship $n_{x/y} = f\,\sigma_{x/y}$ was lower on average in physiologically measured receptive fields. In Fig. 2.8h, we show an overlay of the data adapted from Ringach (2002) (macaque V1, black circles), and from J. P. Jones and Palmer (1987) (cat V1, black crosses), with our data (blue dots). In this case, we fit the Gabor-functions with the orientation of the elliptical Gaussian envelope fixed at $\theta = 0$ degrees. While standard sparse coding and ICA results in values $n_{x/y} > 0.5$ for the majority of kernels / basis vectors, many kernels from convolutional LCA sparse coding were characterized by lower values. Note that we bound the fitting procedure to $n_{x/y} \geq 0.25$ and did therefore not allow blob-like fits, so that these kernels do not appear in the panel. The plot also shows that physiological receptive fields, as well as the LCA kernels, had a tendency for $n_y > n_x$, which was not true for standard sparse coding and ICA, as reported by Ringach. Kernels with small values for n_x, i.e., with a small extend of the Gaussian envelope perpendicular to their orientation, are better suited for disparity inference. If the kernel shape consisted of only one sinusoidal lobe ($n_x = 0.25$), the associated tuning map had a single elongated peak, as opposed to kernels with more than one sinusoidal lobe, which had multiple, parallel, elongated peaks. The disparity they represented was therefore not ambiguous. Indeed, we observed aliasing effects in the disparity inference if image structures were represented by multi-lobe kernels. For examples, see both "Matched Gabor"-kernels from Fig. 2.5.

2.5.2 Evaluation of disparity inference

In this subsection we evaluate whether disparities can successfully be obtained with simple readout from the LCA representation. We explored the limitations by means of the error of the estimates. Inference of disparity was carried out with the full processing pipeline, subject to overcompleteness and sparsity penalty in

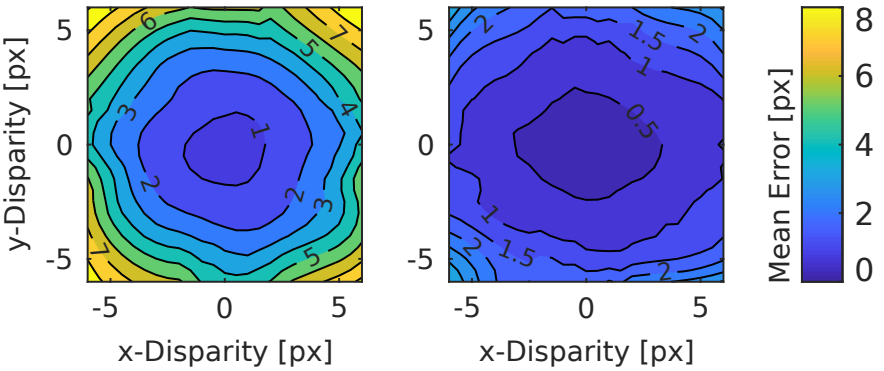


Figure 2.9. Mean error of absolute disparity estimates as a function of x- and y-disparity. *Left*: 1× overcomplete, $\lambda = 0.04$. *Right*: 16× overcomplete, $\lambda = 0.04$. Disparity estimates were evaluated with stereo images, in which the left half-image was a shifted version of the right half-image. With more overcompleteness in the LCA representation, the error for large disparities decreased.

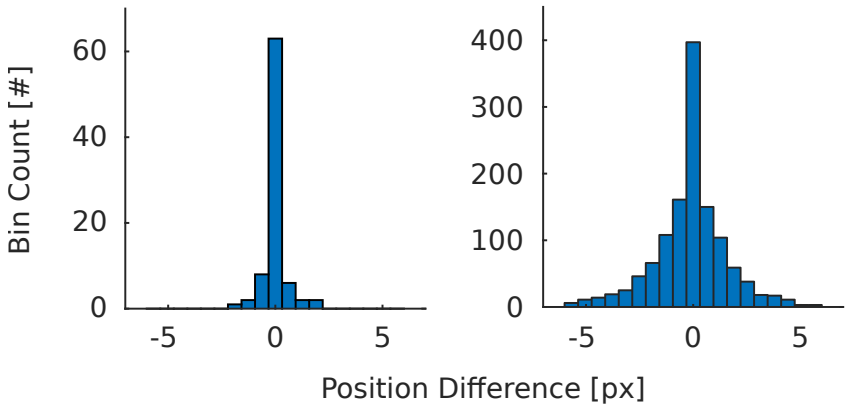


Figure 2.10. Distribution of the difference in position of "Matched Gabor" kernels. position-shift is expressed perpendicular to the orientation of the Gabor function. The plot includes all kernels with $r^2 > 0.93$ and $\phi < 0.3$ rad. *Left*: 1× overcomplete, 84 of 128 kernels. The kurtosis of the distribution is $k = 9.26$. *Right*: 16× overcomplete, 1264 of 2048. The kurtosis is $k = 5.12$. LCA optimization with more overcompleteness yields kernels that represent a wider range of disparities.

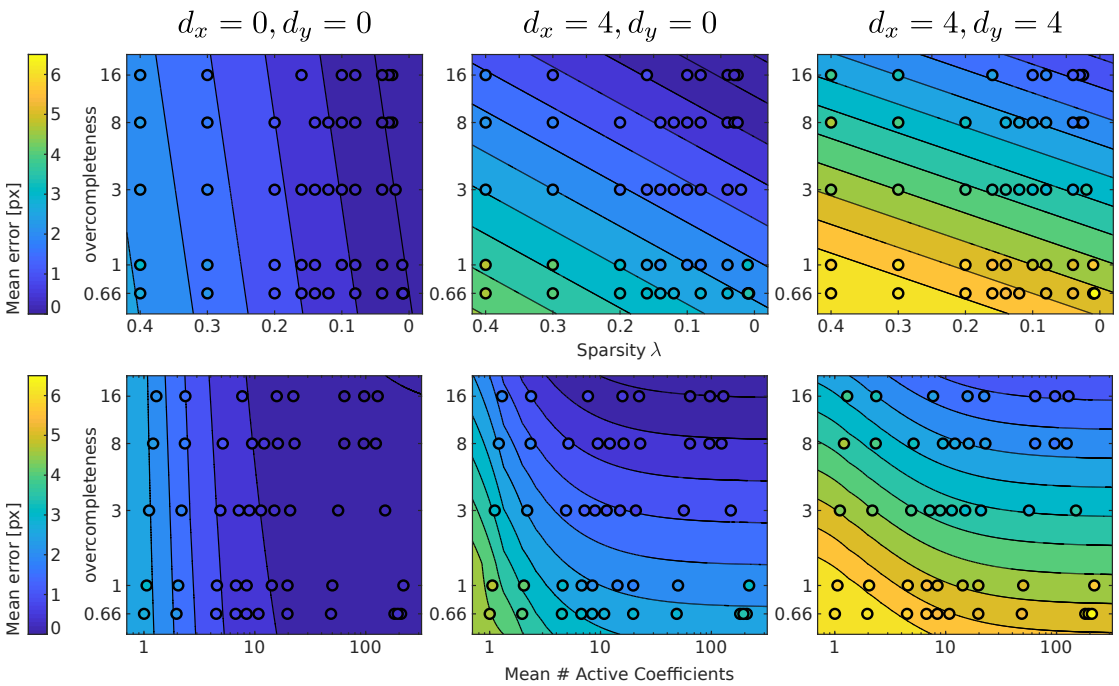


Figure 2.11. Top row panels show dependency of the mean inference error of disparity on overcompleteness (ordinate) and on sparsity load λ (abscissa). Bottom row panels show the same data but with λ mapped to the mean number of active coefficients. The three columns contain evaluations for three different disparities $d_{x,y}$. See Fig. 2.9 for the dependency of the mean error on disparity. Circles indicate evaluations of the mean absolute error (MAE) subject to overcompleteness o and sparsity load λ. Contours show the data fits of the error, with MAE $= a\,(\lambda + \Delta_\lambda) + b\,(\ln o + \Delta_o) + c$ (top row) and MAE $= a/(n + \Delta_n) + b\,(\ln o + \Delta_o) + c$ (bottom row). Close to zero disparity, overcompleteness has little impact, but it becomes increasingly important for larger disparities. The error generally declines with decreasing lambda. See the same data as line plots in supplementary Figs. S01–S06.

the LCA optimization as described in Sec. 2.4.2, and with probabilistic readout as described in Sec. 2.4.3. The mean absolute errors (MAE) of the estimates were calculated with

$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n} \| \boldsymbol{y}_j - \hat{\boldsymbol{y}}_j \| \,, \tag{2.17}$$

where $\boldsymbol{y}_j$ and $\hat{\boldsymbol{y}}_j$ were the ground truth and the the estimate, respectively. In Sec. 2.5.2–2.5.2, we report the MAE of the disparity estimates $\hat{\boldsymbol{y}}_j = (\hat{d}_x\ \hat{d}_y)^\intercal$. Inference was carried out on the test set from the disparity database with shifted images, described in detail in Sec. 2.3.2. In Sec. 2.5.2 we report results on inference of horizontal disparity in naturalistic stereo images.

Higher dimensionality extends the set of detectable patterns

An increase of overcompleteness generally resulted in a decrease of disparity inference errors. The best parameter combination from our evaluation (16× overcomplete, $\lambda = 0.04$) allowed for a mean disparity error below 0.5 px, measured within the range of ∼2–3 px ground truth absolute disparity (Fig. 2.9, right panel). Inference was better for small disparities than for large disparities. The same model performed with an error of ∼1.5 px for disparity of $d_x = 4$ px horizontally and $d_y = 4$ px vertically. The bias was generally small (data not shown) and apparent only at large disparities close to the cut-off at 6 px.

Overcompleteness had its main impact on the range of disparities for which the model performed well. With small overcompleteness, the error increased much more rapidly with the value of disparity. For example, the 1× overcomplete model with $\lambda = 0.04$ evaluated with an error below 1 px within the range of ±1 px disparity but with an error ∼6 px at $d_{x,y} = 4$ px horizontal and vertical disparity (Fig. 2.9, left panel). We show all parameter combinations we tested in the overview in Fig. 2.11. The same data is shown as line plots in supplementary Figs. S01–S06. For low levels of overcompleteness ($d_{x,y} = 0$ px, left-hand column of the plot), error dependency on overcompleteness is negligible, whereas for large overcompleteness ($d_{x,y} = 4$ px, right-hand column), overcompleteness has a substantial effect.

This finding was due to qualitative differences in the sets of learned kernels. With larger overcompleteness, more kernels existed with larger position-shift, i.e., with the differences in position between left and right half-kernel (Fig. 2.10). The distribution of the kernels' disparity was roughly similar to the distribution of disparities in stereo images (see Fig. 2.2), with many kernels that represent small disparities and few kernels that represent large disparities. We conclude that, with more overcompleteness, sparse coding extends the set of patterns that are represented explicitly, ordered by the frequency of their occurrence.

Less sparsity results in lower errors

The sparsity load λ was generally linked to better inference the *lower* its value. Up to a limit of very low values for λ, this is true for all levels of overcompleteness and for all ground truth values of disparity, as can be seen in the top row of Fig. 2.11. Our results are in line with the results from Rigamonti et al. (2011) and from Gardner-Medwin and Barlow (2001) (see Sec. 2.2). The bottom row of Fig. 2.11

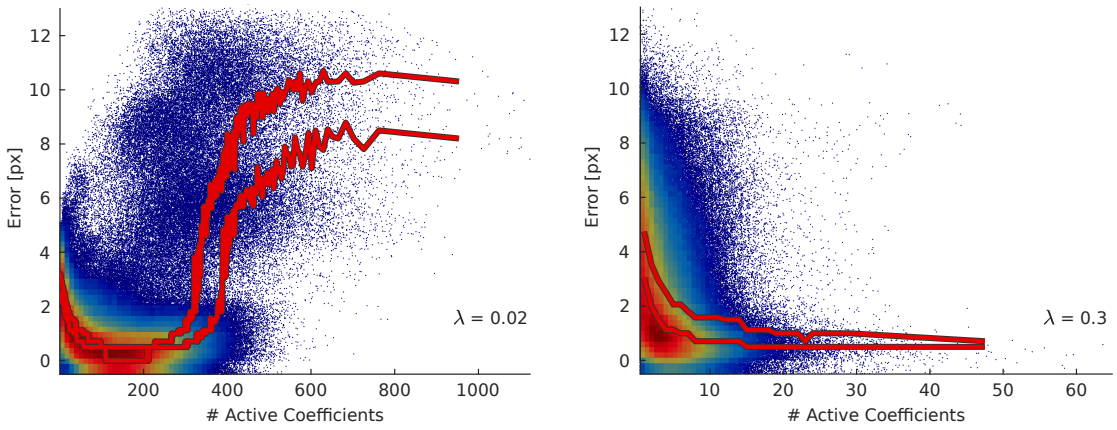


Figure 2.12. Data from 3× overcomplete model. The error of the inference is related to the mean number of active LCA coefficients. Data points represent the errors of single disparity inferences against the number of active coefficients. To counteract rasterization, they were displaced randomly by a small amount. The heat-map overlay is a density histogram (arbitrary units). Red lines: median and 75th percentile of error, calculated on bins of the number of active coefficients (at least 10^3 data points per bin). *Left*: With low sparsity penalty $\lambda = 0.02$, mid-range activity predicts the lowest error, as opposed to a small, or a large number of active coefficients. *Right*: With increased sparsity load $\lambda = 0.3$, a larger number of active neurons is no longer associated with poor performance. Note that overall activity is substantially reduced.

contains the same data as the top row, but with the sparsity load λ mapped to the mean number of active coefficients. Activity was roughly linked by a negative exponential to the range of λ we tested.

In all models except the 16× overcomplete model, we observed slightly increasing errors if sparsity load was very low. For most combinations of overcompleteness and disparity that we evaluated, the lowest mean error was measured at $\lambda \approx 0.04$. The error was below $\lambda = 0.1$ in all cases but one—the 0.66× overcomplete model, measured at 4 px horizontal disparity and 0 px vertical disparity. The minima can be examined in detail in the supplementary Figs. S01–S06. We therefore reject the hypothesis that inference is optimal if the sparsity penalty used during testing matches that used during training. A possible explanation is based on the fact that a binary multi channel code carries most information if the probability of the coefficients to be in one of both states is $p = 0.5$ and independent of other dimensions. (Shannon, 1948). Therefore, assuming that the coefficients were independent, the code carried most information if half of the coefficients were in an active state on average (42.5 for 0.6× overcomplete, 64 for 1× overcomplete, and

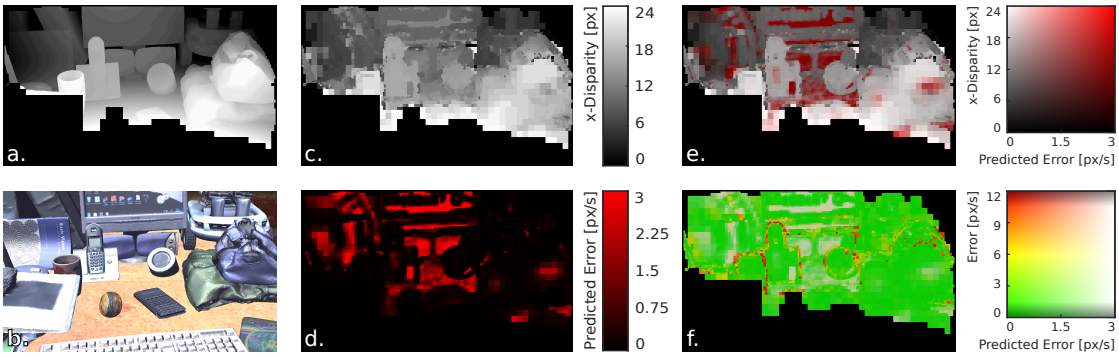


Figure 2.13. Disparity inference of a naturalistic scene from the Genua Pesto database (Canessa et al., 2017). **a)** Ground truth map of horizontal disparity. Values outside the range $[-24, 24]$ px were excluded. **b)** Right stereo half-image. **c)** Inference of horizontal disparity with the 16× overcomplete, $\lambda = 0.04$ processing pipeline. Due to the scale space approach, resolution is better the closer objects are to the horopter. **d)** Errors predicted by the number of active coefficients $a_{k,m',n'}$, relative to scale $s = 1$, 0.75, 0.5, 0.25 in subplots **d**–**f**. **e)** Overlay of disparity map **c** and predicted error **d**. **f)** Error of inference vs. predicted error. Both values are strongly correlated with $r = 0.58$ for predicted errors > 1 px.

192 for 3× overcomplete). Inference was best slightly below these numbers, which shows in the bottom row of Fig. 2.11. It was not possible to confirm this finding for larger overcompleteness or larger values of λ, because very high sparsity load was computationally prohibitive.

The reasoning that Shannon information is the limiting factor is ambivalent. On the one hand, imposing less weight on sparsity in Eq. 2.7 in turn imposes more weight on the reconstruction constraint, and therefore the preservation of information. On the other hand, information of an overcomplete representation is highly redundant. It is opposed to a compressed code that maximizes Shannon entropy (Field, 1994). However, we binarized the output of the LCA sparse coding before inference, which removed much information from each dimension. Therefore, information content was strongly limited if only a few coefficients were in an active state.

The number of active LCA coefficients predicts the accuracy of inference

We encountered a strong relation between the success of disparity inference and the number of active coefficients. We assessed this relation by sorting responses to examples from the disparity database test set, ordered by the number of active

LCA coefficients $a_{k,m',n'}$. The data were binned with a window size of at least 10^3 data points. Note that the bin size was unequal, due to this constraint. Finally, we calculated percentiles of the MAE of disparity inference. Resulting histograms are shown as the red lines in Fig. 2.12. The data points of the disparity inference error are plotted in the same diagram, with a heatmap overlay that displays density where the point cloud is very dense (arbitrary units).

The median error as a function of the number of active coefficients was u-shaped. Therefore, inference was best when an average number of coefficients was in an active state. A low number, as well as a large number of active coefficients was a predictor for large errors (Fig. 2.12, left panel). With a large value of sparsity load λ, the number of active coefficients was greatly reduced (right panel). In this case, the median error was monotonically decreasing as a function of the number of active coefficients.

We assume that a low number of active coefficients was associated to large errors because the few tuning maps did not contain enough information for accurate inference. The finding could simply account for the absence of structure in the image. An explanation for the association of a large number of active coefficients with large errors is not so straight forward. We hypothesize that the sparse optimization was not able to settle on a good representation and therefore reconstructed the input with much more kernels than on average. These kernels were not well suited for the given image structures and therefore only active due to the lack of better representatives. Our perspective is linked to a study from Froudarakis et al. (2014). They report that the stimulation with phase scrambled movies activates mouse V1 more strongly than the stimulation with natural movies. Simultaneous recordings from a large population of cells were analyzed for discriminability of the presented movies with a linear classifier. Similar to our finding, strong activation was a predictor for bad classification performance.

Disparity map of a naturalistic scene

In addition to inference with constant disparity, i.e., with shifted images, we evaluated our visual processing pipeline with a naturalistic scene from the Genua Pesto database (Canessa et al., 2017). We present results from one of the scenes in Fig. 2.13. It consists of disparities in the interval $[-76.7, 77.1]$ px, as opposed to our model, which is limited to inference in the interval $[-6, 6]$ px. We faced the

limitation of the model with a scale-space approach, by downsampling the input image to 80 %, 60 %, 40 % and 20 %. Inference was then only evaluated within the interval $[-6, 6]$ px at each of these four scales, and with the best available spatial resolution at each location. Image locations outside the interval were excluded beforehand. All experiments were carried out with the 16× overcomplete, $\lambda = 0.04$ model.

Disparity was inferred well within the aforementioned limitations. The disparity map we obtained is shown in Fig. 2.13c (compare to ground truth disparity map in Fig. 2.13a). Note that the map is an overlay of the four scales, with best spatial resolution close to the horopter. We chose to plot the predicted error (Fig. 2.13d) as the 80th percentile of the error as a function of the number of active coefficients, as described in Sec. 2.5.2. Errors in subplots d–f are relative to the scale $s = 1$, 0.75, 0.5, and 0.25. Fig. 2.13e is an overlay of the disparity map with the predicted error. The prediction corresponds to clearly identifiable structures in the image. Large errors were predicted for the loudspeakers, for the table texture, and for uniformly colored locations on the monitor. Low errors were predicted for the telephone, for the bags on the right, and for the icons on the monitor. Fig. 2.13f visualizes the predicted error and the actual error with respect to ground truth disparity in one plot. If prediction failed, this was mostly due to occlusion boundaries. Note that occlusion boundaries were not part of the training, so this type of error can not be attributed to the lack of representation in the LCA optimization.

2.5.3 Tuning maps of surface orientation

We showed that a representation formed by LCA sparse coding forms a suitable basis to infer stereo disparity. However, we hypothesized that sparse coding fulfills the requirement for simple readout of a much larger set of patterns. As a second example, we examined tuning maps for tilt- and slant angles φ and α of a textured surface (see Sec. 2.2.4). Results were based on the test set from the surface orientation database, described in detail in Sec. 2.3.2.

We created tuning maps, not only for each kernel $\mathbf{\Phi}_{L/R,k}$, but for each of 7×7 entries from the convolutional feature maps with central fixation point. This decision was based on the expectation that the tuning maps were affected by the disparity tuning of the kernels. We reasoned that surface orientation could be inferred from a set of disparity measurements at positions relative to the fixation

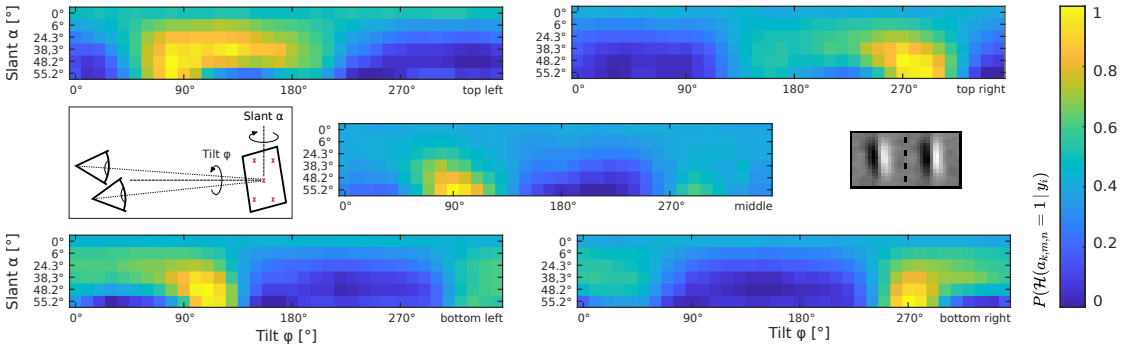


Figure 2.14. Tuning maps for tilt- and slant angles φ and α of textured surfaces. All maps correspond to a single kernel $\mathbf{\Phi}_{L/R,k}$ (grayscale) but with receptive fields at varying feature map locations w.r.t. the central fixation point. Feature map locations of the four tuning maps are indicated with red crosses in the illustration. Data from $1\times$ overcomplete model, with $\lambda = 0.12$.

point. In Fig. 2.14 we show that kernels were tuned for surface tilt- and slant angles. Indeed, the tuning maps of the kernels differed, depending on the position at which it was evaluated. The peak of the tuning maps was the sharper the larger the slant angle of the surface. In tilt-/slant space, the peak described a skewed band, which was expected if disparity tuning was the underlying principle.

Coefficients in the center of the tuning maps, which corresponded to the fixation point, were also clearly tuned for the surface tilt angle (see central tuning map of Fig. 2.14 for an example). They could not be affected by disparity because disparity is zero at the fixation point, irrespective of the surface orientation. Instead, the mode of the tuning for the tilt angle φ was strongly related to the kernels' orientation ϕ, with a circular-circular correlation coefficient $\rho_{cc} = -0.981$ (calculated following Jammalamadaka and SenGupta (2001), using CircStat (Berens, 2009)). A scatterplot of ϕ against φ is displayed in Fig. 2.15. Fleming et al. (2004) showed that a set of Gabor-filters can be used to infer surface orientation in monocular images. In images of slanted, textured surfaces, spatial frequencies that are oriented perpendicular to the tilt angle of the surface are overrepresented. This is due to the homographic projection on the retina, which causes an anisotropic compression of surface textures. The finding qualitatively extends the set of patterns that can be inferred from the LCA representation. It adds information to the inference of surface orientation that is different from the inference based on the local distribution of disparities.

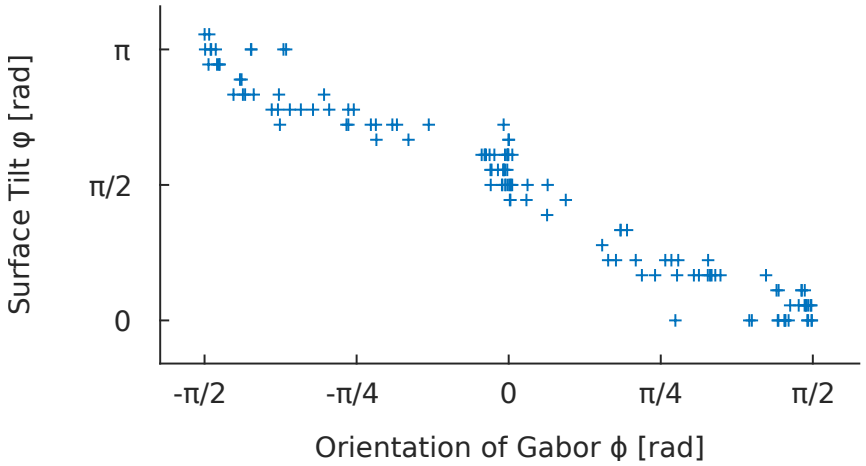


Figure 2.15. The tuning maps' mode of the tilt angle φ against the orientation ϕ of kernels $\mathbf{\Phi}_{L/R,k}$. All tuning maps were evaluated at the central fixation point with zero disparity. Orientation of the Gabor-like kernels accounts for surface tilt tuning, with very strong circular-circular correlation $\rho_{cc} = -0.981$.

2.6 Discussion

We add evidence to an existent body of literature, which shows that Gabor-like, disparity tuned, phase- and position-shifted receptive fields are a good basis for stereo algorithms (see Sec. 2.2.3 and 2.2.4). Simple readout of disparity was possible, due to some degree of selectivity to the stimulus of units from the LCA representation. These units therefore resembled Barlow's *cardinal cells*, with intermediate selectivity for the stimulus. Indeed, we did not observe a single kernel that was not tuned for disparity or surface orientation. At the same time, they represented a variety of other stimulus aspects, like spatial frequency, orientation, or blob-like structures. In combination, the kernels represented the input space well and allowed for accurate inference.

2.6.1 Dimensionality of representations

Larger dimensionality of a representation extended the range of disparities that could be inferred with simple readout. We offer an intuitive explanation for this finding. Local structure in both half-images that originate from the same location in the world can either be represented by one binocular kernel with similar left and right shape; or it can be represented by two kernels: a left and a right monocular

kernel. If binocular kernels are available, sparsity can increase substantially, by activation of only half the number of units that would be needed for reconstruction with monocular kernels. However, a representation that contains binocular kernels requires a much larger dimensionality.

Lets assume that we want to create a new binocular representation from $2\,n$ monocular kernels, which consists of copies of the same set of n monocular half-kernels for each eye. We generate binocular kernels from all monocular kernels, so that the left half-kernels are shifted versions of the right half-kernels. If we assume equally spaced horizontal and vertical shifts in the range $\|\Delta \boldsymbol{d}_{x,y}\|$, the new representation has $\sim n\,c\,\pi/4\,\|\Delta \boldsymbol{d}_{x,y}\|^2$ kernels, with factor c that determines the resolution. Because occlusions are characterized by the lack of corresponding structure, we would have to add the original monocular kernels to the representation, so that the total number of kernels would be $\sim n\,(2\,+\,c\,\pi/4\,\|\Delta \boldsymbol{d}_{x,y}\|^2)$. With either set of kernels, the optimization can reconstruct the image equally well, but much more sparsely with the larger set. Obviously, the amount of information is the same in both representations, because the information preservation constraint of the optimization is not affected. However, the larger representation is much more redundant. Interestingly, it is exactly this redundancy that allows for inference, because binocular kernels that fit corresponding features are tuned for disparities.

With large dimensionality, it was possible to infer disparity with binary classification, even though the binarization discards information (compare Bobrowski (2011)). Burge and Geisler presented an opposed approach to disparity inference, with very low dimensionality (Burge & Geisler, 2014). They asked which filter shapes were optimal to infer disparity and showed that inference was possible with the two most informative kernels. In their model, the activity ratio of these detectors was the crucial parameter for inference. Information was not distributed over many binary dimensions but encoded in the value of a few dimensions.

In biological systems, the value may be encoded in the firing rates of neurons. Fine grained discriminability between neural activities, i.e., large channel capacity, requires high firing rates. Indeed, there are many examples where neurons encode sensory information with high firing rates. Examples include medial superior olivary neurons, which lock precisely to the phase of pure tones (Brand et al., 2002), and the T-units in Gymnotiforms (weakly electric fish), which lock to the phase of electrical signals with up to almost 1000 Hz (Scheich et al., 1973). Although

cortical neurons operate at low mean firing rates of about 4 Hz (R. Baddeley et al., 1997), action potential bursts are known candidates to encode information in firing rates. For a current review on neural coding with bursts see (Zeldenrust et al., 2018).

2.6.2 The trade-off between accuracy and energy efficiency

An alternative explanation for the finding that cortical neurons exhibit sparse activity is energy efficiency. The energetic cost of a neuronal population has two major contributions: the maintenance of neurons, which limits population size, and neuronal activity, measured by the average rate of action potentials (Attwell & Laughlin, 2001; Lennie, 2003). Because neuronal activity is relatively costly, an optimization that takes energy efficiency into account results in reduced activity (Levy & Baxter, 1996).

Indeed, the two terms of the sparse coding optimization Eq. 2.7 are the preservation of information and the sparsity of the representation, weighted against each other with the sparsity load λ. We have shown that the mean inference error also depends on λ. We therefore hypothesize that sparsity in the brain is optimized for the trade-off between the accuracy of upstream processing tasks and energy consumption. This optimization could even occur dynamically and locally, as an attention mechanism that adjusts the error subject to the current task. Such a mechanism could interact with the error prediction we have shown, which relies on counting the number of active coefficients.

The realization in biological substrate is plausible. In neural notion of the LCA sparse coding, the sparsity load corresponds to a shift in the thresholds of neurons (see Sec. 2.4.4). Indeed, physiological studies show that attentional mechanisms involve changes in the excitability of neurons. McAdams and Maunsell (1999) have shown that attention modulates the response of orientation-tuned neurons in V4 multiplicatively. Similarly, tuning maps of LCA kernels were qualitatively indifferent with respect to λ. Therefore, an interesting question for future research is whether inference with variable LCA thresholds and static weights for readout is feasible.

2.6.3 Model-specific issues

Applying our processing pipeline to the naturalistic scene, image locations with disparities larger or smaller than the disparities included in the training set yielded random results. We are confident that an additional category that includes all of the disparities beyond the included range could successfully be added to the training set. The category could rely on activity of coefficients of the "Tuned Inhibitory" type and on lack of activity of the "Matched Gabor" type. Occluded image regions are similarly characterized by the lack of corresponding image structure and might be represented by the same kernel types. Whether it is possible to distinguish between a large-disparity category and an occlusion category is an interesting question for future research.

The resolution of the disparity maps was limited in this study. We used a stride of 8 px for the convolutional LCA sparse coding, which was therefore also the downsampling factor for the disparity map. With the same level of overcompleteness, a larger stride corresponds to a larger number of kernels (Schultz et al., 2014). We assume that a large number of kernels is mandatory in order to represent a large number of disparities. However, we expect that the resolution of the disparity estimates does not depend on the stride. Tuning maps of coefficients in a single column of the feature maps most likely vary with respect to the position of the image structure in their receptive fields. We may explore the limits of the spatial resolution in future research.

We have presented a naturalistic processing pipeline for disparity inference. Our aim was not to find a method which has the lowest inference error, but to learn more about inference based on sparse representations in general. However, we had reasonable success of inferring disparities in a naturalistic scene. With recent progress on neuromorphic hardware, as well as progress on efficient implementations of the spiking LCA algorithm (Tang et al., 2017; Watkins et al., 2019; Zylberberg et al., 2011), research on the hardware implementation of our biologically inspired stereo vision processing stream would be promising and is within reach.

2.6.4 Sparse coding and supervision

Because more overcompleteness extends the set of patterns that can be inferred, the set of patterns that is ecologically relevant may predict the extend of the neuronal population in animals. Patterns, too rare to be represented, subject to the cost of neuronal maintenance, should be omitted. The likelihood that sparse coding represents patterns explicitly may depend on the frequency of their occurrence. The distribution might be divergent from the relevance of the patterns an animal needs to detect. Common patterns may be irrelevant while rare patterns, like cues that reveal the attack of a lurking predator, are essential for survival. In the case of depth inference, a uniform accuracy over the whole disparity range might be optimal. An advantageous learning strategy could profit from the generality of feature extraction based on sensory statistics and augment learning with mild supervision in order to gently shift the representation towards a distribution optimized for behavioral gain. In multi-layered networks, later stages might also benefit from incorporating sparsity constraints, by aiding the clustering towards conceptional representations. Current research supports this assumption. E. Kim et al. (2018) have shown that a standard autoencoder, augmented with lateral inhibition and top-down feedback, develops joined representations of multimodal input data. Hale Berry Neurons were responsive for textual, as well as for visual input. The representation was easily separable and robust for classification tasks.

2.6.5 The link between image statistics and inference

It remains an open question why a method that extracts statistical properties from natural images yields good features for inference. The original perspective on the independent component analysis (ICA), a class of algorithms to which sparse coding belongs, might point towards a possible explanation. The reasoning behind ICA was that data from sensor arrays are in some cases the weighted superposition of a number of individual, independent source signals (Hyvärinen & Oja, 2000). If the superposition is linear, source signals can be reconstructed by multiplying the vector of sensory data with the inverse of the weight matrix. The aim of ICA is to find this inverse matrix.

Clearly, the assumption that sensory data are the weighted sum of source signals is not true for the formation of two-dimensional images on the retina. Images

originate from light rays scattered by objects within a physical, three-dimensional world. The components obtained by ICA are in fact not independent of each other (Bethge, 2006; Eichhorn et al., 2009). They are not the building blocks of an image and the task of inferring depth is not readily solved by extracting these components. However, they seem to coincide with physical causes. The distance of objects manifests in the shift of corresponding image structure, occlusions manifest in the lack of corresponding image structure, and surface orientation manifests in anisotropically compressed texture. Obviously, even though the feature dimensions are not the original components of the image, they are closely linked to the geometrical layout of the scene and therefore allow to infer properties of the external world. They might pose the basis for a heuristic mental model of the external world, established by the clustering of "suspicious coincidences" (H. Barlow, 1987; Földiák, 1990).

We believe that the selectivity for patterns that are linked to physical causes is a general property of sparse representations of sensory data. For example, we have recently shown that applying sparse coding to optic flow data yields rather unexpected kernel shapes, which are tuned to directions of egomotion (Ecke et al., 2020). Screening for such selectivities can be a starting point for identifying the cues that are at the core of inference and it can yield predictions for properties of processing in diverse biological systems.

2.7 Conclusion

With this study, we have extended the knowledge about similarities and differences between representations learned with stereo sparse coding and the visual cortex. We have also shown that statistical properties of the visual sensory stream can be exploited with the sparse coding algorithm and consecutive simple readout of depth parameters. Disparity can be inferred reasonably well, with very good accuracy for low disparities but with increasing error the larger the disparity. The range of disparities that can be inferred with good accuracy grows with overcompleteness. More sparsity reduces the accuracy of inference. Since neuronal activity is directly associated with energy consumption, attentional mechanisms could optimize the trade-off between energy efficiency and the accuracy needed for the task an animal faces. In addition, we have shown that accuracy of the inference can be inferred

from the number of active LCA coefficients itself. The estimate could be used as a feedback parameter to adjust the sparsity of the optimization.

We hypothesized that sparse coding transforms the sensory stream such that an unknown subset of patterns from the external world can be inferred by subsequent, simple readout. After a thorough analysis of disparity inference, we have shown that the representation also carries information that allows to infer surface orientation. Selectivity for this subset of patterns is qualitatively different from disparity tuning because it depends on the orientation of the Gabor-like kernels shapes. We believe that sparse coding generalizes properties from the external world and can be used to infer a much broader range of patterns that are cues for physical causes.

References

Abbott, L. F. (1999). Lapicque's introduction of the integrate-and-fire model neuron (1907) [Publisher: Elsevier BV]. *Brain Research Bulletin*, *50*(5-6), 303–304. https://doi.org/10.1016/s0361-9230(99)00161-6

Anzai, A., Ohzawa, I., & Freeman, R. D. (1999). Neural mechanisms for encoding binocular disparity: Receptive field position versus phase. *Journal of Neurophysiology*, *82*(2), 874–890. https://doi.org/10.1152/jn.1999.82.2.874

Atick, J. J., & Redlich, A. N. (1992). What does the retina know about natural scenes? [Number: 2 Reporter: Neural computation]. *Neural computation*, *4*(2), 196–210. https://doi.org/10.1162/neco.1992.4.2.196

Attwell, D., & Laughlin, S. B. (2001). An Energy Budget for Signaling in the Grey Matter of the Brain [Publisher: SAGE Publications]. *Journal of Cerebral Blood Flow & Metabolism*, *21*(10), 1133–1145. https://doi.org/10.1097/00004647-200110000-00001

Baddeley, R., Abbott, L. F., Booth, M. C. A., Sengpiel, F., Freeman, T., Wakeman, E. A., & Rolls, E. T. (1997). Responses of neurons in primary and inferior temporal visual cortices to natural scenes. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, *264*(1389), 1775–1783. https://doi.org/10.1098/rspb.1997.0246

Barlow, H. (1987). Cerebral Cortex as Model Builder [Reporter: Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience]. In L. M. Vaina

(Ed.), *Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience* (pp. 395–406). Springer Netherlands. https://doi.org/10.1007/978-94-009-3833-5_18

Barlow, H. (2001). The exploitation of regularities in the environment by the brain [Number: 04 Reporter: Behavioral and Brain Sciences]. *Behavioral and Brain Sciences*, *24*(04). https://doi.org/10.1017/S0140525X01000024

Barlow, H. B. (1959). Sensory mechanisms, the reduction of redundancy and intelligence. *National Physical Laboratory Symposium No. 10, The Mechanisation of Thought Processes. Her Majesty's Stationery Office, London.*

Barlow, H. B. (1972). Single Units and Sensation: A Neuron Doctrine for Perceptual Psychology? [Number: 4 Reporter: Perception]. *Perception*, *1*(4), 371–394. https://doi.org/10.1068/p010371

Barlow, H. B. (2001). Redundancy reduction revisited [Publisher: Informa UK Limited]. *Network: Computation in Neural Systems*, *12*(3), 241–253. https://doi.org/10.1080/net.12.3.241.253

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008b). Speeded-up robust features (SURF). *Computer vision and image understanding*, *110*(3), 346–359. https://doi.org/10.1016/j.cviu.2007.09.014

Berens, P. (2009). CircStat: AMATLABToolbox for Circular Statistics [Publisher: Foundation for Open Access Statistic]. *Journal of Statistical Software*, *31*(10). https://doi.org/10.18637/jss.v031.i10

Bethge, M. (2006). Factorial coding of natural images: How effective are linear models in removing higher-order dependencies? [Publisher: The Optical Society]. *Journal of the Optical Society of America A*, *23*(6), 1253. https://doi.org/10.1364/josaa.23.001253

Bhatt, V., & Ganguly, U. (2018). Sparsity Enables Data and Energy Efficient Spiking Convolutional Neural Networks. In V. Kůrková, Y. Manolopoulos, B. Hammer, L. Iliadis, & I. Maglogiannis (Eds.), *Artificial Neural Networks and Machine Learning – ICANN 2018* (pp. 263–272). Springer International Publishing. https://doi.org/10.1007/978-3-030-01418-6_26

Blakemore, C., & Tobin, E. A. (1972). Lateral inhibition between orientation detectors in the cat's visual cortex [Number: 4 Reporter: Experimental Brain Research]. *Experimental Brain Research*, *15*(4), 439–440. https://doi.org/10.1007/BF00234129

Bobrowski, L. (2011). Induction of Linear Separability through the Ranked Layers of Binary Classifiers. *Engineering Applications of Neural Networks* (pp. 69–77). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-23957-1_8

Brand, A., Behrend, O., Marquardt, T., McAlpine, D., & Grothe, B. (2002). Precise inhibition is essential for microsecond interaural time difference coding [Publisher: Springer Science and Business Media LLC]. *Nature*, *417*(6888), 543–547. https://doi.org/10.1038/417543a

Burge, J., & Geisler, W. S. (2014). Optimal disparity estimation in natural stereo images [Number: 2 Reporter: Journal of Vision]. *Journal of Vision*, *14*(2), 1–1. https://doi.org/10.1167/14.2.1

Canessa, A., Gibaldi, A., Chessa, M., Fato, M., Solari, F., & Sabatini, S. P. (2017). A dataset of stereoscopic images and ground-truth disparity mimicking human fixations in peripersonal space. *Scientific Data*, *4*, 170034. https://doi.org/10.1038/sdata.2017.34

Chauhan, T., Masquelier, T., Montlibert, A., & Cottereau, B. R. (2018). Emergence of Binocular Disparity Selectivity through Hebbian Learning [Publisher: Society for Neuroscience]. *The Journal of Neuroscience*, *38*(44), 9563–9578. https://doi.org/10.1523/jneurosci.1259-18.2018

Doya, K., Ishii, S., Pouget, A., & Rao, R. P. N. (Eds.). (2006). *Bayesian Brain*. The MIT Press. https://doi.org/10.7551/mitpress/9780262042383.001.0001

Ecke, G. A., Bruijns, S. A., Hölscher, J., Mikulasch, F. A., Witschel, T., Arrenberg, A. B., & Mallot, H. A. (2020). Sparse coding predicts optic flow specificities of zebrafish pretectal neurons. *Neural Computing and Applications*, *32*(11), 6745–6754. https://doi.org/10.1007/s00521-019-04500-6

Eichhorn, J., Sinz, F., & Bethge, M. (2009). Natural Image Coding in V1: How Much Use Is Orientation Selectivity? (L. Zhaoping, Ed.) [Number: 4 Reporter: PLoS Computational Biology]. *PLoS Computational Biology*, *5*(4), e1000336. https://doi.org/10.1371/journal.pcbi.1000336

Falconbridge, M. S., Stamps, R. L., & Badcock, D. R. (2006). A Simple Hebbian / Anti-Hebbian Network Learns the Sparse, Independent Components of Natural Images [Publisher: MIT Press - Journals]. *Neural Computation*, *18*(2), 415–429. https://doi.org/10.1162/089976606775093891

Field, D. J. (1987). Relations between the statistics of natural images and the response properties of cortical cells [Number: 12 Reporter: JOSA A]. *JOSA A*, *4*(12), 2379–2394. https://doi.org/10.1364/JOSAA.4.002379

Field, D. J. (1994). What Is the Goal of Sensory Coding? [Publisher: MIT Press - Journals]. *Neural Computation*, *6*(4), 559–601. https://doi.org/10.1162/neco.1994.6.4.559

Fleming, R. W., Torralba, A., & Adelson, E. H. (2004). Specular reflections and the perception of shape. *Journal of Vision*, *4*(9), 10. https://doi.org/10.1167/4.9.10

Földiák, P. (1990). Forming sparse representations by local anti-Hebbian learning [Publisher: Springer Science and Business Media LLC]. *Biological Cybernetics*, *64*(2), 165–170. https://doi.org/10.1007/bf02331346

Froudarakis, E., Berens, P., Ecker, A. S., Cotton, R. J., Sinz, F. H., Yatsenko, D., Saggau, P., Bethge, M., & Tolias, A. S. (2014). Population code in mouse V1 facilitates readout of natural scenes through increased sparseness [Number: 6 Reporter: Nature Neuroscience]. *Nature Neuroscience*, *17*(6), 851–857. https://doi.org/10.1038/nn.3707

Gardner-Medwin, A. R., & Barlow, H. B. (2001). The Limits of Counting Accuracy in Distributed Neural Representations [Publisher: MIT Press - Journals]. *Neural Computation*, *13*(3), 477–504. https://doi.org/10.1162/089976601300014420

Goncalves, N. R., & Welchman, A. E. (2017). "What Not" Detectors Help the Brain See in Depth. *Current Biology*, *27*(10), 1403–1412.e8. https://doi.org/10.1016/j.cub.2017.03.074

Graham, D. J., Chandler, D. M., & Field, D. J. (2006). Can the theory of "whitening" explain the center-surround properties of retinal ganglion cell receptive fields? [Number: 18 Reporter: Vision Research]. *Vision Research*, *46*(18), 2901–2913. https://doi.org/10.1016/j.visres.2006.03.008

Guillemot, J.-P., Paradis, M.-C., Samson, A., Ptito, M., Richer, L., & Lepore, F. (1993). Binocular interaction and disparity coding in area 19 of visual cortex in normal and split-chiasm cats. *Experimental brain research*, *94*(3), 405–417. https://doi.org/10.1007/BF00230199

Hand, D. J., & Yu, K. (2001). Idiot's Bayes—not so stupid after all? [Publisher: Wiley Online Library]. *International statistical review*, *69*(3), 385–398. https://doi.org/10.1111/j.1751-5823.2001.tb00465.x

Hartley, R., & Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press. https://doi.org/10.1017/cbo9780511811685

Henriksen, S., Tanabe, S., & Cumming, B. (2016). Disparity processing in primary visual cortex. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *371*(1697), 20150255. https://doi.org/10.1098/rstb.2015.0255

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. [Publisher: Proceedings of the National Academy of Sciences]. *Proceedings of the National Academy of Sciences*, *79*(8), 2554–2558. https://doi.org/10.1073/pnas.79.8.2554

Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the national academy of sciences*, *81*(10), 3088–3092. https://doi.org/10.1073/pnas.81.10.3088

Hoyer, P. O., & Hyvärinen, A. (2000). Independent component analysis applied to feature extraction from colour and stereo images [Number: 3 Reporter: Network: Computation in Neural Systems]. *Network: Computation in Neural Systems*, *11*(3), 191–210. https://doi.org/10.1088/0954-898X_11_3_302

Hubel, D. H., & Wiesel, T. N. (1970). Stereoscopic Vision in Macaque Monkey: Cells sensitive to Binocular Depth in Area 18 of the Macaque Monkey Cortex. *Nature*, *225*(5227), 41–42. https://doi.org/10.1038/225041a0

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex [Publisher: Wiley Online Library]. *The Journal of physiology*, *160*(1), 106–154. https://doi.org/10.1113/jphysiol.1962.sp006837

Hubel, D. H., Wiesel, T. N., Yeagle, E. M., Lafer-Sousa, R., & Conway, B. R. (2015). Binocular Stereoscopy in Visual Areas V-2, V-3, and V-3A of the Macaque Monkey. *Cerebral Cortex*, *25*(4), 959–971. https://doi.org/10.1093/cercor/bht288

Hunt, J. J., Dayan, P., & Goodhill, G. J. (2013). Sparse Coding Can Predict Primary Visual Cortex Receptive Field Changes Induced by Abnormal Vi-

sual Input (M. Bethge, Ed.). *PLoS Computational Biology*, *9*(5), e1003005. https://doi.org/10.1371/journal.pcbi.1003005

Hunter, D. W., & Hibbard, P. B. (2015). Distribution of independent components of binocular natural images [Number: 13 Reporter: Journal of Vision]. *Journal of Vision*, *15*(13), 6. https://doi.org/10.1167/15.13.6

Hyvärinen, A., Oja, E., Hoyer, P., & Hurri, J. (1998). Image feature extraction by sparse coding and independent component analysis. *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170)*. https://doi.org/10.1109/icpr.1998.711932

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications [Number: 4 Reporter: Neural networks]. *Neural networks*, *13*(4), 411–430. https://doi.org/10.1016/S0893-6080(00)00026-5

Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2017). FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2017.179

Jammalamadaka, S. R., & SenGupta, A. (2001). *Topics in Circular Statistics*. WORLD SCIENTIFIC. https://doi.org/10.1142/4031

Jones, J. P., & Palmer, L. A. (1987). An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex [Publisher: American Physiological Society]. *Journal of Neurophysiology*, *58*(6), 1233–1258. https://doi.org/10.1152/jn.1987.58.6.1233

Kalberlah, C., Distler, C., & Hoffmann, K.-P. (2009). Sensitivity to relative disparity in early visual cortex of pigmented and albino ferrets. *Experimental Brain Research*, *192*(3), 379–389. https://doi.org/10.1007/s00221-008-1545-z

Kim, E., Hannan, D., & Kenyon, G. (2018). Deep Sparse Coding for Invariant Multimodal Halle Berry Neurons [Reporter: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition]. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/cvpr.2018.00122

King, P. D., Zylberberg, J., & DeWeese, M. R. (2013). Inhibitory Interneurons Decorrelate Excitatory Cells to Drive Sparse Code Formation in a Spiking Model of V1 [Number: 13 Reporter: Journal of Neuroscience]. *Journal of*

Neuroscience, *33*(13), 5475–5485. https://doi.org/10.1523/JNEUROSCI.4188-12.2013

Knill, D. C., & Pouget, A. (2004). The Bayesian brain: The role of uncertainty in neural coding and computation [Number: 12 Reporter: Trends in Neurosciences Publisher: Elsevier BV]. *Trends in Neurosciences*, *27*(12), 712–719. https://doi.org/10.1016/j.tins.2004.10.007

Kuncheva, L. I. (2006). On the optimality of Naïve Bayes with dependent binary features. *Pattern Recognition Letters*, *27*(7), 830–837. https://doi.org/10.1016/j.patrec.2005.12.001

Kupervasser, O. (2014). The mysterious optimality of Naive Bayes: Estimation of the probability in the system of "classifiers". *Pattern Recognition and Image Analysis*, *24*(1), 1–10. https://doi.org/10.1134/S1054661814010088

Lennie, P. (2003). The Cost of Cortical Computation [Publisher: Elsevier BV]. *Current Biology*, *13*(6), 493–497. https://doi.org/10.1016/s0960-9822(03)00135-0

Levay, S., Stryker, M. P., & Shatz, C. J. (1978). Ocular dominance columns and their development in layer IV of the cat's visual cortex: A quantitative study. *Journal of Comparative Neurology*, *179*(1), 223–244. https://doi.org/10.1002/cne.901790113

Levy, W. B., & Baxter, R. A. (1996). Energy Efficient Neural Codes [Publisher: MIT Press - Journals]. *Neural Computation*, *8*(3), 531–543. https://doi.org/10.1162/neco.1996.8.3.531

Li, Z., & Atick, J. J. (1994). Efficient stereo coding in the multiscale representation [Number: 2 Reporter: Network: Computation in Neural Systems]. *Network: Computation in Neural Systems*, *5*(2), 157–174. https://doi.org/10.1088/0954-898X_5_2_003

Little, W. A. (1974). The Existence of Persistent States in the Brain. *From High-Temperature Superconductivity to Microminiature Refrigeration* (pp. 145–164). Springer US. https://doi.org/10.1007/978-1-4613-0411-1_12

Lonini, L., Forestier, S., Teulière, C., Zhao, Y., Shi, B. E., & Triesch, J. (2013). Robust active binocular vision through intrinsically motivated learning [Reporter: Frontiers in Neurorobotics]. *Frontiers in Neurorobotics*, *7*. https://doi.org/10.3389/fnbot.2013.00020

Lopez-Hazas, J., Montero, A., & Rodriguez, F. B. (2018). Strategies to Enhance Pattern Recognition in Neural Networks Based on the Insect Olfactory System. In V. Kůrková, Y. Manolopoulos, B. Hammer, L. Iliadis, & I. Maglogiannis (Eds.), *Artificial Neural Networks and Machine Learning – ICANN 2018* (pp. 468–475). Springer International Publishing. https://doi.org/10.1007/978-3-030-01418-6_46

Lundquist, S. Y., Mitchell, M., & Kenyon, G. T. (2017). Sparse Coding on Stereo Video for Object Detection. *arXiv preprint arXiv:1705.07144*.

Lundquist, S. Y., Paiton, D. M., Schultz, P. F., & Kenyon, G. T. (2016). Sparse encoding of binocular images for depth inference. *2016 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*. https://doi.org/10.1109/ssiai.2016.7459190

Mallot, H. A. (1999). Stereopsis: Geometrical and global aspects. *Handbook of computer vision and applications Vol. 2: Signal processing and pattern recognition* (pp. 485–502). San Diego: Academic Press.

Mallot, H. A., Roll, A., & Arndt, P. A. (1996). Disparity-evoked Vergence is Driven by Interocular Correlation [Publisher: Elsevier BV]. *Vision Research*, *36*(18), 2925–2937. https://doi.org/10.1016/0042-6989(96)00011-9

Marr, D., & Poggio, T. (1976). Cooperative computation of stereo disparity. *Science*, *194*(4262), 283–287. https://doi.org/10.1126/science.968482

Marr, D., & Poggio, T. (1979). A Computational Theory of Human Stereo Vision. *Proceedings of the Royal Society B: Biological Sciences*, *204*(1156), 301–328. https://doi.org/10.1098/rspb.1979.0029

McAdams, C. J., & Maunsell, J. H. R. (1999). Effects of Attention on Orientation-Tuning Functions of Single Neurons in Macaque Cortical Area V4 [Publisher: Society for Neuroscience]. *The Journal of Neuroscience*, *19*(1), 431–441. https://doi.org/10.1523/jneurosci.19-01-00431.1999

Ohzawa, I., DeAngelis, G., & Freeman, R. (1990). Stereoscopic depth discrimination in the visual cortex: Neurons ideally suited as disparity detectors [Publisher: American Association for the Advancement of Science (AAAS)]. *Science*, *249*(4972), 1037–1041. https://doi.org/10.1126/science.2396096

Olshausen, B. A. (2003). Principles of Image Representation in Visual Cortex. *The visual neurosciences, LM Chalupa, JS Werner, Eds* (pp. 1603–1615). MIT Press.

Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images [Number: 6583 Reporter: Nature]. *Nature*, *381*(6583), 607–609. https://doi.org/10.1038/381607a0

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? [Number: 23 Reporter: Vision Research]. *Vision Research*, *37*(23), 3311–3325. https://doi.org/10.1016/s0042-6989(97)00169-7

Orban, G. A. (2008). Higher Order Visual Processing in Macaque Extrastriate Cortex. *Physiological Reviews*, *88*(1), 59–89. https://doi.org/10.1152/physrev.00008.2007

Orban, G. A. (2011). The Extraction of 3D Shape in the Visual System of Human and Nonhuman Primates. *Annual Review of Neuroscience*, *34*(1), 361–388. https://doi.org/10.1146/annurev-neuro-061010-113819

Poggio, G. F., & Fischer, B. (1977). Binocular interaction and depth sensitivity in striate and prestriate cortex of behaving rhesus monkey [Publisher: American Physiological Society]. *Journal of Neurophysiology*, *40*(6), 1392–1405. https://doi.org/10.1152/jn.1977.40.6.1392

Poggio, G. F., Gonzalez, F., & Krause, F. (1988). Stereoscopic mechanisms in monkey visual cortex: Binocular correlation and disparity selectivity. *The Journal of neuroscience*, *8*(12), 4531–4550. https://doi.org/10.1523/JNEUROSCI.08-12-04531.1988

Poggio, G. F., & Talbot, W. H. (1981). Mechanisms of static and dynamic stereopsis in foveal cortex of the rhesus monkey. *The Journal of physiology*, *315*(1), 469–492. https://doi.org/10.1113/jphysiol.1981.sp013759

Read, J. C. A., & Cumming, B. G. (2007). Sensors for impossible stimuli may solve the stereo correspondence problem. *Nature Neuroscience*, *10*(10), 1322–1328. https://doi.org/10.1038/nn1951

Rehn, M., & Sommer, F. T. (2007). A network that uses few active neurones to code visual input predicts the diverse shapes of cortical receptive fields [Number: 2 Reporter: Journal of Computational Neuroscience]. *Journal of Computational Neuroscience*, *22*(2), 135–146. https://doi.org/10.1007/s10827-006-0003-9

Rigamonti, R., Brown, M. A., & Lepetit, V. (2011). Are sparse representations really relevant for image classification? *CVPR 2011*. https://doi.org/10.1109/cvpr.2011.5995313

Ringach, D. L. (2003). Dynamics of Orientation Tuning in Macaque V1: The Role of Global and Tuned Suppression. *Journal of Neurophysiology*, *90*(1), 342–352. https://doi.org/10.1152/jn.01018.2002

Ringach, D. L. (2002). Spatial Structure and Symmetry of Simple-Cell Receptive Fields in Macaque Primary Visual Cortex [Publisher: American Physiological Society]. *Journal of Neurophysiology*, *88*(1), 455–463. https://doi.org/10.1152/jn.2002.88.1.455

Rolls, E. T., & Tovee, M. J. (1995). Sparseness of the neuronal representation of stimuli in the primate temporal visual cortex. *Journal of Neurophysiology*, *73*(2), 713–726. https://doi.org/10.1152/jn.1995.73.2.713

Rozell, C. J., Johnson, D. H., Baraniuk, R. G., & Olshausen, B. A. (2008). Sparse Coding via Thresholding and Local Competition in Neural Circuits [Number: 10 Reporter: Neural Computation]. *Neural Computation*, *20*(10), 2526–2563. https://doi.org/10.1162/neco.2008.03-07-486

Savitzky, A., & Golay, M. J. E. (1964). Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry*, *36*(8), 1627–1639. https://doi.org/10.1021/ac60214a047

Scheich, H., Bullock, T. H., & Hamstra, R. H. (1973). Coding properties of two classes of afferent nerve fibers: High-frequency electroreceptors in the electric fish, Eigenmannia. [Publisher: American Physiological Society]. *Journal of Neurophysiology*, *36*(1), 39–60. https://doi.org/10.1152/jn.1973.36.1.39

Schiller, P. H., Finlay, B. L., & Volman, S. F. (1976). Quantitative studies of single-cell properties in monkey striate cortex. III. Spatial frequency [Publisher: American Physiological Society]. *Journal of Neurophysiology*, *39*(6), 1334–1351. https://doi.org/10.1152/jn.1976.39.6.1334

Schultz, P. F., Paiton, D. M., Lu, W., & Kenyon, G. T. (2014). Replicating kernels with a short stride allows sparse reconstructions with fewer independent kernels [Reporter: arXiv preprint arXiv:1406.4205]. *arXiv preprint arXiv:1406.4205*.

Shannon, C. E. (1948). A Mathematical Theory of Communication [Publisher: Institute of Electrical and Electronics Engineers (IEEE)]. *Bell System Tech-*

nical Journal, *27*(4), 623–656. https://doi.org/10.1002/j.1538-7305.1948.tb00917.x

Shapley, R., Hawken, M., & Ringach, D. L. (2003). Dynamics of orientation selectivity in the primary visual cortex and the importance of cortical inhibition. *Neuron*, *38*(5), 689–699. https://doi.org/10.1016/S0896-6273(03)00332-5

Simoncelli, E. P., & Olshausen, B. A. (2001). Natural Image Statistics and Neural Representation [Publisher: Annual Reviews]. *Annual Review of Neuroscience*, *24*(1), 1193–1216. https://doi.org/10.1146/annurev.neuro.24.1.1193

Tanabe, S., Haefner, R. M., & Cumming, B. G. (2011). Suppressive Mechanisms in Monkey V1 Help to Solve the Stereo Correspondence Problem. *Journal of Neuroscience*, *31*(22), 8295–8305. https://doi.org/10.1523/JNEUROSCI.5000-10.2011

Tanabe, S., & Cumming, B. G. (2014). Delayed suppression shapes disparity selective responses in monkey V1. *Journal of Neurophysiology*, *111*(9), 1759–1769. https://doi.org/10.1152/jn.00426.2013

Tang, P. T. P., Lin, T.-H., & Davies, M. (2017). Sparse coding by spiking neural networks: Convergence theory and computational results. *arXiv preprint arXiv:1705.05475*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Timofte, R., & Van Gool, L. (2015). Sparse Flow: Sparse Matching for Small to Large Displacement Optical Flow. *2015 IEEE Winter Conference on Applications of Computer Vision*, 1100–1106. https://doi.org/10.1109/wacv.2015.151

Tsao, D. Y., Conway, B. R., & Livingstone, M. S. (2003). Receptive fields of disparity-tuned simple cells in macaque V1. *Neuron*, *38*(1), 103–114. https://doi.org/10.1016/S0896-6273(03)00150-8

Watkins, Y., Thresher, A., Schultz, P. F., Wild, A., Sornborger, A., & Kenyon, G. T. (2019). Unsupervised Dictionary Learning via a Spiking Locally Competitive Algorithm [Reporter: Proceedings of the International Conference on Neuromorphic Systems - ICONS \textquotesingle19]. *Proceed-*

ings of the International Conference on Neuromorphic Systems - ICONS \textquotesingle19. https://doi.org/10.1145/3354265.3354276

Zeiler, M. D., Krishnan, D., Taylor, G. W., & Fergus, R. (2010). Deconvolutional networks. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/cvpr.2010.5539957

Zeldenrust, F., Wadman, W. J., & Englitz, B. (2018). Neural Coding With Bursts—Current State and Future Perspectives. *Frontiers in Computational Neuroscience*, *12*. https://doi.org/10.3389/fncom.2018.00048

Zhang, H. (2005). Exploring Conditions for the Optimality of Naïve Bayes. *International Journal of Pattern Recognition and Artificial Intelligence*, *19*(02), 183–198. https://doi.org/10.1142/S0218001405003983

Zylberberg, J., Murphy, J. T., & DeWeese, M. R. (2011). A Sparse Coding Model with Synaptically Local Plasticity and Spiking Neurons Can Account for the Diverse Shapes of V1 Simple Cell Receptive Fields (O. Sporns, Ed.) [Number: 10 Reporter: PLoS Computational Biology]. *PLoS Computational Biology*, *7*(10), e1002250. https://doi.org/10.1371/journal.pcbi.1002250

Chapter 3

Sparse coding predicts optic flow specificities of zebrafish pretectal neurons[1]

Abstract

Zebrafish pretectal neurons exhibit specificities for large-field optic flow patterns associated with rotatory or translatory body motion. We investigate the hypothesis that these specificities reflect the input statistics of natural optic flow. Realistic motion sequences were generated using computer graphics simulating self-motion in an underwater scene. Local retinal motion was estimated with a motion detector and encoded in four populations of directionally tuned retinal ganglion cells, represented as two signed input variables. This activity was then used as input into one of three learning networks: a sparse coding network (competitive learning), PCA whitening with subsequent sparse coding, and a backpropagation network (supervised learning). All simulations developed specificities for optic flow which are comparable to those found in a neurophysiological study (Kubo et al., 2014), but relative frequencies of the various neuronal responses were best modeled by the sparse coding approach without whitening. We conclude that the optic flow neurons in the zebrafish pretectum do reflect the optic flow statistics. The predicted vectorial receptive fields show typical optic flow fields but also "Gabor" and dipole-shaped patterns that likely reflect difference fields needed for reconstruction by linear superposition.

[1]Ecke, G. A., Bruijns, S. A., Hölscher, J., Mikulasch, F. A., Witschel, T., Arrenberg, A. B., & Mallot, H. A. (2019). Sparse coding predicts optic flow specificities of zebrafish pretectal neurons. *Neural Computing and Applications*, 32(11), 6745–6754. Available from: http://dx.doi.org/10.1007/s00521-019-04500-6

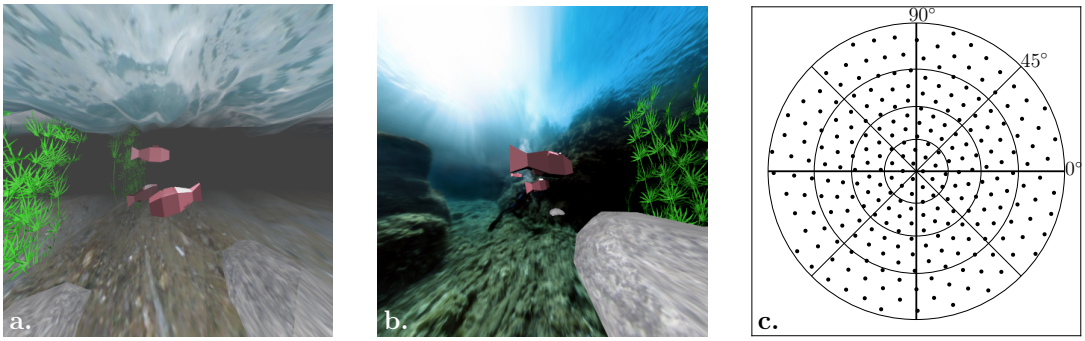


Figure 3.1. a. View of the virtual fish tank with muddy water (low viewing distance). Additional fish and plants will generate optic flow discontinuities. **b.** Example with high visibility. **c.** Mosaic of retinal ganglion cells, used to calculate the motion input. The figure shows 256 sampling points on a spherical cap modeling the fundus of the eye up to an eccentricity of $\Theta_{\max} = 80$ degrees. Approximately equidistant locations on the sphere were calculated according to Eq. 1. The spherical cap was flattened by stereographic projection and the circles of iso-eccentricity $\Theta = \{20, 40, 60, 80\}$ together with the retinal meridians (straight lines) are shown. The same projection was used for all kernels (receptive fields) shown in Fig. 2, 4, and 5.

3.1 Introduction

3.1.1 Optimality of visual receptive fields

In his *"neuron-doctrine for perceptual psychology"*, H. B. Barlow (1972) suggests that the *"nervous system is organized to achieve as complete a representation of the sensory stimulus as possible with the minimum number of active neurons"*. This idea also underlies a number of theoretical approaches to visual processing, such as independent component analysis, sparse coding of predictive coding; for an overview see Hyvärinen et al. (2009). While the general approach is widely accepted, specific predictions about the optimal processing scheme depend on the choice of the optimality criterion employed as well as on the information requirements of each species' life-style. Empirical tests of optimal coding theories of visual processing are therefore often limited to a qualitative level.

For the case of mammalian V1 cortex, B. A. Olshausen and Field (2005) summarized the evidence and concluded that for a full understanding of the system, simultaneous measurements of the activities of a large, unbiased set of neurons in response to natural stimuli would be required. Two-photon calcium imaging is a technology that allows to record activity from large populations of neurons. For

example, simultaneous monitoring of more than 100 cells from the mushroom body in *Drosophila* has provided evidence for sparse representation of odors (Honegger et al., 2011). Similarly, dense coding of odors found in the locust antennal lobe is transformed into a sparse code in the next processing stage, i.e. the mushroom body, by means of a wide field normalizing feedback (Papadopoulou et al., 2011).

We attempt an analysis of this type for the area pretectalis (APT) of the zebrafish, for which the response of thousands of neurons indeed was recorded while the fish was presented with optic flow stimuli (Kubo et al., 2014). Experimentally found response properties from a large, representative sample of neurons was compared to responses predicted from receptive fields of nodes in artificial neural networks. The networks were trained with optic flow patterns that were generated by simulating observer movement in a virtual fish tank. The receptive field predictions were based on three theoretical approaches, (*i*) sparse coding of optic flow patterns (unsupervised), (*ii*) PCA whitening with subsequent sparse coding (unsupervised), and (*iii*) backpropagation learning of ego-motion parameters from the same optic flow patterns (supervised).

3.1.2 Optic flow

Like many other animals, zebrafish larvae generate optokinetic responses of the eyes (OKR) and optomotor responses of the body (OMR) when exposed to visual stimuli simulating egomotion of the fish (Bak-Coleman et al., 2015; Kubo et al., 2014). Both eye- and body movements generate on the retina space-variant patterns of local motion vectors that have to be analyzed by subsequent processing stages. Neural algorithms suggested for optic flow analysis usually consist of at least two components: a local motion detector and a subsequent set of templates or motion models. These templates are used for identifying typical patterns relating to ego-motion maneuvers or encounters with obstacles and self-moving objects such as prey or predator (Franz et al., 2004; Perrone, 1992). Local motion detection can take place in the retina itself, as is generally the case in lower vertebrates, or in early areas of visual cortex. Higher brain areas analyzing optic flow patterns such as the focus of expansion, rotational vertices and left or right yaw rotations were identified in mammalian MST cortex (Orban, 2008) or in the zebrafish area pretectalis, APT (Kubo et al., 2014). Thus, neurons processing optic flow fields seem to represent typical, realizable flow patterns directly, rather than providing

components from which they might be reconstructed by linear combination. This is in line with the idea of sparse coding, where model neurons tend to respond to input patterns in their entirety. For the realizability of two-dimensional vector fields as optic flow see Verri et al. (1989).

Egomotion estimation from optic flow is possible with a large variety of established approaches derived from geometric considerations in the "inverse optics" approach (Marr, 1982; Raudies & Neumann, 2012). More recently, convolutional neural networks (CNNs) were shown to exhibit remarkable learning abilities to recover depth, motion fields, and camera motion simultaneously from image sequences in an unsupervised fashion (Vijayanarasimhan et al., 2017; Zhou et al., 2017). For the recovery of two-dimensional motion fields, algorithms based on deep learning (Dosovitskiy et al., 2015; Ilg et al., 2017) and template matching (Timofte & Van Gool, 2015; Wulff et al., 2012) were developed.

In our model, local visual motion was encoded in the direction-specific tuning curves of retinal ganglion cells. The motion signals themselves were calculated using Flownet 2.0 (Ilg et al., 2017) which uses the same encoding. Output from the retinal ganglion cells was then fed into a layer of simulated APT-neurons which developed optic flow analyzers.

3.1.3 Zebrafish visual system

Zebrafish retinal ganglion cells (RGCs), as well as pretectal cells, exhibit clear tuning to the direction and orientation of drifting gratings (Antinucci et al., 2016). Movement direction is not covered homogeneously, but clustered around three or four major visual field directions (Nikolaou et al., 2012). The larval zebrafish retina contains some 4000 ganglion cells with an average angular separation of about 2.5 degrees of visual angle.

RGCs project to APT, among other targets. The response characteristics of APT neurons were analyzed with visual stripe patterns (drifting gratings) moving either forward or backward and presented to the left, right, or both eyes (Kubo et al., 2014). Activity of *monocular* neurons depends only on the stimulus delivered to one eye and can therefore be considered to be directly driven from this eye's RGCs. In contrast, *binocular* neurons combine input from both eyes to generate specificities to forward or backward translation as well as to clockwise and counterclockwise rotation in the horizontal plane.

3.1.4 Aim of this study

With this study, we aim to establish a link between statistical learning theory, visual neuroscience, and visual ecology using the zebrafish as a model system: What are the informational needs of this species, how does computation proceed in its visual system, and how does this compare to optimal procedures from statistical learning theory? The goal is to make predictions about the detailed visual field organization of the fish and to identify mechanisms by which this organization can arise from adaptive and evolutionary processes.

3.2 Visual front end

Realistic optic flow stimuli were generated from a virtual reality simulation of observer motion in a fish tank, programmed in Blender (https://www.blender.org). The head of the fish was modeled by two cameras rigidly moving together with a rotation center somewhat behind the eyes. The field of view was 160 by 160 degrees with a binocular overlap of 45 degrees (see Kubo et al. (2014)). This resulted in central viewing directions of ± 57.5 degrees for the left and right eye.

The virtual fish tank contained objects at various distances from the observer as well as objects in mid-water (floating plants and other fish) generating optic flow discontinuities in translational egomotion (Fig. 3.1a,b). Note that translatory optic flow depends on object distance whereas rotatory optic flow does not. Visibility was set either low (muddy water, Fig. 3.1a) or high (clear water, Fig. 3.1b). Overall, the scenery was built to resemble the natural habitat of zebrafish as described in Spence et al. (2007).

Virtual fish were placed randomly in the environment and accelerated by a short, random impulse both for translation and rotation. Acceleration for all six degrees of freedom (DoF) were drawn independently from a uniform, zero mean distribution. For rotatory Dofs, we introduced an additional scaling factor in order to equalize the average flow vector lengths of rotatory and translatory flow components. After the acceleration impulse, the motion declined exponentially and a two-frame motion sequence was recorded from the later (slower) parts of this relaxation.

The fish retina was modeled as a spherical cap covering $2\Theta_{\max} = 160$ degrees in which 256 roughly equidistant sampling points were placed using a simple repel-

lence algorithm (Fig. 3.1c). For this, we first observed that the cap covers a fraction $(1-\cos\Theta)/2 = 41.3\,\%$ of the total sphere. We therefore placed $256/0.413 \approx 620$ points $\boldsymbol{r}_i$ randomly on the unit sphere. Repellence was realized as the iteration

$$\boldsymbol{r}_i^{t+1} = \left(\boldsymbol{r}_i^t + \lambda \sum_{j=1, j\neq i}^{256} \frac{\boldsymbol{r}_i^t - \boldsymbol{r}_j^t}{\|\boldsymbol{r}_i^t - \boldsymbol{r}_j^t\|^3} \right)^{\wedge}, \tag{3.1}$$

where λ is a small constant set to 0.05 and the $\wedge$-operator denotes normalization, i.e. projection to the unit sphere. The iteration was terminated when $\sum_i \|\boldsymbol{r}_i^{t+1} - \boldsymbol{r}_i^t\|$ dropped below 10^{-5}. Of these points, we used the 256 points closest to the pole of the sphere. The pole itself was chosen as the origin of the retinal coordinate system.

Planar camera images were warped by stereographic projection and sampled at these points. For each retinal sampling point i the corresponding local motion vector (u_i, v_i) was represented by two signed variables modeling the activity of pairs of RGCs tuned to opposite motion directions (right/left, and up/down).

3.3 Neural network modelling

3.3.1 LCA sparse coding

For *unsupervised learning*, we used the locally competitive algorithm (LCA) (B. A. Olshausen & Field, 1996; Rozell et al., 2008) which can be summarized as follows. Let $\boldsymbol{x} = \{x_n\}_{n=1}^N$ denote the input signal, i.e. the output of ganglion cells that encode local retinal motion. In sparse coding, the goal is to reconstruct $\boldsymbol{x}$ as a linear combination $\boldsymbol{x} \approx \sum_{k=1}^K a_k \boldsymbol{\varphi}_k$ with dictionary elements $\{\boldsymbol{\varphi}_k\}_{k=1}^K$, and activation coefficients $\{a_k\}_{k=1}^K$, for which sparsity is required (B. A. Olshausen & Field, 1996). The $\boldsymbol{\varphi}_k$ are vector fields from which the input vector field can be reconstructed as a linear combination. According to B. A. Olshausen and Field (1997) and Rozell et al. (2008), each $\boldsymbol{\varphi}_k$ can also be considered as the receptive field of the k-th output neuron, if a specific activation function with lateral feedback is assumed. In our application, the dictionary elements model the receptive fields of K APT neurons. The vector $\boldsymbol{a} = \{a_k\}$ contains the coefficients needed to reconstruct a given input pattern from the receptive fields. In our simulations, we require $a_k \geq 0$ at all times. If we write the $\boldsymbol{\varphi}_k$ as columns of a $N \times K$ matrix $\boldsymbol{\Phi}$ we obtain the

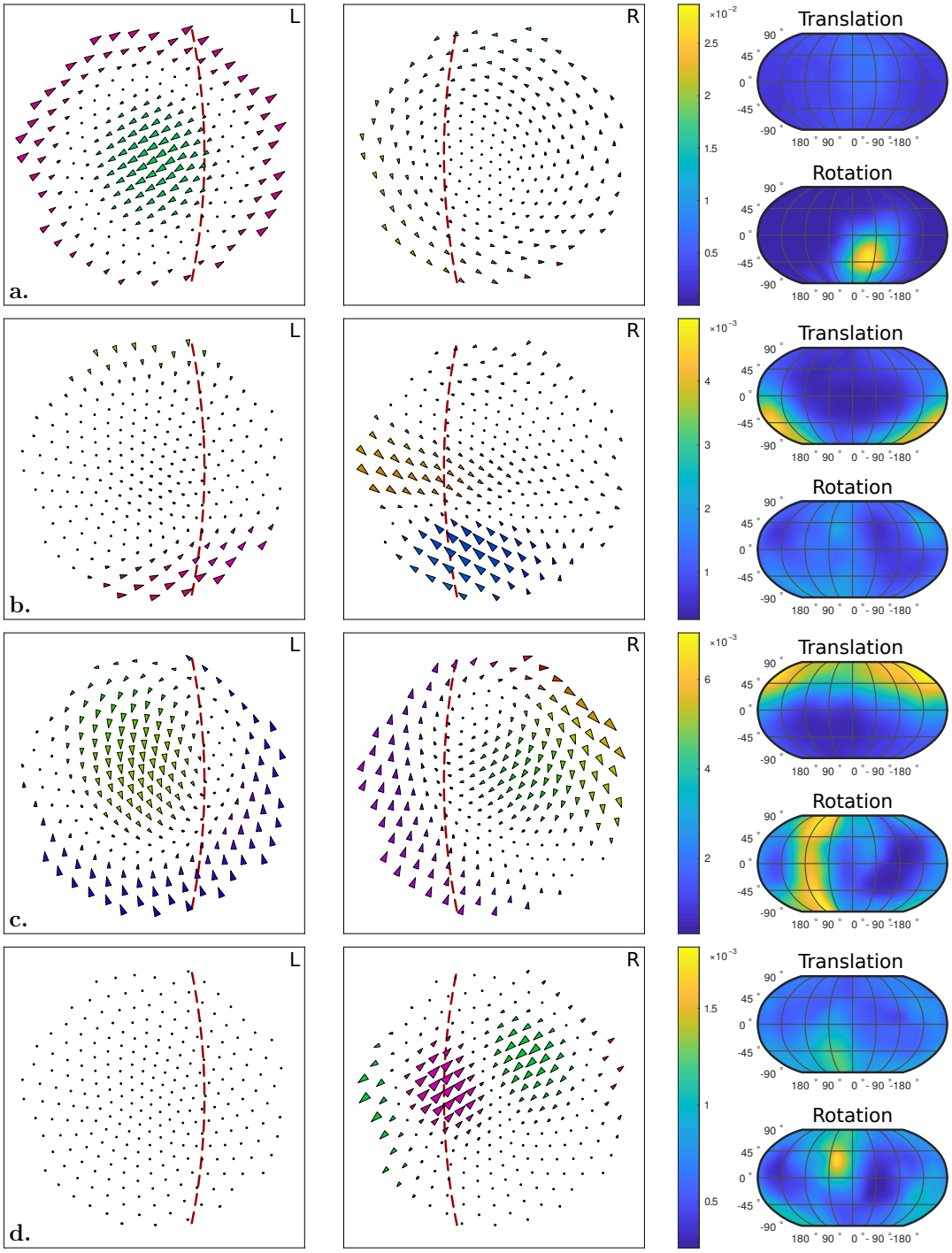


Figure 3.2. Sample binocular receptive fields from the sparse coding network. Each kernel is shown as a pair of vector fields. The red dotted lines mark the margin of binocular overlap. **a.** Rotation selective, **b.** translation selective kernel. **c.** Neuron selective for both, rotation and translation. **d.** Gabor-like kernel with poor selectivity.

error function $E(\boldsymbol{a}, \boldsymbol{\Phi}) = \frac{1}{2} \|\boldsymbol{x} - \boldsymbol{\Phi a}\|_2^2 + S(\boldsymbol{a})$, in which the first term penalizes reconstruction errors and $S(\boldsymbol{a})$ penalizes non-sparse vectors $\boldsymbol{a}$. While the original algorithm (B. A. Olshausen & Field, 1996) is based on the ℓ^1-norm, i.e. the total activity of $\boldsymbol{a}$, the locally competitive algorithm (LCA) seeks to minimize the ℓ^0-norm, i.e. the number of non-zero a-values or the number of active units (Rozell et al., 2008). Since $a_k \geq 0$, this amounts to choosing $S(\boldsymbol{a}) = \sum_{k=1}^{K} \lambda \, \mathcal{H}(a_k - \lambda)$ where $\lambda = 0.015$ is a threshold and $\mathcal{H}(x) = 0$ if $x < 0$ and $\mathcal{H}(x) = 1$ if $x \geq 0$.

For the optimization algorithm see B. A. Olshausen and Field (1996) and Rozell et al. (2008). The algorithm was run in Petavision (https://petavision.github.io, (Schultz et al., 2014)) with $K = 512$ APT-neurons and $77,076$ motion fields each sampled at 256 retinal points for each eye. Since each motion vector is encoded in two (signed) units, this results in $N = 1024$ input units. Examples of the resulting $\boldsymbol{\varphi}_k$ are displayed as vector fields in Fig. 3.2. I.e. for each retinal sampling point i, the components indexed $2i - 1$ and $2i$ are plotted as a vector at location i.

3.3.2 PCA whitening

In this approach we used a PCA of the input set and subsequent whitening as preprocessing. Let us denote the centered matrix of input data $\boldsymbol{X}$, their covaraiance matrix $\boldsymbol{C}$, the eigenvectors $\boldsymbol{U}$, and the diagonal matrix of PCA eigenvalues as $\boldsymbol{\Lambda}$; we then have $\frac{1}{N}\boldsymbol{X'X} = \boldsymbol{C} = \boldsymbol{U\Lambda U'}$. The PCA whitening is achieved by calculating $\boldsymbol{Y} = \boldsymbol{\Lambda}^{-\frac{1}{2}}\boldsymbol{U'X}$. As a result, $\boldsymbol{Y}$ is spherical with zero covariance.

The eigenvalues of the first 64 principal components of $\boldsymbol{X}$ appear in Fig. 3.3. Examples of components are shown in Fig. 3.4. The major part of the variance occurs in six principal components, corresponding to the six degrees of freedom of fish motion. Higher components might capture properties such as unreliable or missing optical flow due to the varying distribution of feature points in the visual field or variation of the translatory components of optic flow by the distance of the feature points from the observer (depth). Periodic or repetitive components of increasing spatial frequency were also reported by Wulff et al. (2012) for optic flow and by B. A. Olshausen and Field (1996) for static images. We therefore assume that they result from general statistic properties of image sequences rather than from egomotion specific origins.

In our simulation, we included the first 64 principal components, covering 99.83 % of the total variance. By whitening, the input variance of the first com-

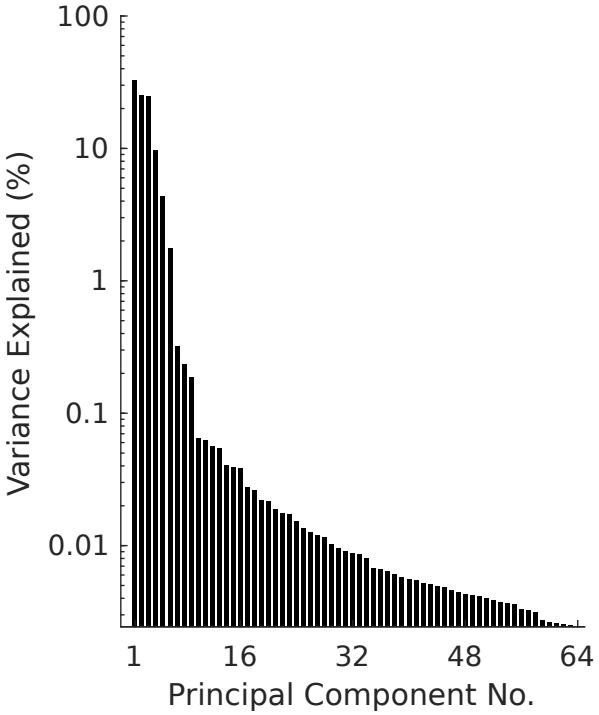


Figure 3.3. Variance explained of the first 64 PCA components calculated on the fish tank motion field database. For whitened sparse coding, the motion fields were projected on these components and subsequently rescaled to unit variance. Note the logarithmic scale of the ordinate that is needed to visualize the quickly diminishing share of higher components.

ponents that capture the prototypical 6 degrees of freedom on fish motion was decreased while variance from subsequent dimensions was increased. The sparse coding step was then applied to the whitened and dimensionality reduced variable $\boldsymbol{Y}$, as described above. We expect that whitening, as well as the redundancy reduction, aids the gradient decent on sparse and independent components (Hyvärinen & Oja, 2000).

3.3.3 Motion selectivity analysis

Tuning maps were calculated by probing each kernel with all motion fields from the fish-tank data base. Reusing the data base is unproblematic because learning was based on the reconstruction of the motion fields and not subject to motion selectivity. Egomotion was analyzed in just four degrees of freedom, i.e. the direction of translation as a unit vector T and the oriented axis of rotation as a unit vector R. Speeds are assumed to be non-negative, but are not differentiated otherwise. Thus, translation is always in the direction of T and rotations are counterclockwise about R. Therefore, clockwise and counterclockwise rotation about the same axis R are represented by the oriented axes R and $-R$, respectively. All motion

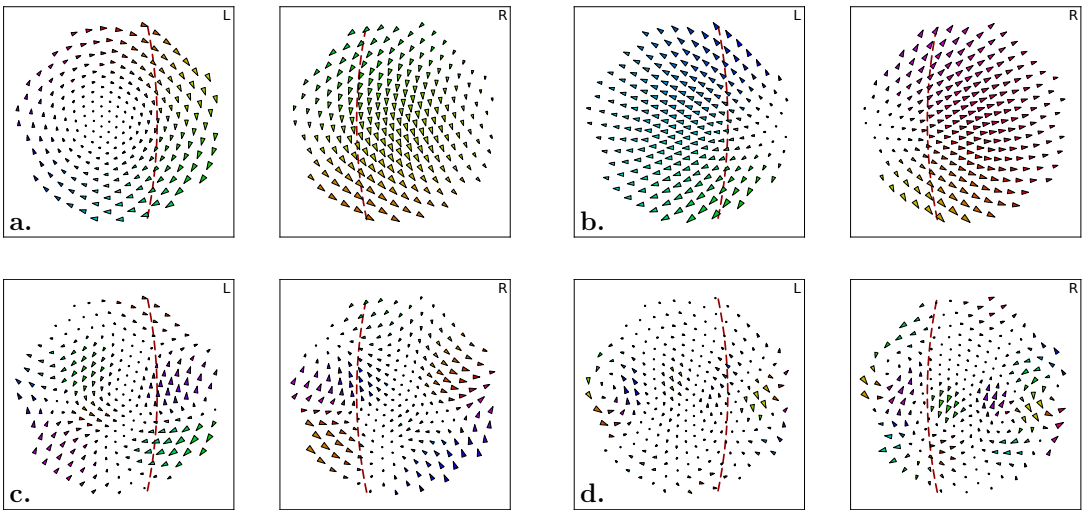


Figure 3.4. Example PCA components (PCs). **a.** PC 2: The first three components are rotation flowfields. **b.** PC 4: Components 3 to 6 represent translational characteristics. **c.** PC 25: Higher components (> 6) reflect structures of the environment such as depth variations and feature distribution. **d.** PC 64 is the last component included in the database transformation. Components with rank $\gtrsim 250$ do no more contain local structure. For a full list of PCs, see supplementary material.

fields used for probing were combinations of translations and rotations, i.e. linear superpositions of the respective pure translatory and rotatory fields. The starting pose of simulated movements was also randomized. Altogether, $77,076$ sets of four motion parameters (T_s, R_s), $s = 1, \ldots, 77,076$ were used for the calculation of the tuning curves.

For the calculation of the translation tuning maps, we defined 75 equally spaced standard directions T_i^* using the same repellence algorithm as before (Eq. 3.1). The stimuli were binned by the distance of their translation component T_s from the standard directions. Let S_i be the set of stimuli falling into the i-th bin; the average response value for a unit $\boldsymbol{\varphi}_k$ and direction of translation T_i^* is then given by

$$\tau_{T,k}(i) = \frac{1}{|S_i|} \sum_{s \in S_i} a_{k|s}, \tag{3.2}$$

where $a_{k|s}$ is the coefficient of kernel $\boldsymbol{\varphi}_k$ when representing stimulus s. An analogous procedure was used for the rotation maps $\tau_{R,k}(i)$. For display, the tuning maps were smoothed and transformed to a Robinson projection of the unit sphere.

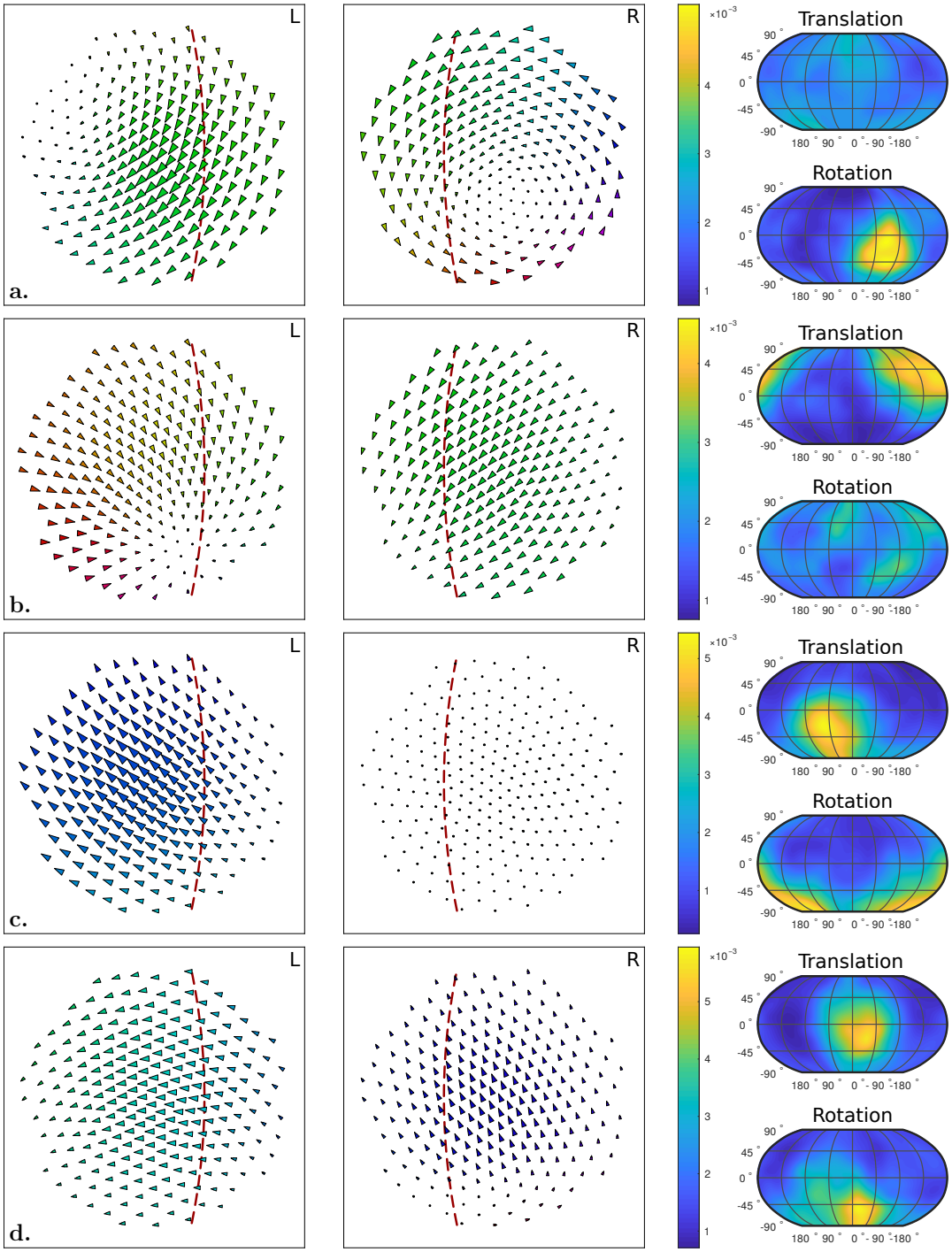


Figure 3.5. Sample binocular receptive fields from the whitened sparse coding network. **a.** Rotation selective and **b.** translation selective kernels. **c.** Monocular and **d.** binocular neuron selective for both, rotational and translational movement directions.

3.3.4 Backpropagation

For comparison, we also implemented a *supervised learning* version of the model that used the same retinal encoding scheme and input data described above. Motion sequences were labeled for egomotion by seven continuous variables, three for the unit-vector of heading (translation), three for the unit vector of the axis of rotation and a non-negative one for rotational speed. Note that translational speed cannot be recovered from optic flow, so we did not attempt to teach this to the network. The network contained three hidden layers with 1000, 600, and 200 units and an output layer with seven units with the above encoding. Implementation was carried out in TensorFlow (https://www.tensorflow.org/).

The network was able to recover the heading direction with a mean angular error of about 15 degrees and the axis of rotation with a mean angular error of about 19 degrees.

3.4 Results

The simulations produced two types of data, i.e. models of vectorial receptive fields, and neuronal responses to optic flow stimuli. We only discuss receptive fields for the two sparse coding networks since no obvious interpretation was found for the backpropagation case.

3.4.1 Kernels and tuning maps

Fig. 3.2 shows four typical examples out of the set of 512 $\boldsymbol{\varphi}_k$ fields for the sparse coding case without whitening. Kernels were ranked according to the average value of the corresponding coefficient over the complete input set.

Fig. 3.2a (rank 18 out of 512) shows a clear specificity for a counterclockwise rotation about a right, downward axis. This is also visible in the vector field for the right eye. In contrast, the left eye shows a center surround organization which might reflect motion parallax of a near object in front of a distant background; however, no clear translation specificity is found. Specificity for translation can be seen in Fig. 3.2b (rank 53). The vector fields do not show a well defined focus of expansion but show a roughly polar pattern. We also find combined specificities for rotation and translation (Fig. 3.2c, rank 52) which result from spiral patterns

in the vector fields. Fig. 3.2d (rank 86) shows a field with lower contribution to the reconstruction which is representative of a large number of fields. It is monocular with clearly delineated lobes of motion preferences in opposite directions, resembling Gabor functions for the horizontal and vertical motion components.

Overall, individual vector fields are often not realizable as optic flow fields in a rigid environment. This is in contrast to the findings in the whitened sparse coding approach where clearly realizable motion fields were obtained. Fig. 3.5 shows example fields for rotation (Fig. 3.5a), translation (Fig. 3.5b) and combined translation and rotation in a monocular and binocular case (Fig. 3.5c,d). The ranks for these kernels were 328, 416, 508, and 127, respectively. It is important to note, however, that kernel usage in reconstruction is much more homogeneous in the whitening case than in plain sparse coding such that the ranks are of minor relevance.

3.4.2 Comparison with physiological results

Binocular receptive fields obtained from either learning scheme were further analyzed by calculating their response to spherical rotating or translating grating stimuli as were used for receptive field mapping in the zebrafish study by (Kubo et al., 2014). Gratings moved either forward or backward and were presented either to the left, the right, or both eyes. Altogether, four monocular and four binocular stimulus types were to be distinguished, see Fig. 3.5. Each neuron or model neuron was classified for its reaction to each of the eight stimulus types, resulting in $2^8 = 256$ response types. Of these, 27 optic-flow-related cases are shown in Fig. 3.5, both for the zebrafish recordings (upper histogram) and for the three network simulations (lower histograms). There is also a substantial number of cells not classified into one of the illustrated 27 response types.

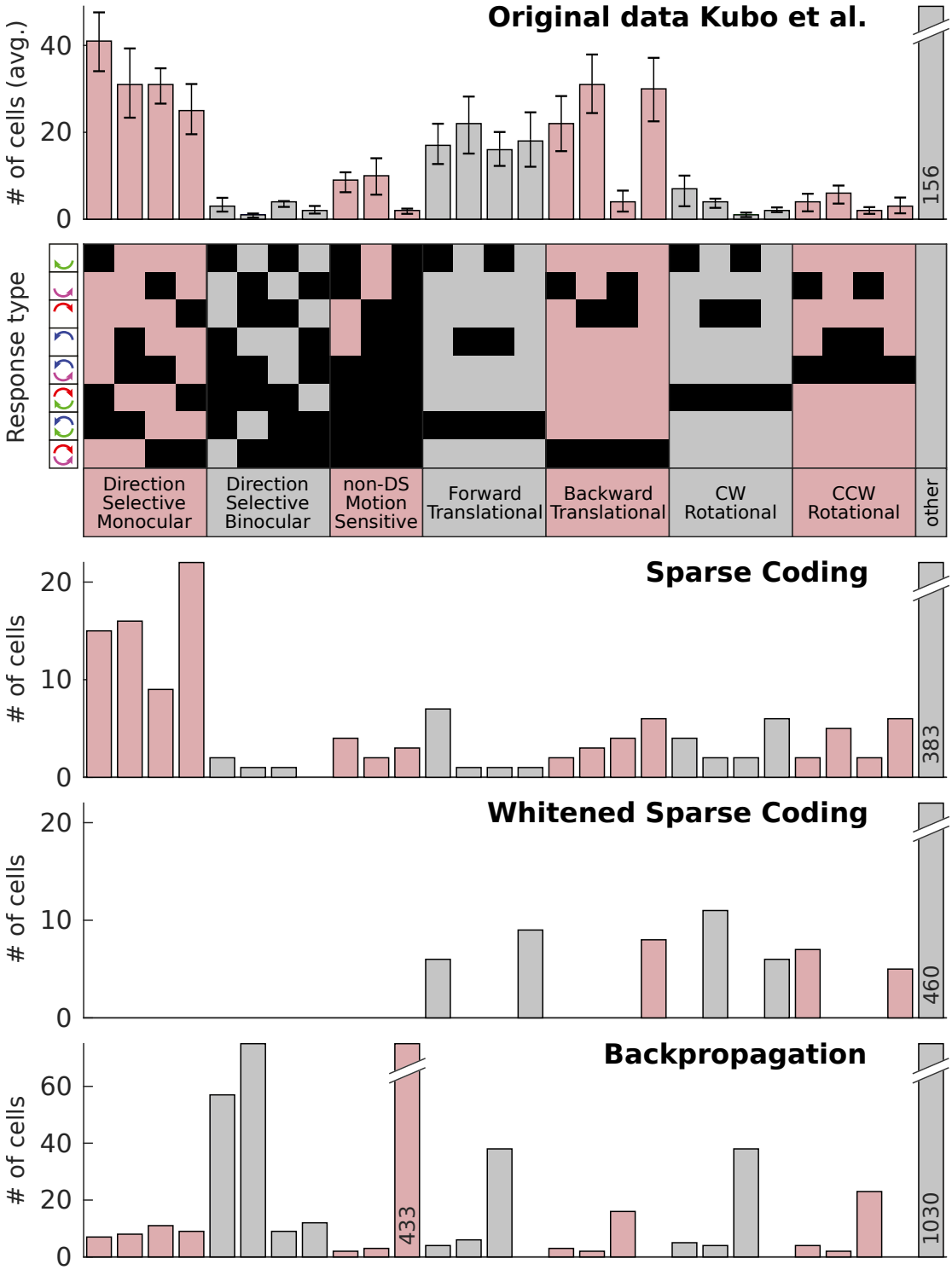
The response-type group "direction selective monocular" is most frequent in the fish as well as in the sparse coding network, but is missing in the whitened sparse coding network and underrepresented in the backpropagation network. It includes neurons that react to the stimulation of one eye, but ignore the stimulus of the other eye. On their own, such neurons cannot analyze egomotion because they cannot distinguish between forward translation and rotation to the contralateral side. However, in the reconstruction approach of sparse coding, they do seem to play an important role in describing the binocular motion fields as well.

The next most frequent response type groups comprise binocular neurons reacting to specific types of binocular optic flow such as translation or rotation. The specificity of these responses is established by integrating directional information across both eyes. Again, the sparse coding network seems to fit the data better than the other two approaches.

One conspicuous property of the whitened sparse coding network is its lack of kernels responding to non-egomotion related flow patterns. In a sense, this network seems to interpret all test patterns in terms of the egomotion it was trained with. This may be related to the fact that the kernels in the whitened sparse coding tend to reproduce characteristic pattens of egomotion.

Expected symmetries in the dataset are not generally found. For example, consider the response type "left and binocular forward" (10000010; first column in box "forward translational") and the response type "right and binocular forward" (00010010; second column in box "forward translational"). In the animal data, these response types are about equally frequent which is not reflected in the sparse coding networks. We do not think, however, that this is a reliable result of our simulation.

Figure 3.5 ***(next page).*** Summary of neuron response characteristics. The top two panels are redrawn from (Kubo et al., 2014). On the left of the "**Response type**" panel, the little arrows symbolize optic flow stimulation when the fish is heading towards the left, i.e. the first row shows forward optic flow stimulation to the left eye, the second row backwards stimulation to the left eye and so on. The response types are indicated by the columns of black squares. E.g., the first column refers to neurons responding whenever there is forward stimulation to the left eye, irrespective of the stimulus delivered to the other eye, and so on. The histogram on top ("**Original data**") shows the frequency per fish of neurons of a given response type found in a sample of 3015 cells from six zebrafish larva APT. Most neurons are monocular direction selective (first block). Also, a substantial fraction of neurons specifically responding to global optic flow fields (e.g., forward translation) was found. The third panel ("**Sparse Coding**") shows the results of the present study which are in good general agreement with the fish data, as opposed to "**Whitened Sparse Coding**" which yields neurons for higher level egomotion patterns. The "**Backpropagation**" block shows the responses of the 1,800 units from all three hidden layers of the supervised learning network, which had been trained to classify optic flow patterns for egomotion.


Original data Kubo et al.
# of cells (avg.)
0
20
40
156
Response type
Direction Selective Monocular
Direction Selective Binocular
non-DS Motion Sensitive
Forward Translational
Backward Translational
CW Rotational
CCW Rotational
other
Sparse Coding
# of cells
0
10
20
383
Whitened Sparse Coding
# of cells
0
10
20
460
Backpropagation
# of cells
0
20
40
60
433
1030

3.5 Conclusion

Our results allow three major conclusions. First, receptive fields of zebrafish APT neurons are clearly related to the statistics of environmental stimuli as extracted by the plain sparse coding network. The whitened sparse coding approach yields interesting results in terms of egomotion recovery but does not reflect the properties of zebrafish APT neurons. Still, it may model properties of higher level neurons in other animals or other areas of the brain.

Second, the statistical analyses of the optic flow stimuli reveal that the representation of the stimulus set requires vectorial receptive fields (kernels) that do not correspond to realizable flow fields such as simple foci or vertices. Examples are directionally opponent center-surround patterns (Fig. 3.2a) or spiral patterns as in Fig. 3.2c. This result is in conflict with template-based models of optic flow processing (Franz et al., 2004; Perrone, 1992) which predict realizable flow patterns as vectorial receptive fields.

The third conclusion is that the objective function of statistical learning approaches plays an important role in biological modeling. Kernels that are optimal for reconstructing retinal motion fields (as are generated by sparse coding) need not be the best for estimating egomotion. Indeed, the backpropagation approach in which egomotion was used as a teacher signal led to a response type pattern which is quite different from the animal data and the other simulations. The question of what exactly is a *complete* stimulus representation in the sense of H. B. Barlow (1972) needs to be re-considered in the light of the animal's lifestyle.

References

Antinucci, P., Suleyman, O., Monfries, C., & Hindges, R. (2016). Neural Mechanisms Generating Orientation Selectivity in the Retina. *Current Biology*, *26*(14), 1802–1815. https://doi.org/10.1016/j.cub.2016.05.035

Bak-Coleman, J., Smith, D., & Coombs, S. (2015). Going with, then against the flow: Evidence against the optomotor hypothesis of fish rheotaxis. *Animal Behaviour*, *107*, 7–17. https://doi.org/10.1016/j.anbehav.2015.06.007

Barlow, H. B. (1972). Single Units and Sensation: A Neuron Doctrine for Perceptual Psychology? [Number: 4 Reporter: Perception]. *Perception*, *1*(4), 371–394. https://doi.org/10.1068/p010371

Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Van Der Smagt, P., Cremers, D., & Brox, T. (2015). FlowNet: Learning Optical Flow with Convolutional Networks. *2015 IEEE International Conference on Computer Vision (ICCV)*, 2758–2766. https://doi.org/10.1109/iccv.2015.316

Franz, M. O., Chahl, J. S., & Krapp, H. G. (2004). Insect-inspired estimation of egomotion. *Neural Computation*, *16*(11), 2245–2260. https://doi.org/10.1162/0899766041941899

Honegger, K. S., Campbell, R. A. A., & Turner, G. C. (2011). Cellular-Resolution Population Imaging Reveals Robust Sparse Coding in the Drosophila Mushroom Body [Number: 33 Reporter: Journal of Neuroscience]. *Journal of Neuroscience*, *31*(33), 11772–11785. https://doi.org/10.1523/JNEUROSCI.1099-11.2011

Hyvärinen, A., Hurri, J., & Hoyer, P. O. (2009). *Natural Image Statistics*. Springer London. https://doi.org/10.1007/978-1-84882-491-1

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications [Number: 4 Reporter: Neural networks]. *Neural networks*, *13*(4), 411–430. https://doi.org/10.1016/S0893-6080(00)00026-5

Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2017). FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2017.179

Kubo, F., Hablitzel, B., Dal Maschio, M., Driever, W., Baier, H., & Arrenberg, A. B. (2014). Functional Architecture of an Optic Flow-Responsive Area that Drives Horizontal Eye Movements in Zebrafish. *Neuron*, *81*(6), 1344–1359. https://doi.org/10.1016/j.neuron.2014.02.043

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information*. W.H. Freeman.

Nikolaou, N., Lowe, A. S., Walker, A. S., Abbas, F., Hunter, P. R., Thompson, I. D., & Meyer, M. P. (2012). Parametric Functional Maps of Visual Inputs

to the Tectum. *Neuron*, *76*(2), 317–324. https://doi.org/10.1016/j.neuron.2012.08.040

Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images [Number: 6583 Reporter: Nature]. *Nature*, *381*(6583), 607–609. https://doi.org/10.1038/381607a0

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? [Number: 23 Reporter: Vision Research]. *Vision Research*, *37*(23), 3311–3325. https://doi.org/10.1016/s0042-6989(97)00169-7

Olshausen, B. A., & Field, D. J. (2005). How close are we to understanding V1? [Number: 8 Reporter: Neural computation]. *Neural computation*, *17*(8), 1665–1699. https://doi.org/10.1162/0899766054026639

Orban, G. A. (2008). Higher Order Visual Processing in Macaque Extrastriate Cortex. *Physiological Reviews*, *88*(1), 59–89. https://doi.org/10.1152/physrev.00008.2007

Papadopoulou, M., Cassenaer, S., Nowotny, T., & Laurent, G. (2011). Normalization for Sparse Encoding of Odors by a Wide-Field Interneuron [Number: 6030 Reporter: Science]. *Science*, *332*(6030), 721–725. https://doi.org/10.1126/science.1201835

Perrone, J. A. (1992). Model for the computation of self-motion in biological systems. *Journal of the Optical Society of America A*, *9*(2), 177. https://doi.org/10.1364/josaa.9.000177

Raudies, F., & Neumann, H. (2012). A review and evaluation of methods estimating ego-motion. *Computer Vision and Image Understanding*, *116*(5), 606–633. https://doi.org/10.1016/j.cviu.2011.04.004

Rozell, C. J., Johnson, D. H., Baraniuk, R. G., & Olshausen, B. A. (2008). Sparse Coding via Thresholding and Local Competition in Neural Circuits [Number: 10 Reporter: Neural Computation]. *Neural Computation*, *20*(10), 2526–2563. https://doi.org/10.1162/neco.2008.03-07-486

Schultz, P. F., Paiton, D. M., Lu, W., & Kenyon, G. T. (2014). Replicating kernels with a short stride allows sparse reconstructions with fewer independent kernels [Reporter: arXiv preprint arXiv:1406.4205]. *arXiv preprint arXiv:1406.4205*.

Spence, R., Gerlach, G., Lawrence, C., & Smith, C. (2007). The behaviour and ecology of the zebrafish, Danio rerio. *Biological Reviews*, *83*(1), 13–34. https://doi.org/10.1111/j.1469-185X.2007.00030.x

Timofte, R., & Van Gool, L. (2015). Sparse Flow: Sparse Matching for Small to Large Displacement Optical Flow. *2015 IEEE Winter Conference on Applications of Computer Vision*, 1100–1106. https://doi.org/10.1109/wacv.2015.151

Verri, A., Girosi, F., & Torre, V. (1989). Mathematical properties of the two-dimensional motion field: From singular points to motion parameters. *JOSA A*, *6*(5), 698–712. https://doi.org/10.1364/JOSAA.6.000698

Vijayanarasimhan, S., Ricco, S., Schmid, C., Sukthankar, R., & Fragkiadaki, K. (2017). Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*.

Wulff, J., Butler, D. J., Stanley, G. B., & Black, M. J. (2012). Lessons and insights from creating a synthetic optical flow benchmark. *Computer Vision–ECCV 2012. Workshops and Demonstrations*, 168–177. https://doi.org/10.1007/978-3-642-33868-7_17

Zhou, T., Brown, M., Snavely, N., & Lowe, D. G. (2017). Unsupervised Learning of Depth and Ego-Motion from Video. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2017.700

Chapter 4

Dual population coding for topological navigation: Combining discrete state-action-graphs with distributed spatial knowledge[1]

Abstract

Topological schemes for navigation from visual snapshots have been based on graphs of panoramic images and action links allowing the transition from one snapshot point to the next; see, for example, Cartwright and Collett (1987a) or Franz et al. (1998a). These algorithms can only work if at each step a unique snapshot is recognized to which a motion decision is associated. Here, we present a population coding approach in which place is encoded by a population of recognized "micro-snapshots" (i.e. features), each with an associated action. Robot motion is then computed by a voting scheme over all activated associations. The algorithm was tested in a large virtual environment (Virtual Tübingen (van Veen et al., 1998)) and shows biologically plausible navigational abilities.

[1]Mallot, H. A., Ecke, G. A., & Baumann, T. (2020). Dual Population Coding for Path Planning in Graphs with Overlapping Place Representations. *Spatial Cognition XII*, 3–17. Available from: http://dx.doi.org/10.1007/978-3-030-57983-8_1

4.1 Introduction

4.1.1 An evolutionary view of spatial representation

The evolution of spatial cognition is generally thought to have started from simple stimulus-response behaviors such as stimulus-driven orienting reactions and to proceed further by a number of innovations that include (i) mechanisms for egomotion perception and path integration, (ii) the memorization of stimulus-response (or state-action) pairs composed of a distinguishable landmark and a navigational action ("recognition-triggered response"), (iii) the concatenation of such recognition-triggered responses into chains or routes, and (iv) the linking-up of multiple recognition-triggered responses into networks or graphs in which novel routes can be inferred by the combination of known route segments (Madl et al., 2015; H. A. Mallot & Basten, 2009; Trullier et al., 1997; J. M. Wiener et al., 2011). In addition, mechanisms for invariant landmark and place recognition, strategic selection of way- or anchor-points, metric embedding of place-graphs, or hierarchical graph structures may improve navigational performance and are thus likely to play a role.

Since many different models can be build on these elements, it is interesting to ask for a minimal or most parsimonious model supporting a given level of behavioral flexibility. In this paper, we address this question for the case of the minimal cognitive architecture supporting way-finding behavior. By a minimal model, we mean a model meeting the following requirements:

1. A minimal model should be close to the evolutionary starting point of stimulus-response, or state-action schemata;

2. it should require only a small amount of visual invariance in object recognition and therefore work with the rawest possible image information;

3. it should use simple decision processes in path-planning such as recognition-triggered response; and

4. it should not rely on explicit metric information which is hard to obtain.

With these constraints in mind, we present a model for graph-based navigation that marks a lower bound of cognitive complexity required for way-finding and that can be used to study further improvements by additional evolutionary innovations.

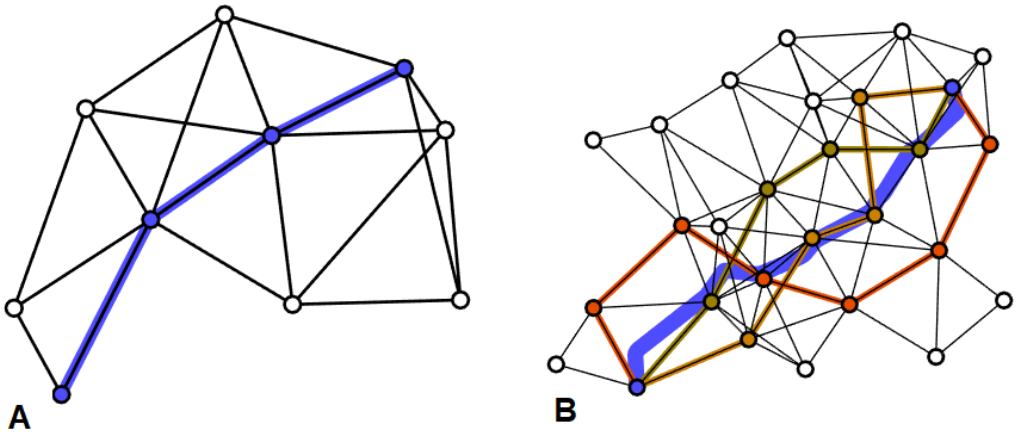


Figure 4.1. Graph-navigation in single-unit and population coding. A. In standard topological navigation, every place is represented by a unique node and route segments are given by the graph-links. The desired trajectory shown in blue is therefore a graph path. B. In dual population coding, a bundle of paths is constructed for a given navigation problem. Three such paths without common nodes (except start and goal) are shown in the figure. T he desired trajectory is then calculated by a voting scheme over the currently visible nodes of all paths. It is generally not a path of the graph.

4.1.2 Dual population coding

Two basic elements of spatial representations are (i) state-action associations such as Tolman's (Tolman, 1932) means-ends-relations, Keefe & Nadel's O'Keefe and Nadel (1978) taxon system, Kuipers' (Kuipers, 1978) control laws, or the place recognition-triggered response of Trullier et al. (Trullier et al., 1997), and (ii) the representations of places and place relations such as Cartwright & Collett's (Cartwright & Collett, 1982) snapshot-codes for places or O'Keefe & Nadel's (O'Keefe & Nadel, 1978) locale system. The two systems are connected by the role of place recognition as activator of a state in the state-action schemata involved.

In most models, it is assumed that each place is represented by just one node of a graph (Franz et al., 1998a; Kuipers, 1978, 2000; R. U. Muller et al., 1996) such that a unique state-action schema will control each navigational step. If place recognition fails, navigation will go wrong. Robustness of navigation therefore depends foremost on the robustness and invariance of place recognition as a prerequisite. Here, we argue that in an evolutionary view of navigation, robust way-finding should be possible even with rudimentary place recognition and

distributed place representations.

Our model differs from standard models of topological navigation (Franz et al., 1998a; Kuipers, 1978, 2000; R. U. Muller et al., 1996) in two major respects that can be summarized as "dual population coding": first, at any instant in time, many nodes of the graph are activated and encode the agent's position in a population scheme. This avoids costly selection processes of strategic anchor points and has the additional advantage that the visual cues and recognition processes can be kept simple. Of course, population coding of space is well in line with empirical findings in the place-cell literature (Wilson & McNaughton, 1993). Second, as a consequence of population coding of space, route selection has to be based on many interacting recognition-triggered response schemata, one for each active unit in the population code. This is implemented by a voting scheme where the suggested motion decisions from all active schemata are averaged. The idea of view voting has been suggested earlier for behavioral data by H. A. Mallot and Gillner (2000a). In insect navigation, a similar scheme has been suggested for route following with multiple snapshots by B. Baddeley et al. (2012) and Smith et al. (2007), but unlike our model, this model does not allow for alternative route decisions from a given position. As a result of dual population coding, the trajectory eventually found by the algorithm is not a path of the graph, but a metric average of bundles of many paths connecting individual nodes in the population codes for start and goal.

4.1.3 Elements of topological navigation

Graph nodes. Graph models of spatial representations have used various types of nodes. As mentioned above, the standard models (Cartwright & Collett, 1987a; Franz et al., 1998a; Kuipers, 1978, 2000; R. U. Muller et al., 1996) use place-graphs where nodes represent places defined as visual areas from which the corresponding landmark cues are recognized. The places are not geometrical points but have an extension, also called a "confusion area" (Franz et al., 1998a; Hübner & Mallot, 2007a), within which the cue is always recognized and further distinctions of location are therefore impossible. This idea of place representation is well in line with the snapshot mechanism in insect vision, where raw panoramic images are used as place cues (Cartwright & Collett, 1982; Fleer & Möller, 2017; Franz et al., 1998b; Möller & Vardy, 2006). Note, however, that more complete models of place recognition will also include information about three-dimensional layout

and place neighborhoods (Epstein & Kanwisher, 1998; Lowry et al., 2016; H. A. Mallot & Lancier, 2018; Wilson & McNaughton, 1993). A more general view on the content represented by the graph nodes has been suggested by Tolman (1932) in his means-ends-network. Here the "nodes" are inner states of the agent or intermediate goals (the "ends") that can be achieved. Of course, such goals could be places to reach, but Tolman's view of a cognitive map can easily be extended to non-spatial problem solving as in Arbib & Lieblich's world graph approach (Arbib & Lieblich, 1977). Another idea of spatial graphs assumes that nodes do not represent place or observer position, but are also specific for viewing direction. The result is a graph of views, each specific for an observer pose (position plus orientation) (Gaussier et al., 2002; Röhrich et al., 2014; Schölkopf & Mallot, 1995). This idea is taken to an extreme in the current approach: graph nodes are now specific for small image features or "micro-snapshots" that are visible from within certain "place fields", but these place fields are vague, largely overlapping and subject to aliasing. In the implementation discussed here, micro-snapshots are realized as SURF-features (Bay et al., 2008a). For features that can be recognized with a higher level of invariance, see for example the "bag-of-words" algorithm for object- and scene recognition (Sivic & Zisserman, 2003a).

Population code for space. Neural models of place recognition are generally based on distributed representations such as the ensemble code for space found in the hippocampal place cell system (O'Keefe & Dostrovsky, 1971; Wilson & McNaughton, 1993). Places are then represented by activity peaks ("attractors") on a layer of places cells (Bicanski & Burgess, 2018; Byrne et al., 2007; McNaughton et al., 2006; Sheynikhovich et al., 2009) in which many place cells are simultaneously active. The attractor model assumes that cell connectivity reflects spatial nearness of each cell's place field, although the hippocampus does not show a topographic organization. Connectivity is local (when plotted in the coordinates of the firing fields) and this locality is required for the formation of a stable attractor (Amari, 1977). Attractor models nicely explain egocentric working memory processes such as spatial updating, approaches to a goal or mental imagery, but are not easily generalized to large-scale representations in longterm memory and to way-finding in scarcely connected graphs.

Arbitration. In their distinction between route and map memories, O'Keefe and Nadel (1978) emphasize the fact that in a route, the last element is the goal, whereas in a map, every node can become the goal. This implies that a path-planning mechanism must exist that allows to generate a route from the current starting point to an arbitrary goal. In simple graph models, the required computation is the solution of a shortest-path problem such as the well-known Dijkstra algorithm (Dijkstra, 1959). Shortest-path algorithms are commonly used in models of topological navigation (Kuipers, 2000; R. U. Muller et al., 1996; Schölkopf & Mallot, 1995; J. M. Wiener & Mallot, 2003). The resulting route is a path in the graph, i.e. a sequence of graph nodes connected by graph links that contain information of the required action for each step. Shortest path algorithms require a full planning cycle before the onset of navigation, which is probably not the case in animals and humans where planning depth is limited to a few steps (Cramer & Gallistel, 1997; Huys et al., 2015; J. M. Wiener et al., 2009). Complete start-to-goal planning with a small number of steps can however be achieved in hierarchical representations (Botvinick, 2008; J. M. Wiener & Mallot, 2003).

If places are represented by a population of graph nodes, each associated with a suggested action, a mechanism for decision-making is required. In our approach, all possible actions can be expressed as angular heading changes, which can simply be averaged in a voting scheme. If possible choices cannot be superimposed or averaged, non-linear interaction schemes are needed as have been studied e.g. by Arbib and Lieblich (1977).

Metric information. Topological navigation is possible without the use of any metric information whatsoever. This would be the case if action links are specified as guidances, i.e. by the next landmark cue to reach, and each node is within the catchment area of all its connected neighbors (Cartwright & Collett, 1987a; Franz et al., 1998a). This approach requires large catchment areas and therefore elaborate place recognition, which is not available in the micro-snapshot approach. Rather, actions need to be represented by associating a movement direction to each micro-snapshot. These movements are carried out "ballistically", i.e. without further control.

Directional action information could be stored in a purely egocentric way, i.e. relative to the current bearing of each micro-snapshot (feature). Initial experi-

ments with this encoding, however, failed, probably because each feature can be detected under a wide range of viewing directions but should always code for the same action. For example, if the agent passes a feature in a narrow alley and the associated motion direction is straight ahead, the angle between the feature bearing and the required movement direction could vary almost between 0 and 180 degrees.

In this study, we therefore decided to store an allocentric reference direction and express both heading and action angles relative to this reference. Compared to true north, this allocentric reference direction is subject to errors. However, it will be shown to be stable over repeated visits of a given position and to drift only continuously between different positions in the maze. Estimates of an allocentric reference direction seem to be computational requirement in path integration as well (Cheung & Vickerstaff, 2010; Vickerstaff & Cheung, 2010). In biological systems, the reference direction is represented by the head direction system (Seelig & Jayaraman, 2015; Taube et al., 1990; Taube, 2007).

In the present approach, odometry is used only implicitly by keeping a fixed distance of travel between subsequent feature recordings. Graph links therefore contain local metric information, i.e. the fixed distance and the required turn as has been used also in the graph approaches by Foo et al. (2005), Schölkopf and Mallot (1995), and Warren (2019). Global metric embedding of the graph should be possible, but is not attempted in this study.

The model was implemented, tested, and evaluated in in two virtual environments, one modeled after the downtown area of Tübingen, Germany (van Veen et al., 1998), which the algorithm learned by random walk exploration, and an iterated Y-maze called *Hexatown*, inspired by a study by H. A. Mallot and Gillner (2000a).

The algorithm was able to guide an agent to any known location in the virtual environments; its ability to find good routes for navigation was confirmed experimentally by comparing the algorithm's routes to optimal length, with the result being routes that were only 10 % - 20 % longer than the optimum on average. Evaluations were performed on a customary home computer, achieving good real-time performance.

Table 4.1. List of variables.

variable	description
f_i	feature stored in graph
$\boldsymbol{d}_i$	descriptor (64-dimensional description vector of feature f_i)
ϑ_S	similarity threshold for feature recognition
F, F_t	set of all features f_i in the graph and subset visible at time t
$N(i)$	neighborhood set of all features co-visible with f_i.
ϑ_N	neighborhood threshold for feature recognition
ψ_i	place field (region of visibility) of feature f_i
t	time step (frame) in multiples of 1/15th of a second
l	learning step occuring every 15 frames during exploration
a_{ij}	directional graph edge leading from feature f_i to f_j.
$\nu, \boldsymbol{\nu}$	reference direction as angle or unit-vector
β_i	current perceived bearing of feature f_i relative to ν.
$\hat{\beta}_i$	stored bearing label of feature f_i
$\eta_t, \boldsymbol{\eta}_t$	heading angle of the agent at time step t, relative to ν, expressed as angle or unit vector
$\alpha_{ij}, \boldsymbol{\alpha_{ij}}$	stored heading label for a_{ij} as angle or unit vector (interpreted relative to ν)
λ, κ	weights for updating of ν and η
$\boldsymbol{n}$	random error drawn from normal distribution, for ν updating.
c	counter used for calculating iterative means
p, P	Dijkstra path and path bundle (set of multiple non-overlapping Dijkstra paths)
E_p, F_p	edge and node sets of a Dijkstra path p.
J_t	set of currently visible features contained in paths of bundle P.
$\overline{\boldsymbol{\alpha}}_t$	movement consensus at time step t, obtained from features in J_t.

4.2 Navigation algorithm

4.2.1 Feature detection

Micro-snapshots are defined by as "upright speeded-up robust features" (U-SURF) as implemented in the OpenCV computer vision library (Bay et al., 2008a; Bradski, 2000). SURF finds interest points as intensity blobs by searching local maxima of the determinant of the image Hessian; color information is ignored. Scale invariance is achieved by considering each feature point at its optimal scale. In a second step, a 64-dimensional vector ("descriptor") is associated with each blob, containing information about image intensity gradients in a small patch around

the interest point. The descriptor is used to compare and match features with each other. In U-SURF, it is assigned a unique orientation and is therefore not rotation invariant. Rotation invariance is not required in our algorithm since the agent is confined to movements in the plane. The number of scale levels was limited to two octaves with two layers each since information about the viewing distance of a feature should not be completely ignored.

The features of a frame were ranked according to the value of the determinant of the local Hessian, i.e. their contrast. Up to 30 features from each frame were used for further analysis. We denote the features as f_i and their descriptors as $\boldsymbol{d}_i$; and $F = \{f_i \,|\, i = 1, \ldots, n\}$ is the set of all features stored in the system.

4.2.2 Feature matching

Whenever a feature is detected by the U-SURF procedure, it is checked for identity with all stored features in F using two criteria. First, the root mean squared difference between the descriptors of the compared features should be below a threshold ϑ_S. Second, to avoid aliasing in large sets of features, we require that the features share a context of at least ϑ_N other features. To this end, we store for each feature f_i the set N_i of simultaneously visible other features. Two features f_i, f_j are thus identified with each other, if $\|\boldsymbol{d}_i - \boldsymbol{d}_j\|^2 < \vartheta_S$ and $|N_i \cap N_j| \geq \vartheta_N$. If an encountered feature is found to be novel, it is included into F.

The value for ϑ_N depends on the total number of features detected in each image. In our simulations, the value was set to $\vartheta_N = 4$ at up to 30 different features per frame. Note that aliasing still occurred occasionally even with expanded feature-neighbor matching (see Fig 4.4 below). In practice, the algorithm is robust against a small amount of outliers and can find and navigate routes even with faulty map data. See section "Pathfinding and voting" below for more details.

Fig. 4.2 shows two features in the respective images from the "Virtual Tübingen" dataset. In the second row, the position from which each feature was first defined and added to F is marked by a cross. For all positions in open space, color indicates the similarity of the most similar visible feature with the stored one. The third row of Fig. 4.2 shows the area from which the feature is detected using the two-step comparison procedure with similarity of descriptors and feature context. It will be called the place field of the feature and roughly corresponds to the catchment areas in snapshot homing or the firing fields of a neuron tuned to

the feature.

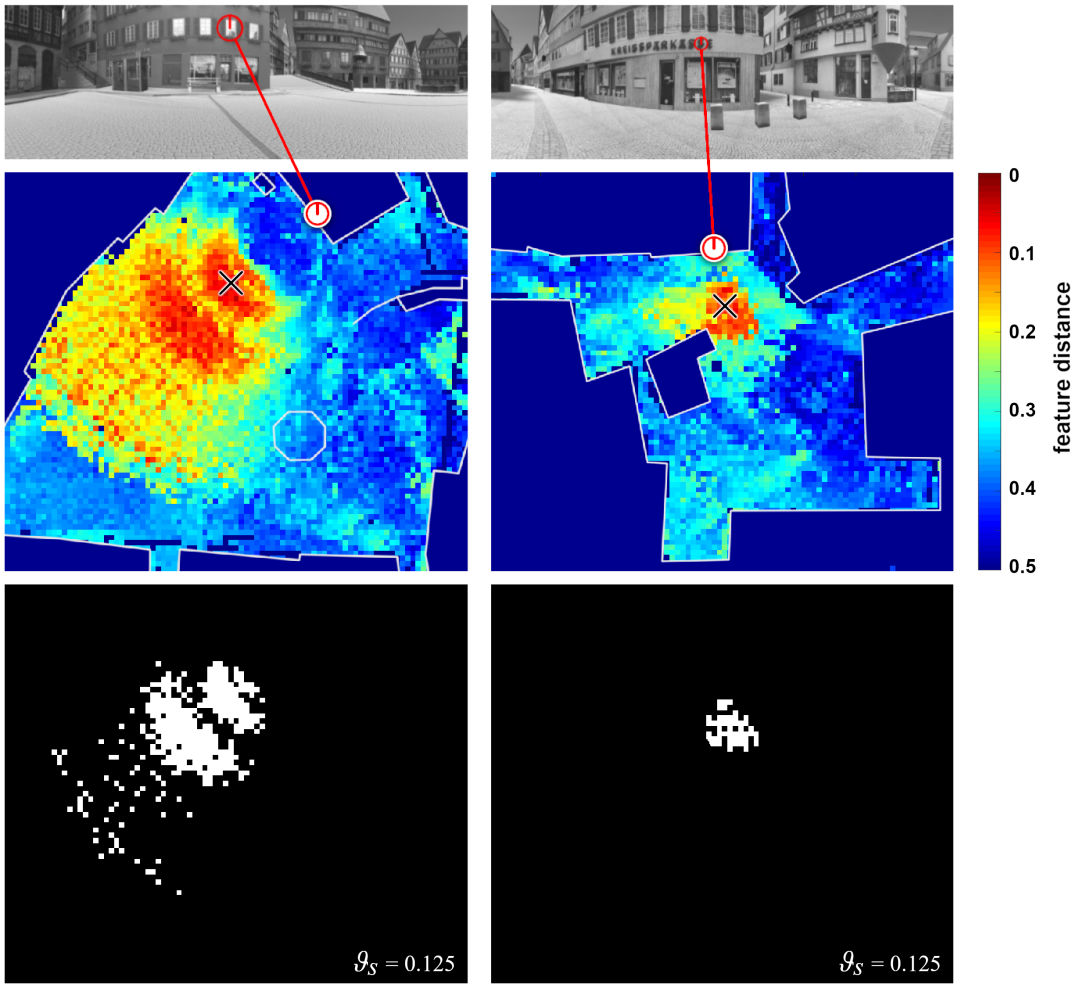


Figure 4.2. Place fields. Top row: two views from a scene with detected features (a window and a letter from a company nameplate). Middle row: local maps of the environment superimposed with the "feature distance" of the reference feature to the most similar feature detected from each position. (Feature distance is $\min_k \|\boldsymbol{d}_{ref} - \boldsymbol{d}_k\|^2$ where k numbers all features visible from each location.) The black $\times$ marks the position where the reference feature was first detected. Third row: map of locations where one feature was identified with the reference feature, based on both the similarity criterion and the consensus criterion. Note that the set of locations is not connected. Also in the larger open space (left column: Market place), the place fields tend to be larger than in smaller places (right column: Street crossing "Krumme Brücke").

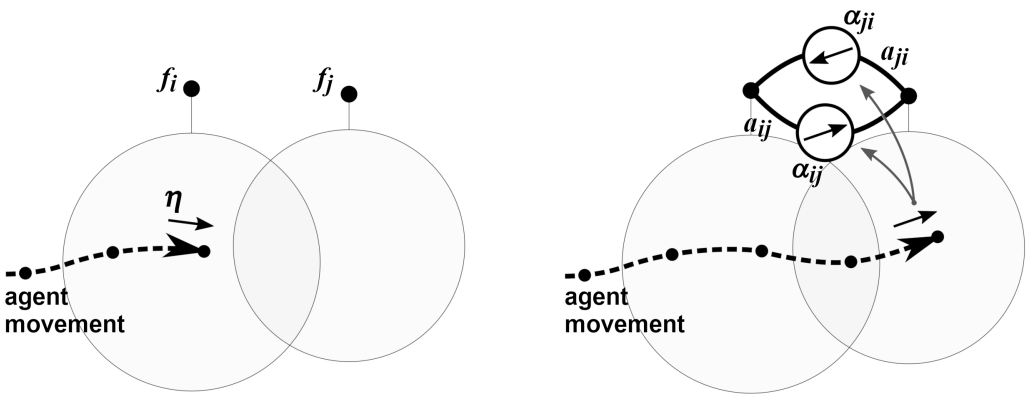


Figure 4.3. Graph edge learning. *Left:* Dotted line is the trajectory of agent with time steps (small dots) and learning steps (bold dots). Two features $\boldsymbol{f}_i$ and $\boldsymbol{f_2}$ have already been encountered and added to the feature set; the circles indicate their place-fields. The agent is currently moving with heading $\boldsymbol{\eta}$ in the place-field of feature $\boldsymbol{f}_1$, but outside of the place-field of feature $\boldsymbol{f}_2$. *Right:* The agent has now passed the overlap zone where both features are detected. At the next learning step, feature $\boldsymbol{f}_2$ is again detected but feature $\boldsymbol{f}_1$ has moved out of sight. In this situation, a bidirectional pair of edges a_{ij}, a_{ji} is added to the graph and the edges are labeled with the current heading and its inverse, $\pm\boldsymbol{\eta}$.

4.2.3 Graph edge formation

If a feature that was previously visible gets out of sight, the agent must have traveled a path out of the place field of this feature to some point inside the place fields of other features that remain or have become visible. This is the basic idea of learning graph edges in the algorithm. In order to avoid too high densities of graph links, new edges can be stored not at every time step, but only at a slower pace.

The basic time step of the algorithm is the frame, i.e. the recording of one image; we denote frames by the index t. The frame rate used in the graphics simulations below is 15 frames per second. Graph learning does not occur at every time step, but only once in a while, when the agent has moved sufficiently far away from the last learning event. Learning steps are counted by a second counter l. In the simulations below, the distance that the agent must have traveled before a new learning step occurs was set to about two simulated meters, or 15 frames. Note that we use the position ground truth of the VR simulation for stepping l. This can easily be relaxed by some simple path integration algorithm which was,

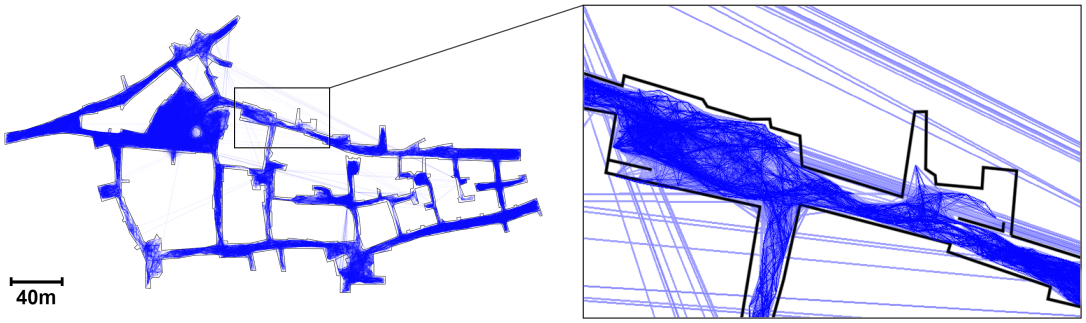


Figure 4.4. Full graph of the testing environment "Virtual Tübingen". For visualization, the view graph is embedded into a map by placing each feature at the agent's position from where it was first detected, and drawing the edges between them (blue lines). The shown graph completely maps the virtual environment and consists of 222,433 nodes and 3,492,096 edges. The blue lines crossing empty white space result from aliasing.

however, not implemented. The time steps at with learning counter l is stepped, are denoted by $t(l)$.

Let F_1 and F_2 be the sets of features visible at two subsequent learning steps l and $l+1$, respectively. Assume $f_j \in F_1$ and $f_j \notin F_2$. We then add a pair of directed edges a_{jk}, a_{kj} (forward and backward) between f_j and up to three randomly chosen features $f_k \in F_2$ to the graph. The edges are labeled with the current heading or its inverse, respectively (see Fig 4.3 and next paragraph). The number of edges created per vanishing feature was limited to three to avoid exceedingly high computation costs in later graph search. For the same reason, the upper limit of a node's degree after repeated visits is set to 100. A depiction of the full graph appears in Fig 4.4.

4.2.4 Edge labeling and reference direction

During the entire travel, the agent is estimating and maintaining an allocentric reference direction ν which is initialized to the value $\nu = 0$ at frame 1 (see Fig 4.5). All other angles are expressed relative to this reference direction, i.e. in an allocentric scheme similar to the head direction in allocentric path integration (Cheung & Vickerstaff, 2010; Taube, 2007; Vickerstaff & Cheung, 2010). The dependent angles are (i) the current heading angle η_t, (ii) the feature bearings $\hat{\beta}_i$ stored with each feature f_i upon definition of the feature, and (iii) the directional labels of the edges a_{ij}, a_{ji} which are initialized with or against the current heading angle, i.e.

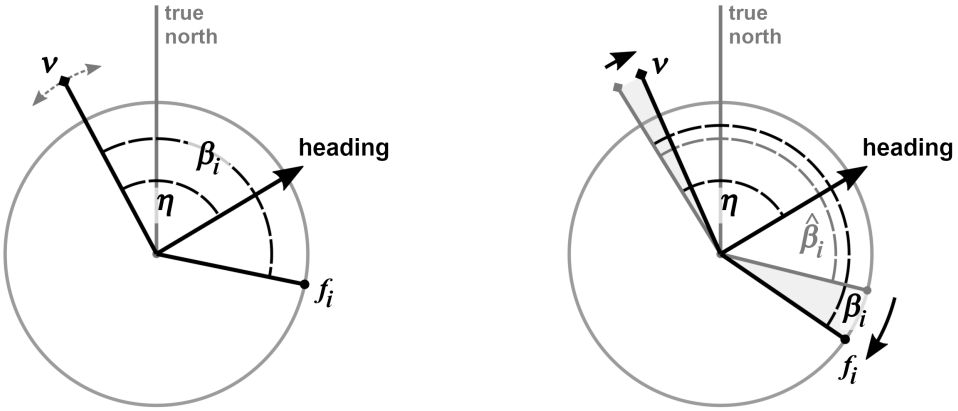


Figure 4.5. Head-direction system. *Left:* During the entire simulation, the system is maintaining a reference direction ν with is initialized to the movement direction in the first frame. Heading angle η and feature bearings β_i are always expressed relative to ν. The "true north" direction is known to the virtual reality simulation, but not to the agent. *Right:* If a feature is detected, its stored bearing label $\hat{\beta}_i$ is compared to the actual bearing in the current image, β_i and the reference direction is updated so as to reduce the difference between $\hat{\beta}_i$ and β_i. Of course, this is done for many features simultaneously, as described in Eq. 4.1.

$\alpha_{ij} = \eta_t$ and $\alpha_{ji} = \eta_t + \pi$, respectively.

The reference direction is constantly affected by a noise process $\boldsymbol{n}$ and updated according to the the available landmark cues, i.e. the bearings of known features. Let F_t denote the set of known features visible at frame t and $\hat{\beta}_i$ be their stored bearings. The agent then compares the current feature bearings with the stored ones and computes the average deviation as a circular mean. The reference direction is then updated as

$$\nu_{t+1} = \nu_t + \frac{\lambda}{|F_t|} \operatorname*{cmean}_{\{i|f_i \in F_t\}} \left(\hat{\beta}_{t,i} - \beta_{t,i} \right) + \boldsymbol{n}, \tag{4.1}$$

where λ is set to 0.05 and the standard deviation of $\boldsymbol{n}$ is set to $\sigma = 0.025$ rad. The circular mean of a set of angles $\{\gamma_i \mid i \in A\}$ is defined as

$$\operatorname*{cmean}_{A}(\gamma_i) := \operatorname{atan2}\left(\sum_{i \in A} \cos \gamma_i, \sum_{i \in A} \sin \gamma_i \right). \tag{4.2}$$

This updating rule attributes the average bearing error to the reference direction.

Figure 4.6. Compass direction drift over a large explored area. The ν estimate deviates substantially (over 90°) from its starting value, but remains locally consistent.

It can compensate for the noise, but introduces a new type of error if the features are unequally distributed in the image. Assume, for example, that the agent relies only on features on its left. As it moves forward, these features will move further to the left, leading to positive deviations $\hat{\beta}_i - \beta_i$. The algorithm will then assume that the reference direction has turned to the left. As a result, the reference direction in a large environments drifts with the agent's position, as is illustrated in Fig. 4.6. However, in prolonged exploration, the assumed reference directions convergence to a stable, locally consistent distribution over explored space.

In addition, the stored bearings for each feature, $\hat{\beta}_i$ are updated at each learning step at which the feature i is re-detected by the iterative mean:

$$\hat{\beta}_{t(l+1),i} = \frac{c_i}{c_i + 1}\hat{\beta}_{t(l),i} + \frac{1}{c_i + 1}\beta_{t(l+1),i}, \tag{4.3}$$

where c_i is a counter stepped at each update and $\beta_{t(l),i}$ is measured relative to the current compass direction ν_t.

Finally, a link a_{ij} may be rediscovered upon a later visit of the same location. In this case the associated direction label α_{ij} is updated as

$$\alpha_{ij}^{\text{new}} = \frac{c_{ij}}{c_{ij} + 1}\alpha_{ij}^{\text{old}} + \frac{1}{c_{ij} + 1}\eta_t. \tag{4.4}$$

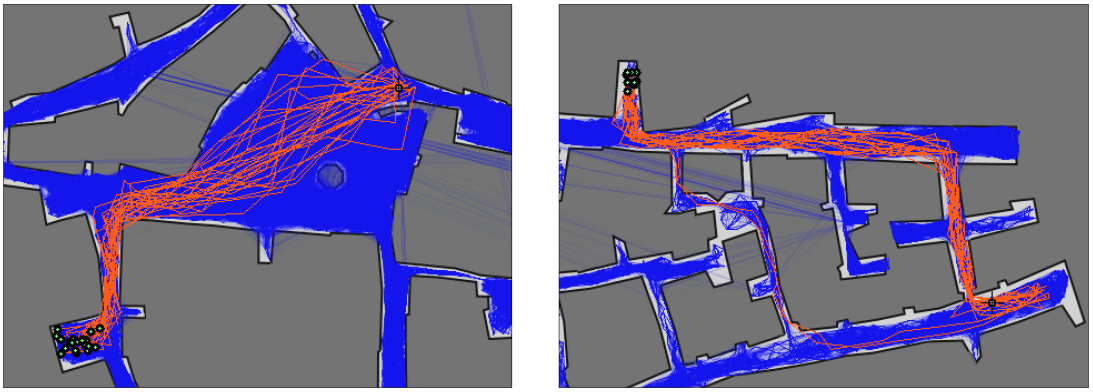


Figure 4.7. Route examples. Two different examples of path bundles (orange) from the agent's current position (black cross) to a goal locations (set of green dots). The blue background shows the edges of the view graph, as described in Fig 4.4

.

Again, this can happen only when the slow learning counter l is stepped. As for the bearings, c_{ij} is a counter stepped at every update and η_t is measured relative to the current reference direction ν. Note that the counters c_i and c_{ij} in Eqs. 4.3 and 4.4 can be avoided by replacing the bearing and heading angles by unit vectors and storing sums of these unit vectors as labels. From the accumulated vectors, the angles can than simply be obtained by the atan2 function.

4.2.5 Pathfinding and voting

In path-finding, the goal location is defined by a set of known features, and can for example be provided by an image depicting the goal. The algorithm then calculates multiple non-overlapping paths from features at the agent's current position to the set of goal features, and uses a voting scheme to obtain navigable trajectories the agent can follow towards the goal location (Fig 4.7).

In each pathfinding event, for one currently visible feature, the shortest path is found to one of the features in the goal set with Dijkstra's algorithm (Dijkstra, 1959). Then, the nodes and edges of that path are temporarily removed from the graph, except for the first and last nodes, and the search is repeated for another randomly selected pair of nodes. Due to the node removal, each path will have zero overlap with all previous paths. Still, when represented in a metric map, path trajectories will be similar due to overlapping place fields.

The search terminates when the pair of randomly selected start and goal nodes

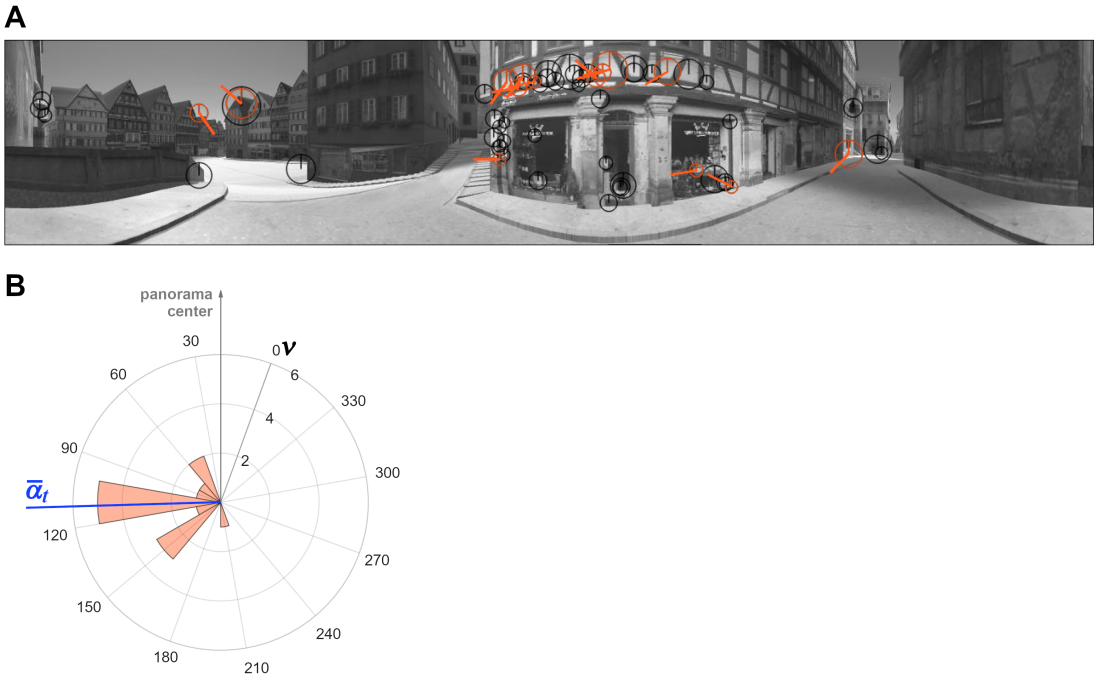


Figure 4.8. Direction voting. A. 360° panorama frame with detected SURF features (black and orange circles with vertical stripe). During path-finding, movement is derived from features that are also part of the path bundle (orange circles): The thick lines originating from the orange features show their respective movement direction vote relative to the reference direction ν (thin vertical stripe). B. The histogram shows the votes sorted into 10°-bins and the resultant mean direction $\overline{\alpha}_t$.

is unconnected in the graph lacking the temporarily removed nodes or when an upper limit has been reached. For example, in the tests detailed in the "Evaluation" section below, we used up to 30 successive Dijkstra searches, but the algorithm regularly found a lower number of routes, depending on the amount of exploration. Note that all edges are considered to have the same length, i.e., Dijkstra paths only differ in the number of edges they traverse.

Once a bundle of paths has been obtained, at the starting location, we could use the initial edge labels α_{ij} to determine the current movement direction. Later however, it is not clear which step of each path applies at each position along the overall travel. Therefore, at each position, we determine the set of currently visible features also included in the present bundle. From each such feature we take the next edge along the respective path and thus obtain a set of movement votes (Fig 4.8).

Each Dijkstra path p in the bundle P has an ordered set of edges E_p =

$\{a_{ij}, a_{jk}, a_{kl,...}\}$ and set of nodes $F_p = \{f_i, f_j, f_k, f_l, ...\}$. At navigation time, the set of visible features is F_t. We now consider the indices of all outgoing edges of currently visible features contained in a path of the bundle, $J_t = \{(i,j) \mid f_i \in F_t \wedge a_{ij} \in \bigcup_{p \in P} E_p\}$. The set of locally applicable motion directions is then given by $\{\alpha_{ij} \mid (i,j) \in J_t\}$. From these we obtain the movement consensus as

$$\overline{\alpha}_t = \underset{J_t}{\text{cmean}}\, \alpha_{ij}, \tag{4.5}$$

where cmean is the circular mean as defined in Eq. 4.2.

The final heading vector $\boldsymbol{\eta}_{t+1}$ is calculated with stiffness $\kappa \in [0,1]$ as

$$\boldsymbol{\eta}_{t+1} = \kappa \boldsymbol{\eta}_t + (1-\kappa)(\cos\overline{\alpha}_t, \sin\overline{\alpha}_t)^\top. \tag{4.6}$$

This results in a smoothing of the trajectory to reduce sway and corner-cutting behavior; κ was set to 0.7.

Moving into the direction of $\boldsymbol{\eta}_{t+1}$ ideally leads the agent along a route specified by the bundle of paths, where it will continue to encounter labeled nodes. To facilitate this process the number of features detected in each frame is doubled during path following. If the number of usable nodes, $|J_t|$, drops below a threshold of two, we assume that the agent has diverged from the path. In this case, a new bundle of Dijkstra paths is calculated with the current feature set F_t as a starting point. If the agent hits a wall, we simply use the Unity obstacle avoidance function which results in wall following in the direction closest to the current heading η.

The algorithm determines if the goal is reached by comparing the set of currently visible features, F_t, to the set of goal features, F_g, and considers the goal to be reached if $|F_t \cap F_g| \;/\; |F_g| \geq 0.35$. Note that there may be some offset between the agent's final position and exact goal location since there is no optimization or "homing" step in our ballistic procedure.

Relying on the average of a set of movement instructions partially solves the problem of wrong edges introduced into the graph due to aliasing, if enough alias-free paths are present. Such aliases are formed occasionally since feature matching relies on visual similarity only, and tend to be shorter than navigable connections (see Fig 4.4). In the Virtual Tübingen environment, identical texture files are occasionally used at different places, leading to an increased number of aliases. However, as long as a sufficient number of correct edges corresponding to navigable

trajectories exist, the votes of the erroneous connections will not cause navigation to fail.

Finally, if the agent is unable to move during navigation, for example due to obstacles or being stuck in a corner, or if no consensus can be found in the set of movement instructions, the bundle of paths is recalculated, which has always solved the problem in our simulation. If the agent ever gets lost, for example because no known features are recognized, it may return to exploration behavior for a short while (e.g., random walk).

4.3 Experiments

4.3.1 Experiment 1: Virtual Tübingen

The algorithm was tested and evaluated in a virtual environment of the downtown area of Tübingen, Germany (3D model based on van Veen et al. (1998)), rendered in the Unity engine (Unity Technologies, 2018). The agent in the virtual environment was equipped with a 360° horizontal FoV and 60° vertical FoV camera projecting to a 1280×240 pixel image. Depending on location, the SURF feature detector would detect some 20 to 350 features per image, ranked by contrast. Of these, up to 30 were used during exploration and up to 60 during path following.

The environment was explored with a random walk heuristic, i.e., the heading η was rotated every 5 to 10 frames by a random angle drawn from a uniform distribution between ± 20 degree. Obstacle avoidance in the exploration phase was realized as a reflection from the obstacle surface.

A first test concerned to continuity of the estimated reference direction ν in loop-closing (Fig 4.9) which is important for consistent movement voting over nearby locations. Fig 4.6 already shows this continuity for the standard version of the algorithm. Here, the noise term $\boldsymbol{n}$ in the update rule for ν (Eq 4.1) was given a fixed bias of 0.3 degree left, leading to a spatially continuous, but large, compass drift. If the agent closes a loop, the ν estimate will therefore change discontinuously. Fig 4.9A shows this situation shortly before the actual loop closing. If the agent now proceeds further, the starting point is recognized, and ν is updated from the current feature bearings according to Eq 4.1. Since this happens at each time step t, the original ν estimate is quickly recovered (Fig 4.9B). If the loops

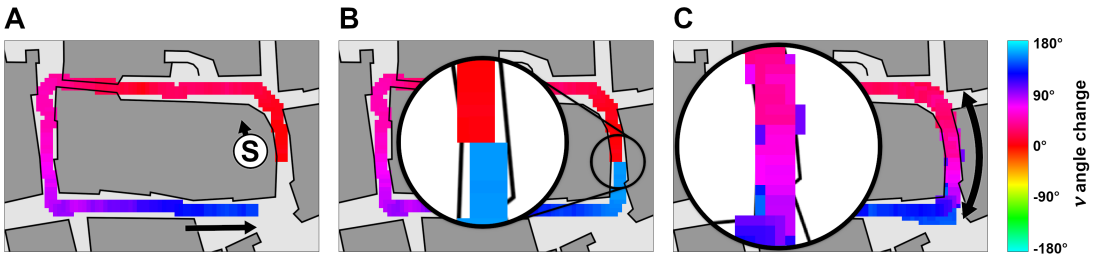


Figure 4.9. Reference direction during loop closing. A: Compass drift accumulates during exploration (effect overdone for demonstration purposes). The agent starts at point S and starts a counter-clockwise loop. B: When a loop is closed, the ν estimate changes discontinuously. Since movement is performed relative to the current ν estimate, sharp angle changes need to be avoided to ensure smooth movement. C: ν estimates are smoothed over repeated visits of the loop closure point.

are repeated, or if the agent travels back and forth across the loop closing point (Fig 4.9C), the ν estimates are spatially smoothed.

In a second test, exploration was continued until the agent had explored every street at least once, resulting in the built-up of a full graph as in Fig 4.4. Five path-finding tasks were defined as pairs of start and goal views. Each task was repeated 20 times, and the traveled distance was measured and compared to that of the shortest possible route (Fig 4.10A). The agent solves each task by first selecting features from the start view and estimating the reference direction ν from the features' bearings. Next, 30 Dijkstra paths are calculated, and from these, the overall trajectory is generated by movement voting as described above.

The trajectories found for a single task may greatly differ between repetitions due to many stochastic influences, such as node selection for start and goal nodes and the noise added to the reference direction update. Further variation is introduced by numerical effects in collision detection and the latency between concurrent components of the programs running the algorithm and simulation. The algorithm may even guide the agent along different roads over multiple trials if they are close in length to the optimal route (see Fig 4.10C,D).

The algorithm managed to guide the agent in a steady directional movement from start to goal in all trials. On average, the agent's routes were only 10 % to 20 % longer than the optimal routes (Fig 4.10B). The agent performed better, i.e., the routes were shorter, when they were leading mostly through roads and alleys rather than traversing large open spaces such as the market square. The algorithm was able to guide the agent even with large drift in the reference direction ν of

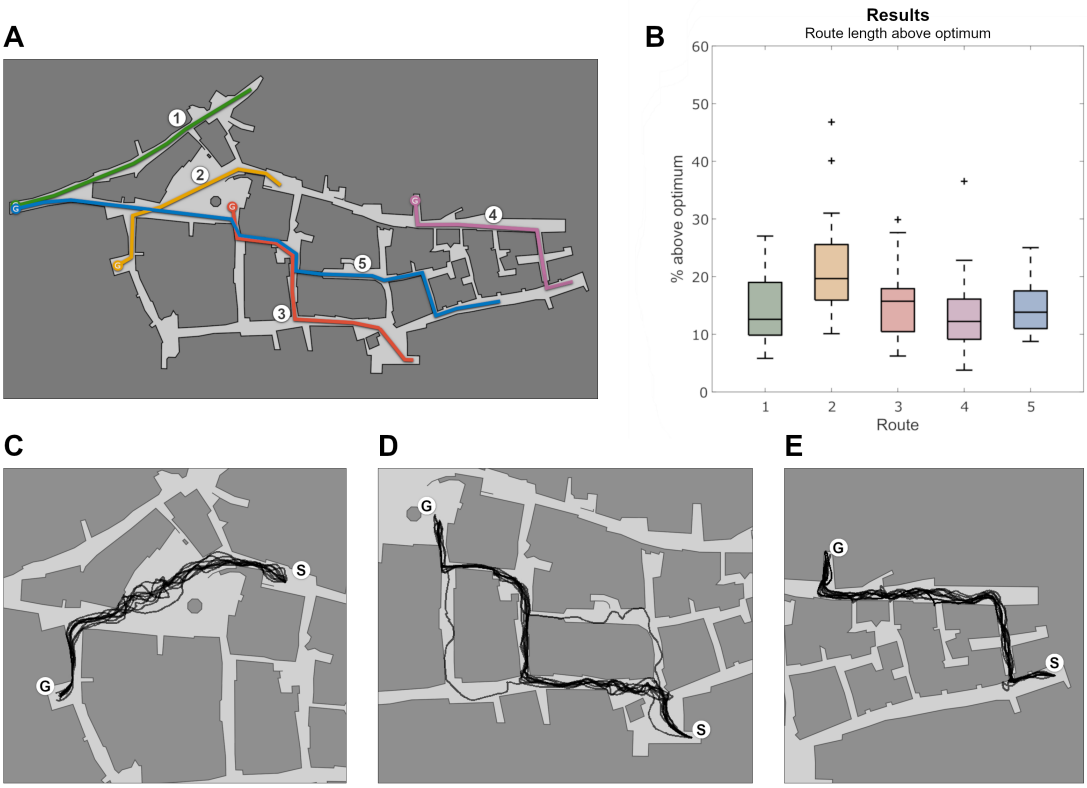


Figure 4.10. Performance in Virtual Tübingen. A: Map depicting the evaluation routes. B: Results of the evaluation. The boxplot shows route length above an optimal trajectory, as depicted in A. Performance is somewhat worse for route 2, because it traverses a wide open area containing lots of distant landmarks, which are worse for exact navigation. C - E: Ten repetitions of each of the routes 2, 3 and 4. C: repetitions of route 2 show larger variations in open spaces. D: repetitions of route 3 show occasional choice of route alternatives as well as directional sway within single repetitions. E: repetitions of route 4 show the lowest variability among the evaluation routes

over 90° offset from the starting ν (see Fig 4.6).

The difference in length between routes through alleys and routes through large open spaces can be ascribed to two factors:

1. Trajectories are not straight lines but reflect the steps of the random walk exploration they were built from. The variation in trajectory directions is lower in narrow alleys since the close walls force the agent to move along the alley in both exploration and path following.

2. Roads, and especially alleys, contain more features with small place fields due to natural occlusion. In wider spaces, features' place fields are larger and

the agent's movement may be influenced by distant features containing movement instructions which are relevant only to a part of the feature's place field.

4.3.2 Experiment 2: Landmark replacement

In a behavioral experiment, H. A. Mallot and Gillner (2000a) studied recognition-triggered response in human subjects by exchanging landmark positions after learning. They found that navigation is impaired if cue exchange leads to inconsistent movement votes whereas normal performance is found for replacements preserving the movement consensus. Navigational decisions thus seem to be based on movement votes associated not with entire snapshots but with smaller image structures such as landmark views. Indeed, votes based on micro-snapshots, although not tested by H. A. Mallot and Gillner (2000a), would lead to the same predictions.

In order to test this behavior in the algorithm, we generated a virtual environment, similar to the one used by H. A. Mallot and Gillner (2000a), consisting of an iterated Y-maze with three junctions (A, B, C), nine object models (landmarks $L_1 - L_9$) placed in the 120° angles between roads, and a start and a goal location (S, G, Fig 4.11). Each place (junction) with its landmarks was surrounded by an opaque cylinder that could be permeated by the agent but prevented it from seeing within-junction features from a distance. Thus, decisions had to be based solely on the features of the current place and new Dijkstra searches were generally initiated when reaching a new junction during path-finding. In addition, environmental features were provided along the streets between junctions.

During learning, the agent explored all places of the environment in an experimenter-controlled scheme. When entering or leaving a place, it learned graph edges and movement labels between features within the place and environmental features. In the test phase, the route from start to goal had to be found in one of four conditions:

1. Control: No landmarks were exchanged.

2. Consistent-across: Landmark replacement was across junctions but with the same associated movement directions, realized by replacing landmarks L_4 and L_6 with copies of L_9 and L_8. In a path to the goal, these landmarks all encode a left turn.

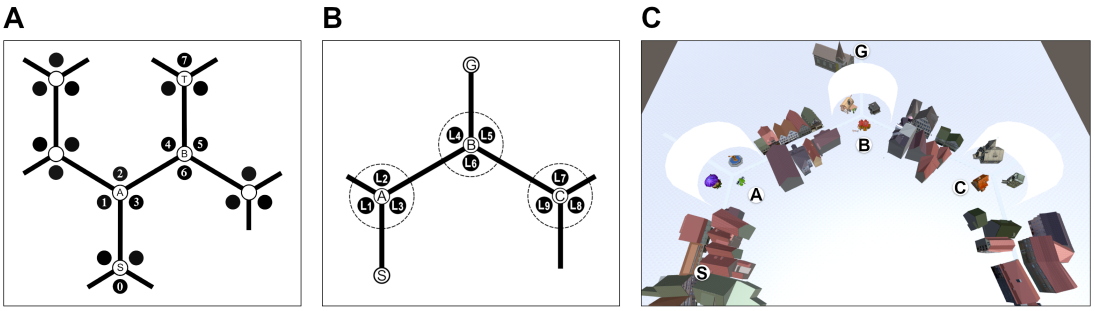


Figure 4.11. Hexatown. A: Left: Original Hexatown setup by H. A. Mallot and Gillner (2000a). Subjects had to learn the route from the starting point S to a turning point T and back to the start. During exploration, they could also visit the unnumbered places. Right: Our Hexatown setup, consisting of three junctions with three landmarks each. The algorithm had to repeatedly guide an agent from the start S to the goal G. Junction B was modified in the different conditions. B: The actual 3D environment. The junctions were connected by short roads lined with buildings to provide guidance outside of the junctions. The landmarks at the crossroads are free assets from the Unity Asset Store (https://assetstore.unity.com/).

3. Inconsistent-across: Landmark replacement was across junctions but with inconsistent associated movement directions, realized by replacing L_4 and L_6 with copies of landmarks L_1 and L_3. In paths towards the goal, Landmarks L_1, L_3 encode a right turn.

4. Within-junction: The lateral landmarks at junction B, L_4 and L_6, were swapped. In this case, the landmarks' features are consistent, but not their position and associated reference direction. In this condition, we expected conflict as well.

The replacing landmarks were rotated to retain their original facing towards the center of the junction, such that many of the features exposed to the agent were also visible after replacement. Note that complete agreement is not possible in the within-junction condition, since the landmarks were only rotated, but not mirrored. Each condition was tested ten times by having the algorithm find a route from start to goal repeatedly; these routes will differ subject to the random elements of the algorithm (e.g., random start and goal selection in path-finding, see section "Pathfinding and voting" above). The evaluation was repeated with five different permutations of the landmark placements, for a total of 200 runs for all four conditions. The algorithm had to learn the environment again for each permutation of the landmark placements. A run was considered a failure if the

Table 4.2. Results of Experiment 2 for each landmark permutation $P_1 - P_5$. p-vales are for paired t-test against control.

	number of successful searches (out of 10)			
	control	cons.	incons.	within
P_1	10	10	0	0
P_2	10	10	9	0
P_3	10	10	0	9
P_4	10	10	0	0
P_5	10	10	0	0
total	**50**	**50**	**9**	**9**
p	–	*n.s.*	0.0019**	0.0019**

algorithm chose the wrong road at the junction B.

The logic of this experiment requires that the goal is reached by left and right turns on different occasions. In the Mallot and Gillner experiment, this was achieved by using one route in forward and backward direction; subjects were not instructed to move towards a specific goal but to proceed in the same direction as during training. Since the algorithm needs an explicit goal specification, we added a third junction C such that the goal has to be reached either by a left or a right turn (see Fig 4.11A).

The results of this experiment appear in Table 4.2. In the control condition, as well as the consistent-across condition, the goal was always reached in all five landmark configurations. The procedure failed in the inconsistent and within conditions. Path length was increased by 5.4 % in the consistent condition as compared to control. In the inconsistent-across and in the within-junction conditions where the agent succeeded to reach the goal, excess path lengths were 30.4 % and 26.8 %, respectively.

The variable performance of the algorithm for different landmark configurations results from the feature selection process as described in section "Pathfinding and voting" above. If many salient features happen to be found on a non-replaced landmark, performance will be high whereas errors are more likely if features from the replacing landmarks are used. The same effect applies to the estimation of the compass direction ν. In addition, ν will also be influenced by the changed landmark bearings of the replacing landmarks(β in Eq 4.1). This leads to occasional

backward turns of the agent at junction B.

Our findings match the results of Mallot and Gillner in the control, consistent-across and inconsistent-across conditions. In the within-junction condition where the algorithm fails, human data show no effect. This may be due to the fact that in the within replacements, novel views of each landmark become visible which look completely different to the algorithm but seem to contain similarities for the human observer. Indeed, humans can easily recognize a church building, say, from different viewing directions even if the visible features differ. They thus seem to use more abstract landmark information than our algorithm.

4.4 Discussion

The aim of this study was twofold: first, to reconcile the ideas of spatial population coding and state-action representations via a voting scheme, and second, to identify a lower bound for the amount of image processing and invariance needed in a topological navigation algorithm. The results show that state-action navigation is possible with vaguely defined position information and that the voting scheme is an efficient way of decision making.

The sought minimality of the algorithm can clearly be claimed for the feature extraction part. Image features are arguably the smallest element of visual input carrying spatial information. The algorithm might even run with still simpler feature types such as Gabor patches or Haar features (see, for example, B. Baddeley et al. (2012), Sheynikhovich et al. (2009), and Smith et al. (2007)), but this was not tested in the present study.

The situation is less clear for the total storage and processing needed to memorize the graph. The Virtual Tübingen experiment was implemented on a standard PC and ran in real time. Typical graphs had 2×10^5 features each represented by its 64-dimensional descriptor plus a bearing angle, which results in some 1.3×10^7 floating point numbers or 52 MB. In addition, 3×10^6 graph links were stored as a list for which $3 \times 10^6 \times \log_2(2 \times 10^5)^2 \approx 10^8$ bits or 13 MB are required. Finally, the heading labels attached to the links take another 12 MB. Thus the map of Virtual Tübingen took some 77 MB which is not much in technical terms (e.g., the equivalent of three high resolution images fetched with a custom camera) but may be beyond the storage capacity at least of small brains.

It should be noted, however, that we did not systematically optimize the algorithm for storage reduction. The dimension of the feature descriptors (64), their resolution (single precision float), the maximum number of new features stored per learning step (30), the maximum number of links added per observed feature transition (3), and the learning rate (every 15 time frames) were chosen freely to get a working version. In order to reliably judge the biological plausibility of the storage requirements of the algorithm, these parameters would have to be optimized and neural encoding schemes would have to be applied.

The algorithm requires comparison operations between each newly detected feature and all stored ones. This is a costly procedure in serial computers but can easily be parallelized in a neural network. Moreover, since the agent is moving in the environment with finite speed (about two simulated meters per second or 13 cm per frame), there is indeed plenty of time for performing these operations. This is in line with the general idea that way-finding in human navigators accounts for just a minor part of the cognitive load of navigation (Spiers & Maguire, 2008). The cost of feature comparisons can also be reduced by hierarchical representations in which the total graph is subdivided into regions and search can then be restricted to one region at a time.

The algorithm makes no use of path integration although some improvement could likely be obtained by way of local metric (Foo et al., 2005; Warren, 2019) or metric embedding (Hübner & Mallot, 2007a). This is mainly a design decision taken in order to test the graph approach in isolation. It turned out, however, that the representation and maintenance of a global reference direction is necessary in order to obtain good results. The reason for this is that the interpretation of feature bearing angles and the directional labels of the graph links must be consistent under parallactic movements of the features in the image of the moving agent. The reference direction used in the algorithm is equivalent to the "point zero" of the ring-attractors used in many models of head-direction cells and path integration (Hartmann & Wehner, 1995; McNaughton et al., 1996; Seelig & Jayaraman, 2015; Taube, 2007). In our implementation, the agent does not actually perform turns, but simply moves in arbitrary directions while the viewing direction (retinal coordinate system) remains fixed relative to the world coordinate system ("true north" in Figure 4.5). If it moves straight, the estimated heading direction might still change, since it is measured relative to the reference direction with in

turn is subject to noise. Overall, the system of reference direction, heading and bearing angles is functionally equivalent to the head direction system. Note that the advantage of an "allocentric" reference direction, even if it is maintained with direct measurements, has also been demonstrated for path integration (Cheung & Vickerstaff, 2010; Vickerstaff & Cheung, 2010).

Path planning is based on multiple Dijsktra searches and the subsequent voting scheme as described in Eq. 4.5. As a result, the path eventually found by the agent is not a chain of features or place fields visited one after the other, but a semi-metric average of the trajectories that each Dijkstra-path results in (cf. Figure 4.1). This is reminiscent of Tolman's (Tolman, 1948) notion of the route as a "strip map", which has a non-zero extension to the sides. Finite-width strip maps should allow smoother navigation than simple place-action chains.

The Dijkstra algorithm was chosen for computational efficiency only and could easily be replaced by biologically more plausible procedures such as the decremented back-propagation of a signal from the goal, as suggested for example by Voicu and Schmajuk (2002). In this case, the bundle of Dijkstra paths would be replaced by a wave of activation sweeping over the feature set and activating the relevant action links. Let g_i denote the value of the decremented back-propagation process for a feature i, i.e. its nearness to the goal. The eventual movement direction would then be calculated as the weighted circular average of movement directions suggested by the links between the currently visible features and all connected features with larger nearness,

$$\bar{\alpha}_t = \operatorname{atan2}\left(\sum_{i\in F_t}\sum_{(i,j)\in E} \max(0, g_j - g_i)\begin{pmatrix}\cos\alpha_{ij}\\ \sin\alpha_{ij}\end{pmatrix}\right) \tag{4.7}$$

where F_t is the set of features visible at frame t and E is the edge set of the graph.

Our algorithm does not employ an explicit homing procedure (Cartwright & Collett, 1982; Franz et al., 1998b; Möller & Vardy, 2006). Of course, the graph does contain three-way associations of the "state–action–next state" type and these association are used in generating the Dijkstra paths during the planning process. However, since we do not search for specific features as subgoals along the route, the information about the next state does not play a role in actual navigation. Indeed, we would not be able to say, which of the many possible next "subgoals" in the Dijkstra bundle should be approached. Navigation is therefore "ballistic",

i.e. it relies only on the two-way associations of states to actions, without using predictions about further states. The reason for this is that we consider homing and search as additional mechanisms that a minimalist model should be able to do without. In richer models, or models using backward causal planning, they are likely to play an important role.

The algorithm has not been developed as a model of a particular biological system. However, on a computational level, we think that its elements are biologically plausible both in terms of possible implementations in neural networks and mechanisms identified in the brain. If the features (micro-snapshots) and their detectors are considered as "cells" with their respective place fields (Figure 4.2), they share some obvious properties with hippocampal place cells in rodents, but at the same time are much simpler. Most notably, our feature cells rely exclusively on visual input and do not depend on path integration, which is absent in our algorithm.

The memory acquired by the proposed algorithm allows to find novel routes and shortcuts by recombination of segments of previously learned routes. It thus qualifies as a cognitive map in the sense of Tolman (1948), O'Keefe and Nadel (1978) or Kuipers (1978). However, it does not build a full metric map in the sense of Gallistel (1990) or the robotic SLAM (simultaneous localization and mapping) literature (see, for example, Durrant-Whyte and Bailey (2006) and Thrun and Leonard (2008)).

4.5 Conclusion

Dual population coding is a novel scheme to combine population coding of places or navigational states with a state-action representation of spatial knowledge by means of action voting (the second population step). The results presented in this paper demonstrate the functionality of the scheme in knowledge acquisition and way-finding. They pave the way for models of the evolution of spatial cognition from vague representations of places and recognition-triggered responses to full-fledged cognitive maps. The model requires the additional representation of an "allocentric" reference direction but no other metric components. Future versions will need to include path integration and spatial hierarchies addressing the granularity of cognitive space.

References

Amari, S.-.-I. (1977). Dynamics of pattern formation in lateral–inhibition type neural fields. *Biological Cybernetics*, *27*, 77–87.

Arbib, M. A., & Lieblich, I. (1977). Motivational learning of spatial behavior. In J. Metzler (Ed.), *Systems neuroscience* (pp. 221–239). Academic Press.

Baddeley, B., Graham, P., Husbands, P., & Philippides, A. (2012). A model of ant route navigation driven by scene familiarity. *PLoS Computational Biology*, *8*(1). https://doi.org/10.1371/journal.pcbi.1002336

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008a). Speeded-up robot features (SURF). *Computer Vision and Image Understanding*, *110*, 346–359.

Bicanski, A., & Burgess, N. (2018). A neural-level model of spatial memory and imagery. *eLife*.

Botvinick, M. M. (2008). Hierarchical models of behavior and prefrontal function. *Trends in Cognitive Sciences*, *12*, 201–208.

Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Byrne, P., Becker, S., & Burgess, N. (2007). Remembering the past and imagining the future: A neural model of spatial memory and imagery. *Psychological Review*, *114*, 340–375.

Cartwright, B. A., & Collett, T. S. (1982). How honey bees use landmarks to guide their return to a food source. *Nature*, *295*, 560–564.

Cartwright, B. A., & Collett, T. S. (1987a). Landmark maps for honeybees. *Biological Cybernetics*, *57*, 85–93.

Cheung, A., & Vickerstaff, R. (2010). Finding the way with a noisy brain [e112544]. *PLoS Computational Biology*, *9*(11).

Cramer, A., & Gallistel, C. (1997). Vervet monkeys as travelling salesmen. *Nature*, *387*, 464.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, *1*(1), 269–271.

Durrant-Whyte, H., & Bailey, T. (2006). Simultaneous localization and mapping: Part I. *IEEE Robotic & Automation Magazine*, *13*, 99–108.

Epstein, R., & Kanwisher, N. (1998). A cortical representation of the local visual environment. *Nature*, *392*, 598–601.

Fleer, D., & Möller, R. (2017). Comparing holistic and feature-based visual methods for estimating the relative pose of mobile robots. *Robotics and Autonomous Systems*, *89*, 51–74.

Foo, P., Warren, W. H., Duchon, A., & Tarr, M. J. (2005). Do humans integrate routes into a cognitive map? map- versus landmark-based navigation of novel shortcuts. *Journal of Experimental psychology: Learning, Memory, and Cognition*, *31*, 195–215.

Franz, M. O., Schölkopf, B., Mallot, H. A., & Bülthoff, H. H. (1998a). Learning view graphs for robot navigation. *Autonomous Robots*, *5*, 111–125.

Franz, M. O., Schölkopf, B., Mallot, H. A., & Bülthoff, H. H. (1998b). Where did I take that snapshot? Scene–based homing by image matching. *Biological Cybernetics*, *79*, 191–202.

Gallistel, C. R. (1990). *The organization of learning*. The MIT Press.

Gaussier, P., Revel, A., Banquet, J. P., & Babeau, V. (2002). From view cells and place cells to cognitive map learning: Processing stages of the hippocampal system. *Biological Cybernetics*, *86*, 15–28.

Hartmann, G., & Wehner, R. (1995). The ant's path integration system: A neural architecture. *Biological Cybernetics*, *73*, 483–497.

Hübner, W., & Mallot, H. A. (2007a). Metric embedding of view graphs. a vision and odometry-based approach to cognitive mapping. *Autonomous Robots*, *23*, 183–196.

Huys, Q. J. M., Lally, N., Faulkner, P., Eshel, N., Seifritz, E., Gershman, S. J., Dayan, P., & Roiser, J. P. (2015). Interplay of approximate planning strategies. *Proceeding of the National Academy of America*, *112*, 3098–3103.

Kuipers, B. (1978). Modeling spatial knowledge. *Cognitive Science*, *2*, 129–153.

Kuipers, B. (2000). The spatial semantic hierarchy. *Artificial Intelligence*, *119*, 191–233.

Lowry, S., Sünderhauf, N., Newman, P., Leonard, J. J., Cox, D., Corke, P., & Milford, M. J. (2016). Visual place recognition: A survey. *IEEE Transactions on Robotics*, *32*(1), 1–19.

Madl, T., Chen, K., Montaldi, D., & Trappl, R. (2015). Computational cognitive models of spatial memory in navigation space: A review. *Neural Networks*, *65*, 18–43.

Mallot, H. A., & Basten, K. (2009). Embodied spatial cognition: Biological and artificial systems. *Image and Vision Computing*, *27*, 1658–1670.

Mallot, H. A., & Gillner, S. (2000a). Route navigation without place recognition: What is recognized in recognition–triggered responses? *Perception*, *29*, 43–55.

Mallot, H. A., & Lancier, S. (2018). Place recognition from distant landmarks: Human performance and maximum likelihood model. *Biological Cybernetics*.

McNaughton, B. L., Barnes, C. A., Gerrard, J. L., Gothard, K., Jung, M. W., Knierim, J. J., Kudrimoti, H., Qin, Y., Skaggs, W. E., Suster, M., & Weaver, K. L. (1996). Deciphering the hippocampal polyglot: The hippocampus as a path integration system. *Journal of Experimental Biology*, *199*, 173–185.

McNaughton, B. L., Battaglia, F. P., Jensen, O., Moser, E. I., & Moser, M.-B. (2006). Path integration and the neural basis of the 'cognitive map'. *Nature Reviews Neuroscience*, *7*, 663–678.

Möller, R., & Vardy, A. (2006). Local visual homing by matched-filter descent in image distances. *Biological Cybernetics*, *95*, 413–430.

Muller, R. U., Stead, M., & Pach, J. (1996). The hippocampus as a cognitive graph. *Journal of General Physiology*, *107*, 663–694.

O'Keefe, J., & Dostrovsky, J. (1971). The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, *34*, 171–175.

O'Keefe, J., & Nadel, L. (1978). *The hippocampus as a cognitive map*. Clarendon.

Röhrich, W., Hardiess, G., & Mallot, H. A. (2014). View-based organization and interplay of spatial working and longterm memories. *PlosONE*, *9*(11), e112793. https://doi.org/10.1371/journal.pone.0112793

Schölkopf, B., & Mallot, H. A. (1995). View–based cognitive mapping and path planning. *Adaptive Behavior*, *3*, 311–348.

Seelig, J. D., & Jayaraman, V. (2015). Neural dynamics for landmark orientation and angular path integration. *Nature*, *521*, 186.

Sheynikhovich, D., Chavarriaga, R., Strösslin, T., Arleo, A., & Gerstner, W. (2009). Is there a geometric module for spatial orientation? insights from a rodent navigation model. *Psychological Review*, *116*, 540–566.

Sivic, J., & Zisserman, A. (2003a). Video google: A text retrieval approach to object matching in videos. *Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003)*, *2*, 1176–1183.

Smith, L., Philippides, A., Graham, P., Baddeley, B., & Husbands, P. (2007). Linked local navigation for visual route guidance. *Adaptive Behavior*, *15*(3), 257–271.

Spiers, H. J., & Maguire, E. A. (2008). The dynamic nature of cognition during wayfinding. *Journal of Environmental Psychology*, *28*, 232–249.

Taube, J. S., Muller, R. U., & Ranck, J. B. (1990). Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *Journal of Neuroscience*, *10*(2), 420–435.

Taube, J. (2007). The head direction signal: Origins and sensory-motor integration. *Annual Reviews in Neuroscience*, *30*, 181–207.

Thrun, S., & Leonard, J. J. (2008). Simultaneous localization and mapping. In B. Siciliano & O. Khatib (Eds.), *Springer handbook of robotics* (pp. 871–889). Springer Verlag.

Tolman, E. C. (1932). *Purposive behavior in animals and men*. The Century Co.

Tolman, E. C. (1948). Cognitive maps in rats and man. *Psychological Review*, *55*, 189–208.

Trullier, O., Wiener, S. I., Berthoz, A., & Meyer, J.-.-A. (1997). Biologically based artificial navigation systems: Review and prospects. *Progress in Neurobiology*, *51*, 483–544.

Unity Technologies. (2018). Unity 2018.1.5f1.

van Veen, H. A. H. C., Distler, H. K., Braun, S. J., & Bülthoff, H. H. (1998). Navigating through a virtual city: Using virtual reality technology to study human action and perception. *Future Generation Computer Systems*, *14*, 231–242.

Vickerstaff, R. J., & Cheung, A. (2010). Which coordinate system for modelling path integration? *Journal of Theoretical Biology*, *263*(2), 242–261.

Voicu, H., & Schmajuk, N. (2002). Latent learning, shortcuts and detours: A computational model. *Behavioural Processes*, *59*, 67–86.

Warren, W. H. (2019). Non-Euclidean navigation. *Journal of Experimental Biology*, *222*. https://doi.org/10.1242/jeb.187971

Wiener, J. M., Ehbauer, N. N., & Mallot, H. A. (2009). Planning paths to multiple targets: Memory involvement and planning heuristics in spatial problem solving. *Psychological Research*, *73*, 644–658.

Wiener, J. M., & Mallot, H. A. (2003). 'Fine-to-coarse' route planning and navigation in regionalized environments. *Spatial Cognition and Computation*, *3*, 331–358.

Wiener, J. M., Shettleworth, S., Bingman, V. P., Cheng, K., Healy, S., Jacobs, L. F., Jeffrey, K. J., Mallot, H. A., Menzel, R., & Newcombe, N. S. (2011). Animal navigation. a synthesis. In R. Menzel & J. Fischer (Eds.), *Animal thinking. contemporary issues in comparative cognition* (pp. 51–76). The MIT Press.

Wilson, M. A., & McNaughton, B. L. (1993). Dynamics of the hippocampal ensemble code for space. *Science*, *261*, 1055–1058.

Chapter 5

Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex[1]

Abstract

Testing optokinetic head or eye movements is an established method to determine visual performance of laboratory animals, including chickens, guinea pigs, mice, or fish. It is based on the optokinetic reflex which causes the animals to track a drifting stripe pattern with eye and head movements. We have developed an improved version of the optomotor test with better control over the stimulus parameters, as well as a high degree of automation. The stripe pattern is presented on computer monitors surrounding the animal. By tracking the head position of freely moving animals in real time, the visual angle under which the stripes of the pattern appeared was kept constant even for changing head positions. Furthermore, an algorithm was developed for automated evaluation of the tracking performance of the animal. Comparing the automatically determined behavioral score with manual assessment of the animals' tracking behavior confirmed the reliability of our methodology. As an example, we reproduced the known contrast sensitivity function of wild type mice. Furthermore, the progressive decline in visual performance of a mouse model of retinal degeneration, rd10, was demonstrated.
Supplemental materials: http://dx.doi.org/10.1037/a0033944.supp

[1]Benkner, B., Mutter, M., Ecke, G., & Münch, T. A. (2013). Characterizing visual performance in mice: An objective and automated system based on the optokinetic reflex. *Behavioral Neuroscience*, 127(5), 788–796. Available from: http://dx.doi.org/10.1037/a0033944

5.1 Introduction

Visual impairment, and even more those diseases which lead to complete blindness, represent the most devastating restrictions on quality of life. Most of such diseases originate already in the retina where they may cause a loss of functional light sensitive cells (Hartong et al., 2006). Until now, there is still no satisfactory treatment of inherited photoreceptor degenerations. Numerous studies are performed in mouse models which were genetically altered (K. H. Kim et al., 2008) to simulate human retinal degenerations (Huber et al., 2009). Despite the growing use of mice in vision research, little basic information is available on their spatial vision and how it might be affected by targeted mutations affecting the visual system (Douglas et al., 2005). Behavioral testing of visual performance offers the opportunity to characterize phenotypes and time courses of their development. Furthermore, the interpretation of behavioral tests can help to draw conclusions on integrated brain function (Abdeljalil et al., 2005).

Visual function in mice can be assessed in many ways. Retinal dysfunction is often associated with morphological alterations, which can be examined by histological approaches, or in vivo examination of the fundus (Ball et al., 2003). To evaluate the functions of the different cell types in the retina, electrophysiological approaches are the method of choice. The a-wave of the luminance electroretinogram provides information on photoreceptor function, the b-wave about bipolar cells and amacrine cells (Pinto et al., 2007). To assess the transfer of visual information to the brain and visual integration, visually evoked potentials (VEP) are useful (Porciatti et al., 1999). In addition to these approaches, behavioral measurements are necessary to assess the impact of genetic and morphologic changes on visual performance. So far, there is only little basic information available on spatial vision of mice (Prusky et al., 2004). A possible reason is that, for a long time, only few behavioral techniques were available to test mouse vision. Most of the previously described experiments are based on reinforcement visual discrimination tasks. Because mice are only moderately adaptive to solve behavioral tasks, the design of such tasks is difficult (Whishaw, 1995) and requires a substantial investment of time to generate valid psychophysical thresholds (Busse et al., 2011; Prusky et al., 2004).

An alternative is to measure the optokinetic response, which manifests itself

in an involuntary head and eye movement. Most animals compensate a global movement of the visual environment by moving their eyes and head to stabilize the image of the visual world on the retina. Such compensatory movements can be triggered by presenting a drifting regular stripe pattern (Mitchiner et al., 1976). Because this method is based on the optokinetic reflex it works without any reinforcement training. Thus, an optomotor testing system offers a simple and rapid method to measure visual performance, such as acuity and contrast sensitivity of adult and developing mice (Douglas et al., 2005). A common way to set up such a testing apparatus is to use a mechanically driven drum covered with vertical black and white stripes at fixed spatial frequencies.

One disadvantage of a fixed stripe pattern is that the bar width subtends different visual angles depending on the mouse position inside the drum. Hence, accurate measurement of the spatial acuity is difficult with such a setup. This disadvantage can be overcome by replacing the mechanical drum with a "virtual reality" cylinder, as first developed by Prusky et al. (2004). This setup, consisting of four computer monitors facing into a square, provides more flexibility in changing the parameters of the presented stripe pattern. In Prusky's testing arena the stimulus is manually adjusted at the beginning of each experimental trial so that the virtual cylinder is centered over the mouse's head. In our present study, we advanced this method to move the center of the virtual cylinder in real time with the freely moving animal.

A second substantial problem concerning the interpretation of published behavioral results is that there is no objective method for assessing the animal's tracking behavior. Usually, the decision whether or not an animal tracked a drifting stripe pattern is based on the subjective assessment of the experimenter. Here, we aimed to minimize the observers influence, and developed an automated and, thus, objective algorithm to score the animal's behavior. By automatically assessing the behavior of each individual animal in real time during the experiment, we can reduce the experimental time and the potential stress for the tested animal. In total, the advantages of our enhanced setup comprise higher flexibility to modulate the parameters of the presented stimuli (spatial frequency, contrast), and reduced experimental bias due to automated and objective behavioral scoring.

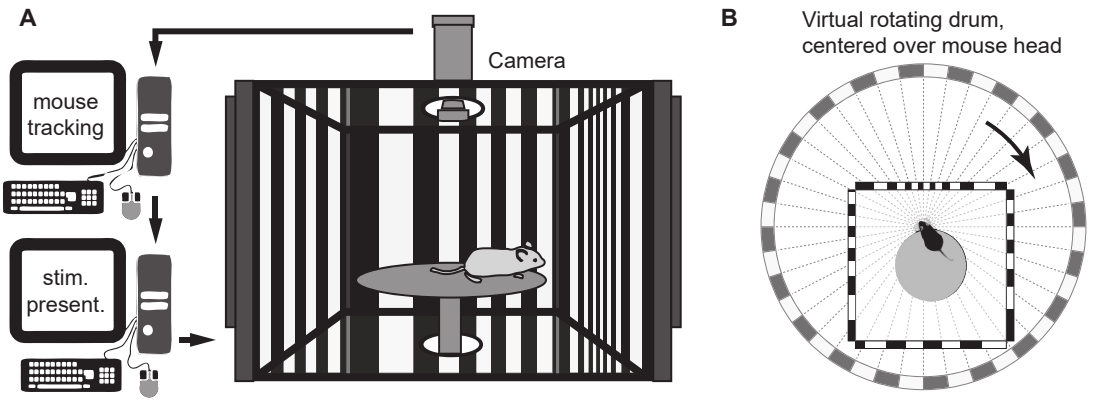


Figure 5.1. Schematic drawing of the experimental setup. A: The "optokinetic drum" virtual arena. Rotating stripe patterns can be flexibly adjusted (width, contrast, and speed). The mouse is observed from above by a camera (Viewer3, BIOBSERVE GmbH), to record its position. B: Stimulus adjustment for spatial resolution. The video-controlled tracking system allows adapting the stripe pattern to the head position in real time. This keeps the spatial frequency of the presented stimulus constant by adjusting the stripe width presented on each monitor.

5.2 Method

5.2.1 Animals

Male and female mice of different age (P24 onward, see Results) were used. All animal strains were originally obtained from Jackson Laboratory/Charles River. Mice were group housed (2–6 animals per cage) in transparent isolated ventilated cages (IVC), with the size 46 cm × 26 cm × 16 cm (l×w×h). Subjects were kept in a 12/12 light/dark cycle. Ambient temperature was standardized to 21 °C, at 55 % relative humidity. Food and water were supplied ad libitum. Breeding pairs were kept together until pups were born, then the male parent was removed. Litters were weaned at P21. Because of its wide use in laboratory studies C57BL/6 mice were used as a standard control group. Furthermore, visually impaired mouse strains were used to investigate their visual abilities during development. Here we used the rd10 strain, which serves as a model for retinal degeneration. This disorder is characterized by a juvenile onset (P16), a progressive disease course and bilateral loss of retinal cells (Chang et al., 2007; Gargini et al., 2006). All experimental procedures adhered to guidelines of the European Union and were approved by the local authorities.

5.2.2 Apparatus

Our virtual-reality arena for vision testing consisted of four 19″ LCD monitors (SV-1900 LED-OEM-HB Sunlight Readable LCD Monitor, Stealth, Woodbridge, Canada) facing into a square to present the visual stimulus to the subjects (Figure 1A). An elevated platform at a height of 14.5 cm was placed in the center of the arena. Mirrors were placed on bottom and top of the arena to increase the impression of an "infinite vertical cylinder." Holes in center of the mirrors (Ø 4.5 cm bottom, Ø 11 cm top) provided openings for the elevated platform and a camera. An analog high sensitivity video camera system (VIDO, Vienna, Austria), B/W camera with Sony 1/3″ super HAD CCD sensor, Model AUCB602, resolution 752×582 pixels) together with an A/D converter (ADVC55, Canopus, Kobe-City, Japan) was placed above the platform. The camera was connected to an interactive tracking system (Viewer3, Biobserve GmbH, St. Augustin, Germany) running on a QuadCore 2.5 GHz, 32 bit WinXP operating system. This provided an online video feedback of the mouse behavior. The Viewer3 tracking-software detected the nose, body, and tail position of the animal without the need of special markings.

Further details: The whole setup was built upon a breadboard (ThorLabs, Munich, Germany). One monitor was mounted on a pivoting arm and functions as a door to the testing arena. We used high bright monitors offering a radiance range of 0.003 to 2 $\mathrm{W/(sr\,m^2)}$ (photon radiance: 8.61×10^{15} to 5.43×10^{18} $\mathrm{photons/(sr\,m^2\,s)}$. The brightness of the four monitors was equalized with 20 kΩ potentiometers. For the adjustment we used an USB2000+UV-VIS Spectrometer combined with a UV-VIS polyimide fiberoptic (Ocean Optics, Filderstadt Germany) covering a wavelength range of 200 nm to 850 nm. To keep the temperature inside the arena in a moderate range, the upper mirror was actively cooled by four 40 mm × 40 mm peltier elements (Peltron GmbH, Fürth, Germany). These were positioned on an aluminum plate of the same size as the top mirror with passive cooling elements made of extrusion profiles. All screens were controlled by one multi monitor graphic card (ATI 2GB HD 5870 Eyefinity 6 from XFX, Hong Kong, China), running on a dedicated computer system (QuadCore 2.13 GHz, 32 bit Windows 7 operating system). The four displays were set in single large surface (SLS) mode. Calculation of the visual stimulus was done in Matlab (R2011, The Mathworks, Munich, Germany) using the Psychophysics Toolbox extensions (Brainard, 1997). For experimental trials with reduced monitor brightness, additional infrared LED light

sources could be used. These were positioned above the animal (Winger Electronics GmbH & Co.KG, Dessau-Roßlau, Germany, 940 nm, 8 mW/sr, size of 5 mm, 50 mA, 0.1 W, angle of aperture 20°) as well as under the platform (Kingbright, Taipei, Taiwan, 940 nm, 25 mW/sr, size of 1 mm, 200 mA, angle of aperture 128°).

5.2.3 Behavioral testing

General outline. Animals were placed inside the virtual optokinetic drum. Horizontally drifting stripe patterns were presented which can potentially trigger the optokinetic reflex, that is, the animal follows the stripe pattern with eye and head movements. We monitored the head movements of the animals to determine if the pattern triggers such a behavior. The parameters of the stripe patterns (contrast, spatial frequency, speed) are adjustable to determine the parameter range to which optokinetic reflex is triggered. Mice can be tested in the optokinetic drum within a few days after eye opening. Animals with progressive retinal degeneration were tested at different developmental stages to characterize the progression of the disease.

Experimental procedure: Overview. Mice were kept in the lab during the day to get used to the new surrounding until the experiment started. For each experiment, the subject was placed on the platform centered in the optokinetic drum. The animal behavior was recorded by the camera system and the mouse position was analyzed in real time.

Each experiment consisted of several trials. Within each trial, the parameters of the visual stimulus were left unchanged. The stimulus consisted of a regular vertical stripe pattern that rotated around the animal. Parameters were: spatial frequency of the stripes, contrast of the stripes, and angular velocity of rotation.

A trial was started manually as soon as the animal had calmed down. The rest of the trial proceeded without user interaction, solely based on the animal's behavior, which was inferred from the coordinates of the body (center of mass) and head (nose tip) returned by the tracking software. The stimulus presentation started as soon as the animal sat still (body coordinates changed by less than one pixel (1 pixel = 0.03 cm) during a period of 0.25 s). The stimulus was shown for at least 1 s and it was terminated either after at most 7.5 s, or when the animal became restless and started walking around. This was detected by a body movement of

more than 10 pixels in a time window of 0.25 s. We call this single presentation of a stimulus a "phase." One experimental trial consisted of several phases. The next phase started as soon as the animal was sitting still again, but not earlier than 2 s after the termination of the previous phase. In this subsequent phase, the same stimulus was presented again, with the exception of the rotation direction: We aimed at presenting each rotation direction (left or right) for approximately the same amount of time during each trial. Thus, for each new phase in the trial, the stimulus was rotated in the direction that had so far been given less total presentation time.

The animal's tracking behavior was evaluated automatically to obtain a so-called "tracking score" after each phase (see section Data Analysis). Thus, we could finish a trial as soon as the tracking score exceeded a certain threshold. The threshold was empirically determined (see section Empirically Determined Threshold). In case the optokinetic reflex could not be elicited, as for example in blind mice, the experimental trial was stopped after a duration of 35 s "calm period," that is, time during the trial in which the animal was neither walking around on the platform, nor making hectic head movements (see "potential tracking time" in Data Analysis). The maximum time limit for a trial in which the animal is very active was 4 minutes. Ultimately, each trial leads to a "yes" or "no" decision, whether tracking occurred or not.

The experimental procedure started always with visual parameter settings for which good tracking behavior was expected. Depending on the "yes" or "no" decision after each trial, the order of tested parameters were adjusted. By proceeding in a staircaselike fashion (see section Data Analysis), we determined the limit of the visual acuity as well as the contrast sensitivity in a fully automated way. The whole experiment was terminated as soon as the perception thresholds for each tested parameter were found. General information about the experimental settings, the video file of the behaving mouse, and the tracked position data were saved for record keeping and potential later offline analysis.

Stimulus presentation. At the beginning of a trial and between the phases, the screens were set to a homogeneous gray value corresponding to the mean brightness of the stripe pattern that was used in that trial. The stripe pattern was presented as a projection of a virtual cylinder located at infinite distance and

centered on the head of the animal, to keep the spatial frequency of the presented stimulus constant despite of varying head-screen distances (Figure 1B, see also Supplementary Video).

To examine the complete range of visual acuity, the spatial frequency of the stimulus was varied between 0.014 and 0.5 cycles per degree (cpd), corresponding to bar widths of 36° to 1°. These stimuli were presented at different contrast levels, namely at Weber contrasts between 0 (equal stripe color) and 641 (black and white stripes). For the experiments described here, rotation speed of the virtual drum was kept constant at 12 °/s, which is in the optimal range to elicit optokinetic reflex in mice (Abdeljalil et al., 2005; Lagali et al., 2008; Mitchiner et al., 1976).

Further details: The center of the virtual cylinder surrounding the mouse was set to the assumed eye positions of the mouse. The assumed eye position was calculated based on the data returned by the tracking software by weighting the nose-body vector by the factor 0.7. The center changed to its new position with a sigmoidal velocity profile. Thus, irritations of the animals by jerky stimulus movements can be avoided, even if this means a slight delay in correcting head-screen distances. To prevent unnatural jittering of the projected stripe pattern, small body movements up to 45 pixels did not lead to a recentering of the pattern.

5.2.4 Data analysis

The goal of the data analysis was to determine if the rotational head movements of the animal sufficiently coincide with the rotational movements of the visual stimulus. The analysis was performed in several stages, which are described in detail below. Stage I: Comparison of the animal's behavior with the stimulus. Stage II: Assignment of a behavioral score to each time point (i.e., to each movie frame). Stage III: Assignment of an overall behavioral score. Stage IV: Comparison of that score with a threshold to reach a yes/no decision.

Stage I: Comparison of the animal's behavior with the stimulus

During the experiment, the animal was tracked from above by a video camera system. The position of head (tip of the nose), body (center of mass), and tail were logged frame by frame. Based on this data the behavior of the animal was analyzed by the following four steps (Figure 2).

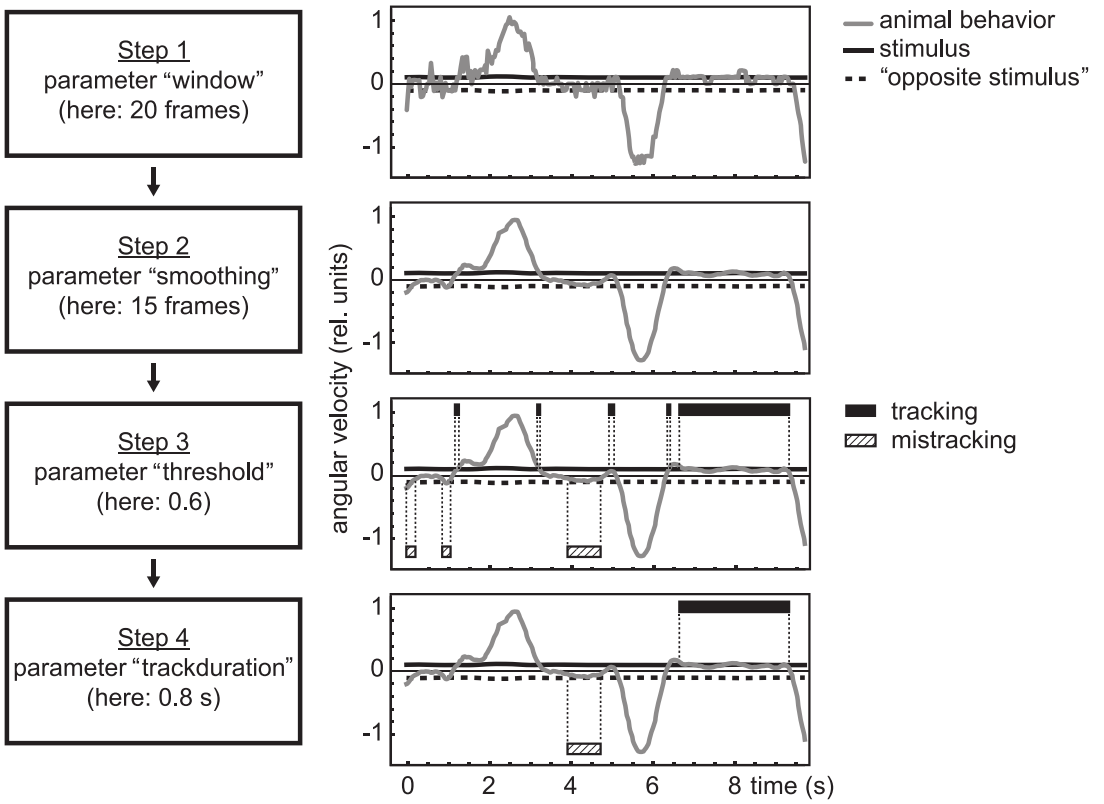


Figure 5.2. Step by step processing of observed angular velocities. Angular velocity of head movement (gray line) recorded during a single trial (positive/negative amplitude indicates left/right rotation of the head). Head angular velocity is compared with rotation speed of the projected stripe pattern (solid black line) and to the "opposite stimulus" (dashed line). Matching velocities are highlighted with solid and striped rectangles, respectively.

Step 1. Calculation of the angular velocity of head movement. We measured the angle between the two body-head vectors at two time points, separated by a certain number of video frames (parameter "window," see below). This yielded a time series, representing the angular velocity of the animal's head movement. Positive values corresponded to leftward turns, negative values to rightward turns. Larger absolute values corresponded to faster rotational movements of the head.

Step 2. Smoothing of the angular velocity trace. Smoothing was performed with a Savitzky-Golay filter with a certain filter length (parameter "smoothing," see below).

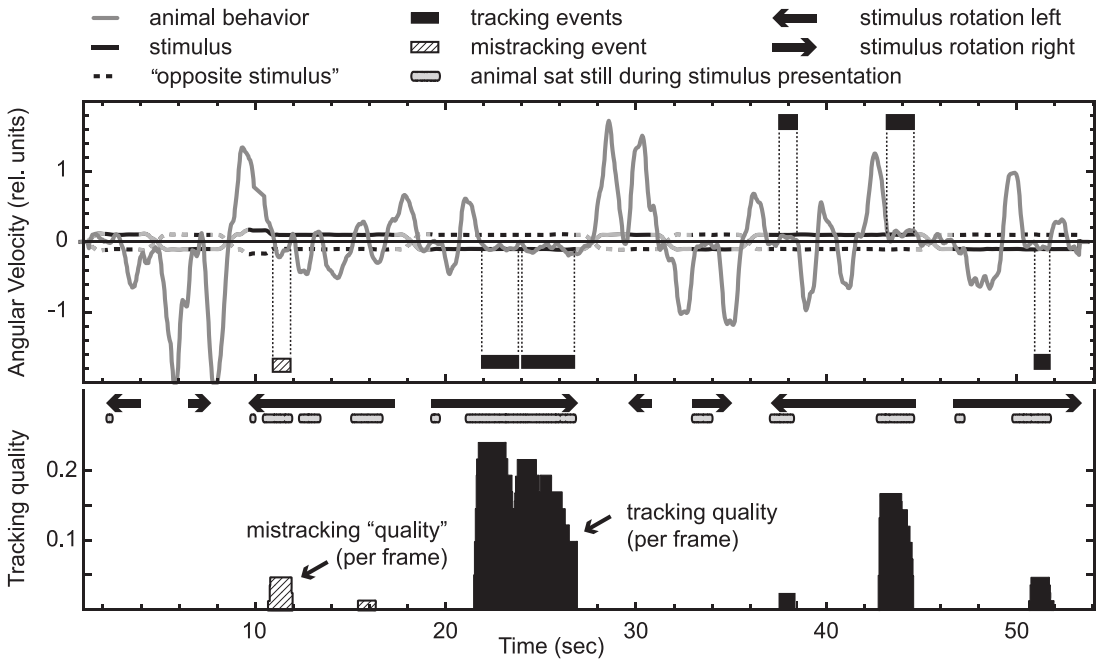


Figure 5.3. Mouse tracking behavior and its analysis. Top: Angular velocity of head movement (gray line) recorded during a single trial (positive/negative amplitude indicates left/right rotation of the head). Stimulus (black line, 0.05 cpd, Weber contrast 0.4) moved at 12 °/s either to the left or right (indicated by arrows below the plot). "Tracking" events (highlighted in black boxes) were determined as close correspondence between head and stimulus movement. Corresponding head movements in the opposite direction are characterized as "mistracking" (highlighted in cross-hatched boxes). Stimulus was presented when the animal was sitting still. Bottom: Assessing tracking quality. Only times were taken into account during which the animal showed little motion ("potential tracking time," gray bars), to make the scoring independent from the animal activity. Raw data (head-body angular velocity) is subjected to the analysis described above (see also Figure 2). "Tracking quality" is the percentage of parameter combinations during which tracking was detected (plotted as histogram).

Step 3. Preliminary tagging of video frames as "tracking," or "mistracking." Each video frame was preliminarily tagged as "tracking" when the filtered angular velocity of the head movement was sufficiently close (parameter: "threshold," see below) to the angular velocity of the presented rotating stimulus. Video frames with corresponding head movements in the opposite direction were tagged as "mistracking." This was used as a control against randomly detected tracking events, and against biases of the animal to rotate in a specific direction.

Step 4. Final classification of "tracking" and "mistracking" events. In order for mouse behavior to qualify as a "tracking event" (or a "mistracking event"), we applied an additional criterion: We demanded that consecutive video frames tagged as "(mis)tracking" exceeded a certain duration (parameter "trackduration," see below). This guards against events during which the head angular velocity is randomly aligned with the stimulus angular velocity. All other video frames were classified as "nontracking."

Taken together, we used four parameters to analyze the animal's tracking behavior ("window," "smoothing," "threshold," and "trackduration"). The parameter values were:

1. Parameter "window": The angular velocity was calculated by taking the difference of the angles over 20 or 25 frames.

2. Parameter "smoothing": We smoothed the time series of the body-head angular velocities with a polynomial regression filter (Savitzky-Golay-Filter). The filter length was either 15 or 20 frames.

3. Parameter "threshold": To be tagged as "tracking," the animal's angular velocity $v(a)$ had to be within a band around the stimulus velocity $v(s)$ such that $(1-k)\,v(s) < v(a) < (1+k)\,v(s)$, with k (the threshold) taking on the values 0.3, 0.6 or 0.9. If the same condition held true for $-v(a)$, the corresponding frame was tagged as "mistracking."

4. Parameter "trackduration": Too short "tracking" sequences were not considered ($< 0.8\,\mathrm{s}$ or $1.2\,\mathrm{s}$), and all frames in that sequence were set to "nontracking."

Stage II: Assignment of a behavioral score to each time point

Each of our four parameters took on two or three different values, leading to 24 different parameter combinations. For each given combination, any video frame is either classified as "tracking," "mistracking," or "nontracking." Overall, each frame is then assigned a value corresponding to the percentage of "tracking" classifications (visualized as a histogram in Figure 3, bottom). For example, a frame that got characterized as "tracking" in 12 out of 24 parameter combinations is assigned a tracking-value of 0.5. This procedure rewards video frames with good

tracking behavior. For example, a frame fulfilling the "tracking" requirements with a strict threshold will also automatically fulfill the requirements with a more lenient threshold, thus leading to a higher score.

Stage III: Assignment of an overall behavioral score

Finally, we converted these per-frame tracking scores into an overall tracking score for the whole trial (Figure 3). For this quantification, we kept a record of the times during the trial that we call "potential tracking times." All video frames were added to the "potential tracking time" that fulfilled the following three conditions for at least one parameter combination:

1. The stimulus was present on the screen;

2. Angular head movements were slower than 2.5 times the rotation speed of the presented stimulus, thereby discarding times of hectic head movements;

3. Body movements across "window" frames were smaller than $2\sqrt{\text{window}}$ pixels.

The last two criteria made sure that we only considered those times in the trial during which the animal did not actively explore the environment. Without excluding these times, the overall tracking score would be artificially reduced. The tracking score T was calculated by summing the per-frame tracking scores, and multiplying that sum by the standard deviation of all per-frame scores during the "potential tracking times." The same procedure was repeated for the "mistracking" events, to obtain a mistracking score M. The final behavioral score was then obtained by

$$\text{behavioral score} = T\,\frac{T-M}{T+M}\,.$$

This score becomes negative when the animal tends to turn more in the opposite direction as the presented stimulus, and positive otherwise. Importantly, this type of analysis gives a score that is robust against the length of the experiment. For example, if one concatenates the same video and analyzes this new (twice as long) video, one obtains the same score.

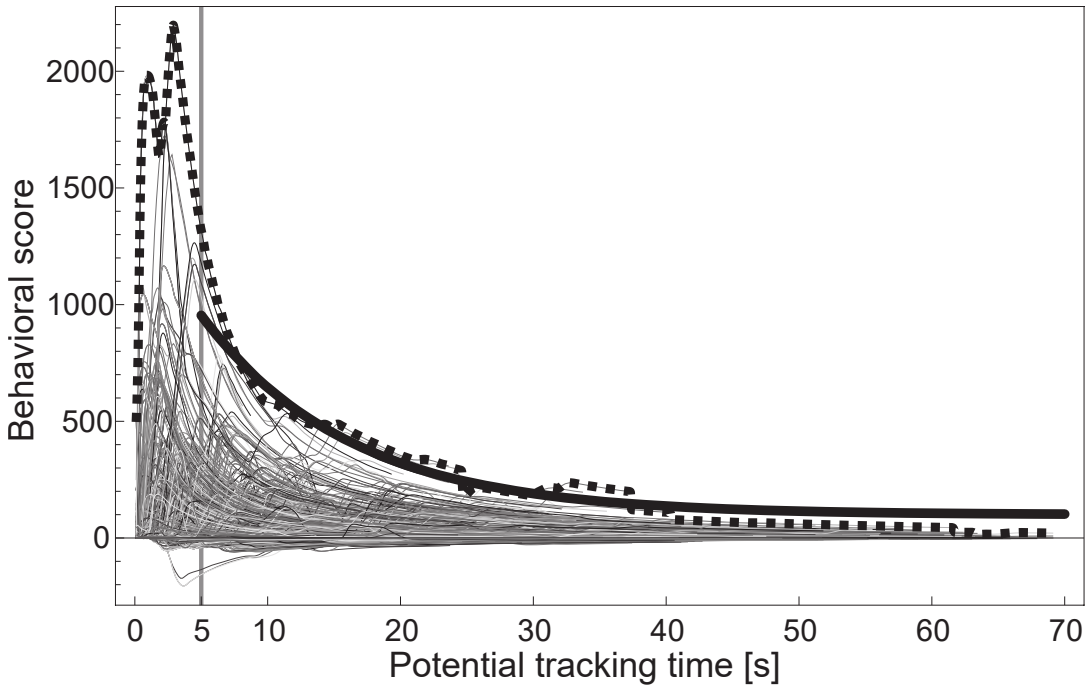


Figure 5.4. Adaptive termination score. Evolving score of 31 tested animals at a contrast of 0 (thin lines represent the behavioral score as a function of experimental time of 4590 trials; 559 real trials plus 4031 randomly shuffled fake trials). Thick dashed line: 100 % quantile of all analyzed trials. Thick solid line: fitted exponential function which is used as final threshold. During the experiment, we required at least 5 s of "potential tracking time," indicated by the vertical gray line, to characterize an animal's behavior as "tracking."

Stage IV: Comparison of that score with a threshold to reach a yes/no decision

The behavioral score can be calculated and updated any time during the trial. At any time point, we could therefore decide if the animal showed tracking behavior against the presented stimulus or not. To do this, we converted the final behavioral score into a "yes" or "no" decision. For a "yes" decision three criteria had to be fulfilled:

1. The score exceeded an empirically determined threshold (Figure 4);
2. The score calculation is based on at least 5 s of "potential tracking time";
3. Both rotation directions were presented nearly the same time (40 % matching).

The last two criteria reduce the risk of detecting false positives. The threshold is explained in the next section.

5.2.5 Empirically determined threshold

To find a useful threshold for a "yes" decision we evaluated the behavior of a wide range of animals. We tested the natural behavior of mice inside the setup while presenting an invisible stripe pattern (contrast of 0, i.e., a homogeneous gray screen) that cannot trigger the optokinetic reflex. With our algorithm, we calculated behavioral scores against this invisible stimulus. Any positive behavioral score is therefore attributable to chance. Several different mouse strains were used to get results from animals with a broad genetic background (C57Bl/6, rd1, rd10, C3H). In total we tested 31 animals with 559 trials for this analysis. To enlarge this data set we shuffled real head and body positions with randomly generated sequences of rotation angles of the pattern, resulting in 4031 additional trials. For each trial, we calculated the evolution of the animals' behavioral scores in 0.1 s temporal resolution. As a result we obtained a distribution of scores for each time point (Figure 4).

We took as a preliminary threshold the 100 % quantile of that distribution, that is, all trials had a score that was lower than this preliminary threshold. Finally, we fit an exponential function to the preliminary thresholds to yield our final threshold

function. Our empirically determined threshold thus describes the maximum score for natural behavior which we expect of nontracking (or blind) mice. If the score of an animal exceeds the threshold during the experiment, it can be classified as seeing with very high chance. Note that the threshold is not a single number, but it depends on the duration of the experiment. Earlier in the experiment, the calculated behavioral score has to be higher in order for the animal to cross the threshold.

5.2.6 Staircase

Each experiment started under conditions (width of stripes and contrast level) which are normally easily recognized by mice. To reduce the required number of trials, the experimental procedure was adapted to the individual performance of an animal. The order of experimental conditions was determined based on the result of the previous trials. Here we use a simplified up-down adapted staircase method. In case the animal exceeded the threshold ("yes" decision), the difficulty was increased in the following trial. The higher the score in the previous trial, the more difficult was the subsequent trial. In case the animal could not reach the threshold, the trial was repeated for a total of three times under identical testing conditions. If this condition could still not trigger tracking behavior, the next easier testing condition was chosen. Thus, we approach the individual limit of perception until the score could not reach the threshold anymore. To ensure that negative outcomes were not caused by lack of the animal's motivation to participate in the experiment, a final trial with an easy condition was performed.

5.2.7 Statistics

As statistical test of significance the Kruskal-Wallis one-way analysis of variance was chosen, with level of significance of $\alpha = 0.05$. Kruskal-Wallis is a nonparametric test whether samples originate from the same distribution, equivalent to the parametric one-way analysis of variance (ANOVA). When we compared three distributions, a Bonferroni correction was made to test subgroups against each other.

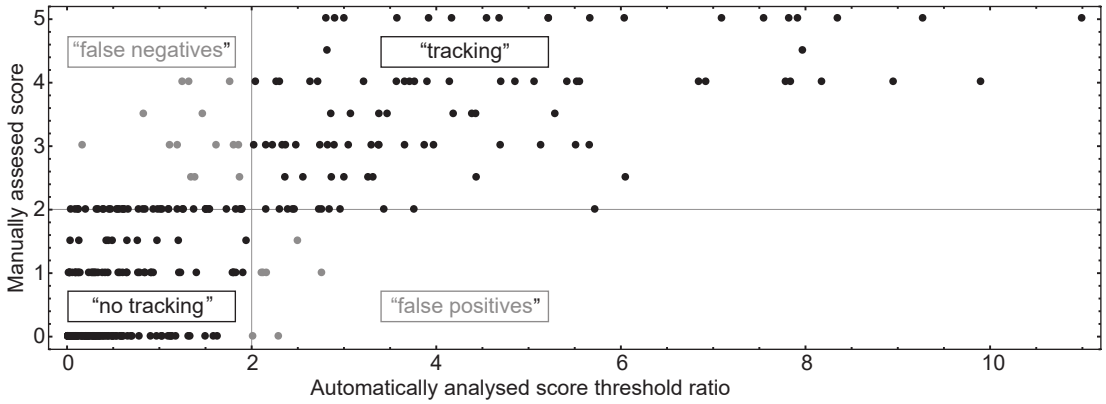


Figure 5.5. Manual assessment versus automated scoring. Behavioral tracking score ($n = 7$ mice, 321 trials). Manual assessment (vertical axis) plotted against calculated "tracking score" (horizontal axis). Corresponding evaluations are shown as black dots. Inconsistent results between manual assessment and automatically calculated scores (false positives and false negatives) are marked gray.

5.3 Results

5.3.1 Evaluation of automated scoring

Most studies using the optokinetic reflex assess the animal's behavior manually. The experimenter decides if the animal has followed the rotating stripe pattern or not. To assess the quality of our automated analyses, we compared the automatically obtained score with a manual scoring scheme. We analyzed 1-min long sequences of 321 experimental trials ($n = 7$ mice) with different stripe widths and contrasts. Each video was analyzed by two independent observers, and classified with a value between 0 and 5: 0 = animal was not paying attention to stripe pattern during the trial (grooming, sleeping, exploring environment); 1 = no tracking; 2 = maybe tracking, not sure; 3 = certainly tracking, but not much; 4 = more tracking; 5 = superb tracking. Figure 5 shows the results of the manual assessment plotted against the automatically calculated "tracking score." Here, we do not show the raw score, but 2(score/threshold). In other words, values larger than 2 correspond to scores in which our automatic scoring algorithm would have led to the conclusion that the animal tracked the rotating stripe pattern.

Compared with manual assessment, our algorithm detected six false positive results, which corresponds to a false positive error rate of 1.9 %. False negative decisions occurred in 14 cases, or 4.4 %. In 248 of 321 events (77 %) the decisions

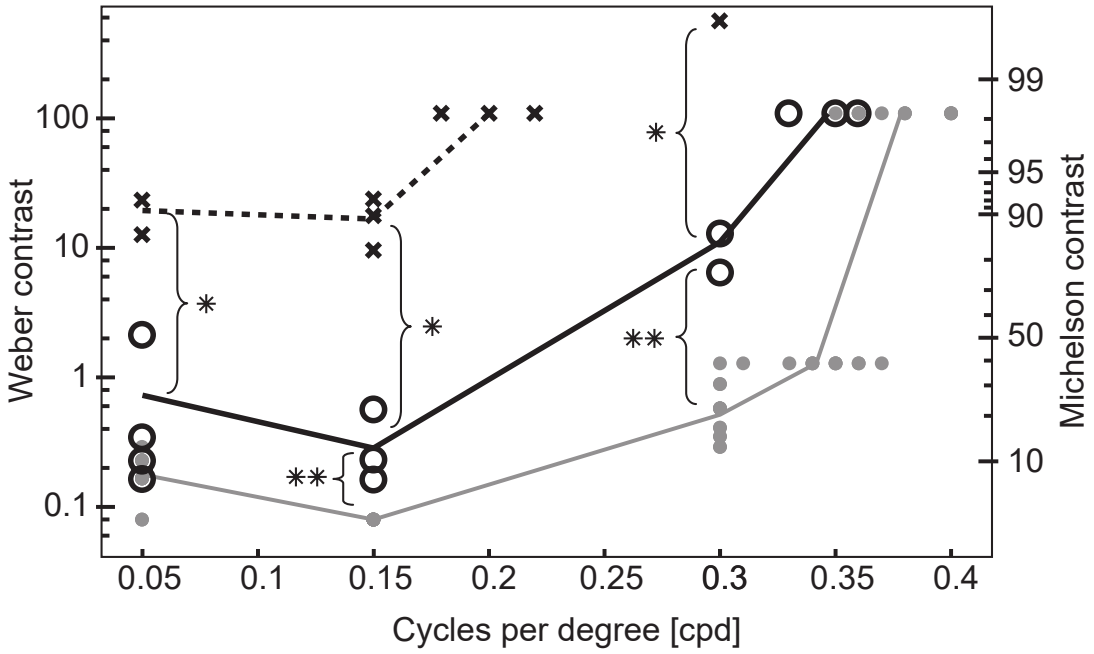


Figure 5.6. Age dependent loss of visual performance of rd10 mice (black) in comparison to the visual performance of C57Bl/6 mice (gray). Contrast sensitivity indicated as Weber contrast (left vertical axis, logarithmic scale) is plotted against the presented cycles per degree (horizontal axis) of adult C57Bl/6 mice (gray, $n = 12$, P59–63) compared with two age groups of retinal degenerating mice (rd10). Young group (solid black) contains four mice (P24–P32), old group (dashed black) contains three mice (P86–P91). The right vertical axis shows corresponding values in Michelson contrast for comparison. Asterisks show significant different values (Kruskal- Wallis test, $\alpha = 0.05$) $^*p < 0.05$. $^*p < .001$. In addition, for C57Bl/6 mice all contrast sensitivity values (0.05 cpd, 0.15 cpd, and 0.3 cpd) are significant different from each other with $p < 0.001$ (including Bonferroni correction). The determined upper thresholds of spatial resolution at two different contrast levels differ significantly as well with $p < 0.001$ (Kruskal- Wallis test, $\alpha = 0.05$). Rd10 mice were compared old versus young and young versus control group each at identical grating acuity. At 0.05 cpd young rd10 mice do not differ from control with $p = 0.06$. The comparison of the upper thresholds at 108 contrast results in a significant difference only between old animals and control group with $p = 0.003$.

matched the observer's assessment. An additional 53 cases (16.5 %) with the ambiguous manual score "2" were sometimes recognized as tracking by the algorithm, sometimes as nontracking.

5.3.2 Proof of concept – characterization of C57Bl/6

Using our enhanced optokinetic drum we were able to characterize the visual performance of different mouse strains in an easy and objective way. As a first proof of concept we used the well-known C57Bl/6 mouse line. For a comprehensive characterization we measured visual acuity as well as contrast sensitivity. Repeating the experimental trials on the next day resulted in similar thresholds, which indicates good reproducibility of these measures. As control, we tested five old rd1 mice (>P200) at the same conditions for contrast sensitivity. At this age, known from recent publications, these animals are blind. Only in one of 45 trials, tracking behavior was incorrectly detected (false positive rate 2.2 %, data not shown).

Figure 6 shows the visual performance of C57Bl/6 mice with a solid gray line. The required contrast (vertical axis, logarithmic scale) is plotted against the presented cycles per degree (horizontal axis). In this experiment, 12 adult mice (P59–63) were tested. To define a contrast sensitivity threshold for each animal, we used three distinct stripe widths of 0.05, 0.15 and 0.3 cycles per degree (cpd) and varied the contrast of the stripes. Contrast was reduced stepwise (see section Method: Staircase) until no tracking behavior could be triggered. Using 10° wide stripes (0.05 cpd), the mean threshold was 0.18 ± 0.02 Weber contrast (all errors are given as SEM). Under optimum conditions with a narrower stripe width (3.3°, 0.15 cpd) all animals were able to see 0.08 contrast. At 0.3 cpd (1.6° stripe width) the mean contrast threshold was 0.52 ± 0.06. To identify the upper threshold of visual acuity the contrast level was kept constant at low (1.28 contrast) and high conditions (108.9 contrast) and the stripe width was varied. If a stimulus was presented at a contrast of 1.28, the animals needed at least 1.35° to 1.6° wide stripes (0.37 to 0.3 cpd) to trigger tracking behavior. The average resolution required was $0.34\,\text{cpd} \pm 0.006\,\text{cpd}$. A considerably higher contrast of 108.9 shifted the upper threshold of grating acuity to between 1.4° and 1.25° stripe width (0.35 cpd to 0.4 cpd), with an average of $0.38\,\text{cpd} \pm 0.005\,\text{cpd}$.

5.3.3 Characterization of retinal degeneration in rd10 mice

This setup was designed to investigate visual performance not only of wild type animals but also for visually impaired mice. We tested the influence of age on the visual performance of retinal degenerated mice (rd10). In comparison with the rd1

mouse line, retinal degeneration in rd10 mice has a later onset; they are, therefore, a better model for the time course of human retinitis pigmentosa (Bowes et al., 1990; Chang et al., 2007; Gargini et al., 2006). Here we tested rd10 mice ($n = 7$) in two different age groups. Testing procedure and parameters were the same as described above for C57BL/6, with the exception that only the higher contrast level (108.9) was used to define the upper threshold of visual acuity (Figure 6, black solid and dashed lines).

Young rd10 mice ($n = 4$) at the age of P24–P32 show at 0.05 cpd (10° stripe width) a mean contrast sensitivity threshold of 0.73 ± 0.48. Under optimum condition at a visual acuity of 0.15 cpd (3.3°) the mean contrast of only 0.28 ± 0.1 was necessary to trigger tracking behavior. Compared with this, a higher contrast of 11.12 ± 1.56 is needed to elicit tracking at 0.3 cpd (1.6°). The upper threshold measured at contrast of 108.9 was identified at the range of 0.33 cpd to 0.36 cpd (1.5° to 1.38°), with an average of 0.35 cpd $\pm$ 0.009 cpd.

Old rd10 mice ($n = 3$) at the age of P86–P91 showed significantly reduced visual performance at all points measured. At 0.05 cpd (10°) the animals needed a contrast level of at least 12.67 with a mean value 19.43 ± 3.38. Even under optimum conditions of 0.15 cpd (3.3°) the animals required almost the same contrast, 16.7 ± 3.86. At a resolution of 0.3 cpd (1.6°) tracking could only be observed in one animal at a very high contrast of 567.69. Under these conditions both other animals didn't show any tracking behavior at several measurements. The upper threshold, measured at a contrast of 108.9 was determined at a range of 0.175 cpd to 0.22 cpd (2.86° to 2.27°) with a mean value of 0.2 cpd (2.5°).

All curves show a similar overall shape; in all measurements the contrast sensitivity was best at a grating acuity of 0.15 cpd (3.3°). With progressive degeneration, vision was dramatically impaired, reflected by reduced contrast sensitivity as well as a reduced upper threshold of visual acuity.

5.3.4 Flexibility across species

With our enhanced version of the optokinetic drum, we found several positive effects when adapting the stimulus presentation to the animals' head position. By this we kept the spatial frequency of the stimulus constant despite varying head-screen distances. To figure out the versatility of our setup we expanded our selection of testing subjects to other species. In a couple of experimental trials

we could show that the automated process of stimulus adaptation is very flexible. It was also possible to trigger the reflex in fish (tested with e.g., Triplefin, family Tripterygiidae), which could move freely in a cylindrical water tank. Even the adaptation of the stripe width to the animals' position works quite well. Short movies of other species behaving in the optokinetic drum can be found in the supplementary video.

5.4 Discussion

The goal of our study was to develop an enhanced version of an optokinetic drum. The automated analysis offers the great opportunity to be more flexible and to adapt the testing parameters to the individual visual performance of the animal, necessitating fewer experimental trials. Furthermore, real-time centering of the stimulus enhances the testing procedure as well, as we can compensate for the animal's movement. In addition, real-time detection of the animal's behavior also offers the possibility for an online assessment. In doing so, we were able to implement an automated and completely unbiased evaluation of tracking behavior. The comparison of automated and manual assessment has shown that our algorithm tends to be conservative, preventing false positives. Visual performance can thus be determined effectively. This increases time efficiency and thus results in shorter experimental times.

Our experience has shown that it is helpful to habituate mice to the setup one day before starting experimental series. Each time, the mice were handled for a few minutes and then put on the platform inside the drum. Another important requirement for successful data acquisitions is to keep the temperature inside the testing arena in a moderate range. In our case we solved this problem by cooling the lid, which made the animal feel more comfortable and less nervous. Furthermore, changing platforms for each different animal group helps to avoid irritation through smells from other animals.

Our data shows that we were able to confirm contrast sensitivity and visual acuity in C57BL/6 mice, but we are more flexible in adjusting our stimulus. In comparison with all established methods so far, our evaluation is completely unbiased from the judgment of the experimenter. Old rd1 mice were used as a control group and validated our analysis procedure for blind animals. These animals were

highly reliably characterized as blind test subjects.

In experiments on rd10 mice we could confirm published results on retinal degeneration depending on the developmental stage of the animals. Older animals reached significantly lower scores in their tracking behavior. For degenerated mice, a higher contrast was needed to trigger tracking behavior. These results are in line with characterizations of rd10 mice based on OCT, ERG, and other behavioral tests (Fischer et al., 2009; Gargini et al., 2006; Pang et al., 2011; Thomas et al., 2010).

In summary, the advantages of our enhanced virtual optokinetic drum are higher flexibility to modulate the parameters of presented stimuli, as well as a high degree of automation.

References

Abdeljalil, J., Hamid, M., Abdel-mouttalib, O., Stéphane, R., Raymond, R., Johan, A., José, S., Pierre, C., & Serge, P. (2005). The optomotor response: A robust first-line visual screening method for mice [Publisher: Elsevier BV]. *Vision Research*, *45*(11), 1439–1446. https://doi.org/10.1016/j.visres.2004.12.015

Ball, S. L., Pardue, M. T., McCall, M. A., Gregg, R. G., & Peachey, N. S. (2003). Immunohistochemical analysis of the outer plexiform layer in the nob mouse shows no abnormalities [Publisher: Cambridge University Press (CUP)]. *Visual Neuroscience*, *20*(3), 267–272. https://doi.org/10.1017/s0952523803203059

Bowes, C., Li, T., Danciger, M., Baxter, L. C., Applebury, M. L., & Farber, D. B. (1990). Retinal degeneration in the rd mouse is caused by a defect in the $\upbeta$ subunit of rod cGMP-phosphodiesterase [Publisher: Springer Science and Business Media LLC]. *Nature*, *347*(6294), 677–680. https://doi.org/10.1038/347677a0

Brainard, D. H. (1997). The Psychophysics Toolbox [Publisher: Brill]. *Spatial Vision*, *10*(4), 433–436. https://doi.org/10.1163/156856897x00357

Busse, L., Ayaz, A., Dhruv, N. T., Katzner, S., Saleem, A. B., Scholvinck, M. L., Zaharia, A. D., & Carandini, M. (2011). The Detection of Visual Contrast in the Behaving Mouse [Publisher: Society for Neuroscience]. *Journal of*

Neuroscience, *31*(31), 11351–11361. https://doi.org/10.1523/jneurosci.6689-10.2011

Chang, B., Hawes, N. L., Pardue, M. T., German, A. M., Hurd, R. E., Davisson, M. T., Nusinowitz, S., Rengarajan, K., Boyd, A. P., Sidney, S. S., Phillips, M. J., Stewart, R. E., Chaudhury, R., Nickerson, J. M., Heckenlively, J. R., & Boatright, J. H. (2007). Two mouse retinal degenerations caused by missense mutations in the $\upbeta$-subunit of rod cGMP phosphodiesterase gene [Publisher: Elsevier BV]. *Vision Research*, *47*(5), 624–633. https://doi.org/10.1016/j.visres.2006.11.020

Douglas, R. M., Alam, N. M., Silver, B. D., McGill, T. J., Tschetter, W. W., & Prusky, G. T. (2005). Independent visual threshold measurements in the two eyes of freely moving rats and mice using a virtual-reality optokinetic system [Publisher: Cambridge University Press (CUP)]. *Visual Neuroscience*, *22*(5), 677–684. https://doi.org/10.1017/s0952523805225166

Fischer, M. D., Huber, G., Beck, S. C., Tanimoto, N., Muehlfriedel, R., Fahl, E., Grimm, C., Wenzel, A., Remé, C. E., Pavert, S. A. v. d., Wijnholds, J., Pacal, M., Bremner, R., & Seeliger, M. W. (2009). Noninvasive, In Vivo Assessment of Mouse Retinal Structure Using Optical Coherence Tomography (H. D. Mansvelder, Ed.) [Publisher: Public Library of Science (PLoS)]. *PLoS ONE*, *4*(10), e7507. https://doi.org/10.1371/journal.pone.0007507

Gargini, C., Terzibasi, E., Mazzoni, F., & Strettoi, E. (2006). Retinal organization in the retinal degeneration 10 (rd10) mutant mouse: A morphological and ERG study [Publisher: Wiley]. *The Journal of Comparative Neurology*, *500*(2), 222–238. https://doi.org/10.1002/cne.21144

Hartong, D. T., Berson, E. L., & Dryja, T. P. (2006). Retinitis pigmentosa [Publisher: Elsevier BV]. *The Lancet*, *368*(9549), 1795–1809. https://doi.org/10.1016/s0140-6736(06)69740-7

Huber, G., Beck, S. C., Grimm, C., Sahaboglu-Tekgoz, A., Paquet-Durand, F., Wenzel, A., Humphries, P., Redmond, T. M., Seeliger, M. W., & Fischer, M. D. (2009). Spectral Domain Optical Coherence Tomography in Mouse Models of Retinal Degeneration [Publisher: Association for Research in Vision and Ophthalmology (ARVO)]. *Investigative Opthalmology & Visual Science*, *50*(12), 5888. https://doi.org/10.1167/iovs.09-3724

Kim, K. H., Puoris\textquotesinglehaag, M., Maguluri, G. N., Umino, Y., Cusato, K., Barlow, R. B., & Boer, J. F. d. (2008). Monitoring mouse retinal degeneration with high-resolution spectral-domain optical coherence tomography [Publisher: Association for Research in Vision and Ophthalmology (ARVO)]. *Journal of Vision*, *8*(1), 17. https://doi.org/10.1167/8.1.17

Lagali, P. S., Balya, D., Awatramani, G. B., Münch, T. A., Kim, D. S., Busskamp, V., Cepko, C. L., & Roska, B. (2008). Light-activated channels targeted to ON bipolar cells restore visual function in retinal degeneration [Publisher: Springer Science and Business Media LLC]. *Nature Neuroscience*, *11*(6), 667–675. https://doi.org/10.1038/nn.2117

Mitchiner, J. C., Pinto, L. H., & Vanable, J. W. (1976). Visually evoked eye movements in the mouse (Mus musculus) [Publisher: Elsevier BV]. *Vision Research*, *16*(10), 1169–IN7. https://doi.org/10.1016/0042-6989(76)90258-3

Pang, J.-j., Dai, X., Boye, S. E., Barone, I., Boye, S. L., Mao, S., Everhart, D., Dinculescu, A., Liu, L., Umino, Y., Lei, B., Chang, B., Barlow, R., Strettoi, E., & Hauswirth, W. W. (2011). Long-term Retinal Function and Structure Rescue Using Capsid Mutant AAV8 Vector in the rd10 Mouse, a Model of Recessive Retinitis Pigmentosa [Publisher: Elsevier BV]. *Molecular Therapy*, *19*(2), 234–242. https://doi.org/10.1038/mt.2010.273

Pinto, L. H., Invergo, B., Shimomura, K., Takahashi, J. S., & Troy, J. B. (2007). Interpretation of the mouse electroretinogram [Publisher: Springer Science and Business Media LLC]. *Documenta Ophthalmologica*, *115*(3), 127–136. https://doi.org/10.1007/s10633-007-9064-y

Porciatti, V., Pizzorusso, T., & Maffei, L. (1999). The visual physiology of the wild type mouse determined with pattern VEPs [Publisher: Elsevier BV]. *Vision Research*, *39*(18), 3071–3081. https://doi.org/10.1016/s0042-6989(99)00022-x

Prusky, G. T., Alam, N. M., Beekman, S., & Douglas, R. M. (2004). Rapid Quantification of Adult and Developing Mouse Spatial Vision Using a Virtual Optomotor System [Publisher: Association for Research in Vision and Ophthalmology (ARVO)]. *Investigative Opthalmology & Visual Science*, *45*(12), 4611. https://doi.org/10.1167/iovs.04-0541

Thomas, B. B., Shi, D., Khine, K., Kim, L. A., & Sadda, S. R. (2010). Modulatory influence of stimulus parameters on optokinetic head-tracking re-

sponse [Publisher: Elsevier BV]. *Neuroscience Letters*, *479*(2), 92–96. https://doi.org/10.1016/j.neulet.2010.05.031

Whishaw, I. (1995). A comparison of rats and mice in a swimming pool place task and matching to place task: Some surprising differences [Publisher: Elsevier BV]. *Physiology & Behavior*, *58*(4), 687–693. https://doi.org/10.1016/0031-9384(95)00110-5

Chapter 6

Discussion

The aim of this thesis is a better understanding of how mental models emerge from neural coding principles. Chapter 2 and 3 contain examples of models that were established by means of the sparse coding algorithm: first, a model for spatial structure from stereo image data and second, a model for egomotion from optical flow data. Here, I give a short summary on the sparse coding algorithm, including an outline on how simple rules can determine the development of a neural network that implements the sparse coding algorithm. This part extends the view on biological plausibility in chapter 2, section 2.4.4. Based on the two studies, I will also discuss which aspects of the representations qualifies them as emergent models. In chapter 4, a mental model, which can be classified as a topological map, accounts for navigation in cluttered environments. It is based on a dual population code that is close to a plausible neuronal model. I will discuss principles that may lead to the emergence of this model of spatial representation. Chapter 5 contains an automated method for the evaluation of visual acuity in mice. It allows for closed-loop testing of behavior. I will discuss the verification of hypotheses by quantifying behavior in the perception-action cycle.

6.1 Sparse coding establishes elementary sensory models

In this section, I first summarize the objective of the sparse coding algorithm. The summary serves as a basis to understand possible implementations in neuronal substrate. Next, I discuss how representations obtained with sparse coding

are related to mental models. In the final part, I discuss the sparse coding studies included in this thesis with respect to properties that qualify the examined representations as mental models.

6.1.1 The sparse coding algorithm

The sparse coding algorithm, first presented by B. A. Olshausen and Field (1996), encodes the input vector $\boldsymbol{x}$ into a sparse vector $\boldsymbol{a}$. Common examples apply data from sensor arrays modeled after the retina or the cochlear. The algorithm is an optimization that preserves information from the input, weighted against the sparsity of the distribution of values in $\boldsymbol{a}$. Sparsity means that only few values deviate from zero or, as a relaxation, that the values follow a heavy-tailed distribution. The optimization is performed by means of a gradient method that moves along the derivative of a two-termed cost function.

With the first term,

$$\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2 \,, \tag{6.1}$$

the optimization preserves information by minimizing the error between the input $\boldsymbol{x}$ and the reconstruction of the input $\hat{\boldsymbol{x}}(\boldsymbol{a}, \boldsymbol{\Phi})$. The reconstruction, i.e., the reverse mapping from $\boldsymbol{a}$ to $\boldsymbol{x}$, is linear, with

$$\hat{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{a} \,. \tag{6.2}$$

The matrix $\boldsymbol{\Phi}$ consists of k row vectors $\{\boldsymbol{\varphi}_k^\mathsf{T}\}$, which are called kernels or dictionary elements. Usually, the encoding is overcomplete, which means that the number of elements in $\boldsymbol{a}$ exceeds the number of elements in the input $\boldsymbol{x}$. Therefore, the vector $\boldsymbol{a}$ that maps to $\boldsymbol{x}$ is not unique. The second term is the sparsity penalty

$$S(\boldsymbol{a}) \,. \tag{6.3}$$

In the original formulation, the relaxed sparsity penalty (B. A. Olshausen & Field, 1996) is the ℓ^1-norm of the vector $\boldsymbol{a}$. As an alternative, the locally competitive algorithm (LCA) optimizes for an approximation of the number of non-zero elements in $\boldsymbol{a}$, which is sometimes called the ℓ^0-norm (Rozell et al., 2008). The gradient descent is then performed on the negative partial derivatives of the elements

in $\boldsymbol{a}$ and $\boldsymbol{\Phi}$ of the full cost function

$$E \;=\; \|\boldsymbol{x} - \hat{\boldsymbol{x}}(\boldsymbol{a}, \boldsymbol{\Phi})\|^2 \;+\; S(\boldsymbol{a})\,. \tag{6.4}$$

6.1.2 Sparse coding in neuronal substrate

It has early been recognized that the mathematical term of the gradient descent on the elements $\{a_k\}$ in $\boldsymbol{a}$ can be translated to neural network notion (B. A. Olshausen & Field, 1997). In mathematical terms, the optimization rule is

$$\dot{a}_k \;\propto\; \boldsymbol{\varphi}_k^{\mathsf{T}}\boldsymbol{x} - \sum_{c \neq k} \boldsymbol{\varphi}_k^{\mathsf{T}}\boldsymbol{\varphi}_c a_c - a_k\,. \tag{6.5}$$

If we assume a neural network, in which the firing rates of neurons of an ensemble are represented by the values $\{a_k\}$, the three terms represent the rules for the connectivity of the network. The first term is a feed forward term that weighs the sensory input with the kernels $\{\boldsymbol{\varphi}_k\}$. The kernels therefore serve as the receptive fields of the neuron. The second term is a term of mutual competition between two neurons. This lateral inhibition is proportional to the similarity between the associated kernels. The last term induces self-inhibition that drives the neuron towards zero in the absence of input.

While connectivity can be inferred straight forward from the mathematical terms, learning and information sharing requires global information. First, learning of the values in $\boldsymbol{\Phi}$ requires knowledge of the reconstruction error. Only the first term of the cost function depends on $\boldsymbol{\Phi}$. The learning rule therefore simply improves the error of the reconstruction with

$$\Delta\varphi_{k,i} \;\propto\; a_k(x_i - \hat{x}_i)\,, \tag{6.6}$$

where $\{\varphi_{k,i}\}$ are the elements of $\boldsymbol{\Phi}$, $\{x_i\}$ are the elements of $\boldsymbol{x}$ and $\{\hat{x}_i\}$ are the elements of $\hat{\boldsymbol{x}}$. Second, the feed forward term and the lateral inhibition term in Eq. 6.5 share the information in $\boldsymbol{\Phi}$. In addition to information sharing, the values of every kernel need to be compared against the values of any other kernel in order to calculate the amount of lateral inhibition. In conclusion, the network architecture that encodes sensory input into a sparse representation can be inferred directly from the algorithm, whereas it is unclear how to realize learning of the

weights directly from in a biologically plausible way.

However, it was shown that the weights of such a network can be learned by applying local rules. Földiák (1990) studied a network with similar structure as implied by Eq. 6.5, however with learned feed forward weights $q_{i,k}$ and with learned weights for lateral inhibition $w_{c,k}$. The dynamics and the connectivity of the network followed

$$\dot{a}'_k = f\left(\sum_i q_{i,k}x_i + \sum_c w_{c,k}a'_c - t_k\right) - a'_k\,. \tag{6.7}$$

The formulation differed from the sparse coding algorithm by two additional components: the nonlinearity $f(u) = 1/(1+\exp(-\lambda u))$ and the thresholds $\{t_k\}$. Without the nonlinearity, the network did not learn higher-order dependencies between features, but found the k largest eigenvectors of the covariance matrix (Földiák, 1989). The thresholds were used to regularize activity of the neurons and were therefore also subject to learning. They took effect each time the network had settled, by setting $a_k = 1$ if $a'_k > 0.5$ and $a_k = 0$ otherwise.

The weights were then learned by means of simple, local Hebbian rules. The feed forward weights were learned by applying

$$\Delta q_{i,k} \propto a_k(x_i - q_{i,k}) \tag{6.8}$$

where $-q_{i,k}$ is an additional term that induces weight decay. Learning of the competitive weights was subject to anti-Hebbian learning, which means that the weights of *inhibitory* connections grow stronger, the more often two units are active at the same time. The weights were therefore updated by applying

$$\Delta w_{c,k} \propto -(a_c a_k - p^2)\,. \tag{6.9}$$

Values were cropped to 0 if $w_{c,k} > 0$ and $w_{k,k}$ was set to the constant value 0. p was the specified bit probability to which thresholds were adjusted with

$$\Delta t_k \propto a_k - p\,. \tag{6.10}$$

By learning images that were composed by the superposition of simple patterns—stripes of one pixel width and letters—Földiák (1990) showed that this

network learned kernels that recovered the original patterns. Falconbridge et al. (2006) showed that the network can learn Gabor-like kernels from natural image data. Interestingly, a biologically more plausible activation function resulted in a better fit to V1 receptive fields. Zylberberg et al. (2011) translated the network to a spiking neural network. The Hebbian learning was rate based, by counting spikes within a time window. They found that, similar to the sparse coding algorithm, the input to the network can be recovered by linear reconstruction. They also showed that the network accounts for the diverse shapes of V1 simple cell receptive fields.

These studies are conclusive empirical evidence that the learning of a sparse representation is possible in biological substrate by means of simple Hebbian and anti-Hebbian learning. They contribute to our understanding of sparse coding as the underlying principle for sensory data processing in V1. Additional considerations apply to the biological system. For example, inhibitory connections are usually realized with inhibitory interneurons. King et al. (2013) successfully extended the spiking neural network from Zylberberg et al. (2011) with inhibitory interneurons. They related their network to physiological findings from Haider et al. (2010), who showed that interneurons increase sparsity in classical and non-classical receptive fields. Furthermore, the discussed principles of network structure and learning align well with known developmental principles of the striate cortex. The development of the general retinotopic layout is guided by molecular gradients (McLaughlin & O'Leary, 2005). This mapping could enable a simple mechanism to limit the amount of synaptic connections to a reasonable local extend, both for the feed forward connections and for the connections of lateral inhibition. The Hebbian and anti-Hebbian learning rules could then refine the synaptic strength of the existing connections.

6.1.3 Mental models in sparse representations

Categorization by competition between similar concepts

The core mechanism for the formation of a sparse representation is the competition between units, mediated by the simple principle of lateral inhibition. Lateral inhibition indeed is very common in neuronal processing. The most prominent examples are early sensory networks, like in the retina or in the tactile sensory system. However, the same principle applies to the processing of abstract concepts.

In this case, inhibition is not topological but proportional to similarity and overlap of the concepts. For example, it helps to avoid the confusion of similar semantic concepts, like "astronomy" and "astrology" (Baars and Gage (2010), p70). The sparse coding algorithm inhibits units that are lateral in this sense.

Barlow envisioned that the competition between units that respond to similar patterns is part of the mechanism that makes the cerebral cortex a model builder (H. Barlow, 1987). He sketched out a three-step outline of a developmental process as follows. First, cells receive excitatory input from many afferent fibers, which is the basis for processing suspicious coincidences by summation and thresholding[1]. However, integration of the input over many neurons has a blurring effect and many neurons will share information from the sensory input. Therefore, second, a system of mutual inhibition causes the input to be classified into mutually exclusive outputs. This processing step makes the differentiation between categories with overlapping inputs possible. Third, a Hebb-like mechanism strengthens connections that have contributed to the activation of the neuron. With this mechanism, neurons will gain selectivity to compound events—or suspicious coincidences—that occur frequently. Barlow proposed that these nerve cells serve as incidence detectors and therefore represent a model of the associative structure in its inputs.

Barlow's outline is reminiscent to the realization of the sparse coding algorithm in neuronal substrate, described in the previous subsection. Note that many unsupervised learning algorithms rely on similar mechanisms of lateral inhibition. Examples include the Hopkins network, which serves as a model for human associative memory, and the Kohonen network, which explains how a stimulus continuum can be encoded into a self organizing, topological map (Hopfield, 1982; Kohonen, 1982).

Building blocks of a simplified world model

The sparse coding algorithm belongs to the class of independent component analysis (ICA) algorithms (Simoncelli & Olshausen, 2001). ICA assumes a multidimensional mixed signal, in which each dimension is the linear superposition of independent signals. The aim is the reconstruction of the original signals by guessing the inverse of the matrix that specifies the superposition. The algorithms' foun-

[1] for the concept of suspicious coincidences and their relation to mental models, see the introduction.

dation is the central limit theorem, which states that the distribution of summed independent source signals tends towards a Gaussian distribution. Therefore, under the condition that the distribution of the source signals is not Gaussian, the non-Gaussian distribution is a strong footprint in the mixed signal. The source signals can be retrieved by finding directions in multi-dimensional signal space in which the distribution deviates most from a Gaussian distribution. These directions are the eigenvectors of the matrix that specifies the superposition of the source signals (Hyvärinen & Oja, 2000).

The goal of sparse coding is to find directions in sensory space that are zero most of the time. An approximation of this constraint is to find directions, in which the distribution is heavy-tailed, i.e., in which the distribution has a large kurtosis. These target-distributions differ strongly from a Gaussian distribution, so that optimization for a sparse representation is at the same time an optimization that retrieves independent source signals (B. A. Olshausen & Field, 1997). One could naïvely conclude that the transformation recovers the original source signals from any given sensory space. This would come as a surprise, given the complexity of the physical world, which gives rise to the sensory input. Indeed, in the case of visual data from natural scenes, it was shown that the most independent directions in sensory space are still interdependent to a large extend (Bethge, 2006; Eichhorn et al., 2009).

Applying the sparse coding algorithm to sensory data results in a representation that is a rather crude model of its sources, compared to the actual complexity of the sensory world. However, representations obtained with sparse coding contain features that are related to physical causes. These features can serve as the building blocks of a simple and heuristic mental model of the sensory world. This thesis contains two chapters which report the relation between features obtained with sparse coding and their physical causes. First, a model for spatial structure from stereo image data and second, a model for egomotion from optical flow data. In the following, I will discuss the aspects of these representations that qualify them as mental models.

6.1.4 Examples of sparse representations that qualify as mental models

Sparse coding of stero images

Chapter 2 contains a study where we applied sparse coding to data from natural stereo images with vergence. We used the convolutional locally competitive algorithm (LCA) with five levels of overcompleteness: 0.6, 1, 3, 8 and 16 times the size of the input. The kernel shapes were an overall good match for physiological receptive fields of cells in the visual cortex, as already described in detail by others (Hunter & Hibbard, 2015; Hyvärinen & Oja, 2000; Ringach, 2002). Without exception, units of these representations showed smoothly varying and clear selectivity to stereo disparity. With a simple, probabilistic readout of the population code it was possible to estimate disparity with low error within a range of ±6 px. The units of the representations were selective to at least one additional element of spatial layout: the orientation of surfaces with respect to the observer. Our study resulted in two additional outcomes: first, we found that the inference error decreased with decreasing sparsity load. Second, we found that the inference error can be estimated by counting the number of units that contribute to the reconstruction of an individual image. The error is linked to the number of active units by a u-shaped function. We have proposed that both relations could be exploited for an attention mechanism which controls for the trade off between energy efficiency through increased sparsity and accuracy of the inference. Attention is an important mechanism in the perception-action cycle that mediates top-down causation between levels of the living system hierarchy (Tsotsos & Rothenstein, 2011). In neural network notion of the LCA, the sparsity load is controlled by the threshold of neurons. Therefore, by dynamically adjusting the thresholds within a local extend, the network could control for the inference error.

The sparse coding algorithm is a unsupervised learning method, which makes statistic features of the input data explicit (B. Olshausen & Field, 2004). Applied to visual stereo data, the extracted features are related to their physical cause. They serve as representatives of the spatial layout of the scene. From disparity and the known distance of the two stereo cameras it is possible to calculate the distance to structures within the field of view, and surface orientation is a direct feature of spatial layout. We expect that these features are selective to addi-

tional elements of scenes, like for example occlusion boundaries or the curvature of surfaces. The selectivity of features obtained with sparse coding can be a guide towards simple categories that describe the layout of the scene. It seems like the first steps towards more abstract concepts, like the partitioning of the scene into objects by means of object boundaries or like categorization of objects or places by means of their shape. Such concepts, associated with causal relationships like the Gestalt principles, can serve as simple world models that allows an agent to make inferences about the world. As already described in the beginning of this section, the hierarchical formation of this kind of models on the level of the neuronal code might be supported by mechanisms similar to the ones that induce sparsity: association with feed-forward hebbian learning and differentiation with anti-hebbian learning of lateral inhibition. Obviously, the discussed mechanisms are not sufficient to explain how cognitive model building emerges from simple principles of interaction. However, they are sufficient as a first step to build a representation of visual input in V1 and they may play a role in later stages of neuronal processing.

Sparse coding of optic flow

Chapter 3 contains a study where we applied sparse coding to optic flow data from egomotion. Properties of this study were designed to match the visual system of zebrafish larvae and results were compared against data from physiological experiments. The two eyes of the fish covered large, circular visual fields of about 160° each, with 45° overlap to the front, so that most of the visual surrounding was covered. The fish retina extracts optic flow information. We therefore rendered images from the movement of virtual fish in a fish tank and extracted optic flow fields from two consecutive images. We used the locally competitive algorithm (LCA) for learning a sparse representation of optic flow.

Kubo et al. (2014) showed that the majority of pretectal neurons of zebrafish respond to large, homogeneous visual motion fields. A substantial portion of these neurons is selective to flow fields that originate from forward or backward motion or from clockwise or counterclockwise motion of the fish. With our setup of LCA sparse coding without whitening, we were able to partially reproduce the statistical distribution of responses. In addition, individual units of the sparse representation showed clear and smoothly varying selectivity to directions of translation and rotation. In some cases, units were only selective to either translation or rotation,

in other cases they were selective to a combination of both.

To consider the neuronal processing of zebrafish larvae in the context of mental models might seem odd at first sight. However, with the LCA algorithm, the same principles for the formation of a neuronal code were applied as in the previous study to visual stereo data. Directions of translation and rotation are representatives of a physical cause, the ego-motion of the fish in its environment. Interestingly, the selectivity for ego-motion related flow fields was learned as a second step after first extracting optic flow. It can be expected that a representation obtained by applying sparse coding to the time series of images is selective to optic flow, since the problem of detecting optic flow is similar to the stereo problem. Indeed, it was shown that, in this case, the shapes of the spatiotemporal kernels are often Gabor-like and translated over time (B. A. Olshausen, 2003). Therefore, it might be worthwhile to experiment with a completely unsupervised, hierarchical processing pipeline that starts with the time series of images and ends with ego-motion selectivity to further explore how unsupervised hierarchical model building could emerge from simple principles of neuronal interaction.

6.2 An evolutionary plausible navigation hierarchy accounts for the constitution of a mental map

Chapter 4 contains a study of a biologically inspired navigation scheme with dual population coding. Many biological entities have the ability to navigate in their environment, and neuronal correlates like place cells and the associated navigational models reflect complex cognitive processing capacities. If we want to understand how complex behavior like the navigation skills of animals emerges, it is instructive to follow the path of evolution. Navigational mechanisms can be classified into methods of various complexity, where each step requires new skills on top of the lower level skills (Franz & Mallot, 2000). They can be classified into local navigation and wayfinding. Local navigation includes *search, direction following, aiming*, and *guidance*, which all rely on cues available as sensory information at navigation time. Wayfinding includes *recognition-triggered response, topological-*, and *survey navigation*, which are characterized by connecting more than one place with mech-

anisms on top of local methods. In the following, I discuss these navigational skills, closely following (Franz & Mallot, 2000), with a focus on the underlying neuronal principles. I then discuss how dual population coding is positioned within the navigational hierarchy.

6.2.1 Local navigation

Of all local navigation principles, the simplest one is search, where an agent moves by chance until it detects its goal. The crucial cognitive competence for search is the detection of the goal, which can also be very simple in some cases. For example, a food source can be sensed by means of a distinctive color and a place of shelter can be marked with a specific, detectable molecule. However, if the goal is a unique place, it is often necessary to detect more complex patterns. One example for such remarkable recognition capacities comes from the first experiments on place recognition in scientific literature. Tinbergen and Kruyt (1938) described the capabilities of a wasp, the european beewolf (*Philanthus triangulum*). It recognizes the small entry to its nest by means of the surrounding patterns. The robust place detection necessary for such a task relies on the detection of distinctive and unique cues. Pattern detection is a large field of its own, with many examples of various levels of sophistication over the animal kingdom. In the human cortex, the inferior temporal gyrus (IT) contains cells with a strong degree of selectivity to abstract patterns. The human object recognition performance can be predicted from a weighted sums model of IT cell activity (Majaj et al., 2015). Interestingly, the Dentate Gyrus performs pattern separation by means of competitive learning with sparse coding (Hanuschkin et al., 2018). In conjunction with grid-cells, these patterns contribute to the firing of place-cells in the hippocampus (Rolls, 2013). For an outline of how pattern selectivity of this kind may emerge from simple neuronal principles, please refer to the introduction of this thesis and to section 6.1.3.

Direction following refers to an agent aligning its movement to a locally available directional cue. Two examples for such cues are the polarization patterns in the sky, exploited for navigation by the desert ant *Cataglyphis fortis* (Müller & Wehner, 2007), and the geographical slant, which improves navigational performance in humans (Restat et al., 2004). In the case of global directional cues, direction following requires path integration in order to estimate arrival at the goal

location. For example, in the case of *Cataglyphis fortis*, path integration of a homing vector is realized by a combination of counting steps, weighted by the direction of movement (M. Muller & Wehner, 1988). Direction following is closely related to *aiming*, where the agent recognizes a beacon and moves towards it. As before, it is not always necessary to use sophisticated mechanisms of pattern detections to recognize a goal. A salient beacon could be a simple chemical source, which allows to navigate by means of chemotaxis, or a source of light, which allows to navigate by means of phototaxis. In many cases, direction following and aiming may be realized in a surprisingly simple manner. This insight is most evident in the impressively simple connectivity models of Braitenberg-vehicles (Braitenberg, 1986). Very elementary relationships between sensors and actuators, which may evolve by combinatorial try and error, are sufficient to reproduce the behavior to navigate towards a beacon, to follow a gradient, or to follow a trail.

Using the spatial configuration of landmarks to navigate towards a goal is called *guidance*. It requires complex computational steps, including the detection of landmarks and a control mechanism to move into a direction that gradually improves the perceived spatial relation between the landmarks. Cartwright and Collett (1983) presented empirical evidence that honey bees (*Apis mellifera*) searched for food using guidance. The search location was best predicted by the matching angular relative position of landmarks on an assumed 360° retina. In order to detect lanmdmarks, a matching snapshot template might be sufficient in simple cases, like in the honey bee. An example for an efficient cue is the profile of the skyline, which is easy to detect due to the high contrast (Basten & Mallot, 2010). Additionally, the algorithm must have access to the distances between the landmarks on the retina. From these assumptions, a simple control algorithm was proposed by Cartwright and Collett, which was able to reproduce the homing behavior of bees (Cartwright & Collett, 1987b). Möller et al. (1999) presented a very simple, and only local, network which performed similarly well. How such a network could emerge from simple neuronal principles is an interesting topic for future research.

6.2.2 Wayfinding

Of all wayfinding navigation principles, the simplest is *recognition-triggered response*. It is based on local navigation towards a goal location. However, as an additional element, the recognition of a starting position triggers the locomotion.

With the association of starting positions with movement instructions, it is possible to chain up several waypoints and therefore follow a route. As soon as local navigation schemes exist, this associative scheme seems to be the natural next evolutionary step. Association is the most prominent feature of neural circuitry formation, ubiquitous throughout synaptically local learning algorithms. Indeed, the transition of local navigation towards navigation with recognition-triggered response is fluent. Local methods need to be evoked or suppressed, depending on the behavioral goal of the animal. In simple cases, these triggers may be the need to forage, due to a deficit in nutrients, or the need to return to shelter, due to weather events or the recognition of a natural enemy. From this, it is only a small step to associate cues that represent one place with local navigation behavior towards another place.

Topological navigation is closely related to the recognition-triggered response scheme. Likewise, places are linked to each other by methods of local navigation. However, these links are independent of the goal, by only representing transition from one place to the other. This extension renders it possible that two routes to independent goals pass through the same place. In order to make the crossing of routes possible, moving instructions to more than one place have to be associated to one place. At navigation time, the selection of one of the instructions depends on the goal. Places are therefore not chained up, but embedded into a graph, called a topological map. In this graph, each node represents a place and each edge contains the instruction for an action that leads from one place to the other. This representation serves as a mental model or a cognitive map of the environment.

In order to make use of the representation, planing capabilities are required. an agent can calculate the path to arbitrary goal positions by means of graph search from one place to another. On the neuronal level, places may be represented as neurons, or they may be represented as a set of feature neurons and transitions from one place to another may be represented by axons and synapses between these neurons(H. Mallot et al., 1995; Scholkopf & Mallot, 1995). The planning of routes could be realized by mental exploration. Ponulak and Hopfield (2013) presented such a neuronal graph search algorithm. In this model, activation is induced into the network at the goal node, propagating a wave of activity through the network in parallel. Using spike time dependent plasticity, the wave imprints a trace of its first path through the network, which corresponds to the shortest

path. A readout mechanism is required to retrieve the whole trace.

Survey navigation denotes the embedding of places into a common frame of reference. It differs largely from topological navigation by the need to be accessible as a whole, as opposed to the isolated spatial relationship between pairs of places. It enables an agent to find shortcuts through novel terrain or detours around obstacles. Survey navigation has only limited prevalence in biological systems (Franz & Mallot, 2000), but is predominant in robot navigation. State of the art robot navigation relies on simultaneous localization and mapping (SLAM), which is a global optimization of the metric embedding of landmarks and the agent in a common frame of reference (Fuentes-Pacheco et al., 2015). In the evolutionary path, survey navigation can be thought to work top of topological navigation. The metric embedding of a view graph allows for transition between topological and survey navigation (Hübner & Mallot, 2007b). Indeed, grid cell like metric representations, used for the integration of odometry information, may help to improve survey competence in the case of loop closures (Banino et al., 2018; Milford et al., 2007).

6.2.3 Dual population coding in the navigation hierarchy

We presented a new, biologically motivated navigation method, which we coined dual population coding. It can be classified as a relaxed topological navigation scheme. We assume a representation based on numerous overlapping and ambiguous place fields, similar to the fields represented by place cells in rats (O'Keefe & Speakman, 1987). The movement instructions are not associated to places, but to place fields. A route from one place to another is linked to a number of movement instructions from all place fields that intersect with the current location to all place fields that intersect with the goal. The decision to move in a certain direction is solved by a voting scheme over all movement instructions towards the goal that are linked to the place fields at the current location. The term "dual population coding" originates from the fact that both, place and movement instruction, are not discrete in space, but associated to several overlapping place fields.

Our model greatly simplifies the need to detect patterns associated to places. Each node represents a pattern which is not necessarily unique for a place. Only a set of features distinctively describes places or objects. In our model, we used SURF features because a set of SURF or SIFT features fulfills this requirement

(Bay et al., 2008b; Lowe, 1999; Sivic & Zisserman, 2003b). However, any mechanism with similar or better spatial selectivity, like for example selectivity in IT cortex, would be sufficient. With this diffuse representation of places, the distinction between local navigation and wayfinding is vague. Close to the goal, navigation is local, in the sense that simple stimulus-response pairs are sufficient for homing. As described earlier, it is not unlikely that this kind of association between recognized stimuli and a locomotion response has emerged in evolution.

Connections between adjacent place fields may also be explained based on association of place fields firing at the same time. However, the association of places is not sufficient. The transition between two place fields must be associated with an appropriate movement instruction. Therefore, a mechanism for detecting the time in point at which transition occurs is required. With neurons, the detection of changes in firing rate can be realized with adaptation mechanisms and learning that depends on timing can be realized with spike time dependent plasticity.

In addition, our method relies on path planning. The mental exploration of routes is a graph search task, for which we used the Dijkstra algorithm. The associative network of places sketched out in the paragraphs before can be used for this purpose. The parallel nature of neural networks is an excellent fit for this task (Ponulak & Hopfield, 2013). Inducing activity in the goal node spreads through the network in waves. In our model, it is sufficient to detect the node from which the wave first hits the start node. The agent could navigate to the corresponding place close by and iterate the procedure from place to place until it reaches the corresponding location.

With our study, the aim was to develop a minimal model, sufficient for wayfinding in a naturalistic environment. It is derived from a stimulus-response scheme, requires only a small amount of visual invariance for localization, uses a simple decision process in path planning, and does not rely on metric information. The embedding into the navigational hierarchy sketches out a course evolutionary path on which spatial cognition may have emerged in succession. However, we did not develop a detailed evolutionary path that may explain the emergence of wayfinding. Moreover, we did not transform the model to an implementation with neural networks. But our model can help to uncover the path of emergent evolution, opening the door for many interesting future research questions. It offers a minimal, yet efficient and evolutionary plausible framework for wayfinding and it is

a promising starting point for more complex and sophisticated models of spatial cognition.

6.3 Testing entities of living systems in the perception-action cycle

The perception-action cycle was described by J. Fuster as "the circular flow of information from the environment to sensory structures, to motor structures, back again to the environment, to sensory structures, and so on, during the process of goal-directed behavior". The hierarchically organized brain areas communicate with each other, with internal and external feedback at every level, which makes them circular. Early levels of the hierarchy in sensory areas process simple stimulus responses, whereas higher levels, like association and prefrontal cortex control more complex behavior (Fuster, 2004). Many cognitive phenomena are subject to the perception-action cycle, like for example perception, attention, cognitive control, decision making, conflict resolution and monitoring, knowledge representation and reasoning, learning and memory, planning and action, and consciousness (Cutsuridis et al., 2011).

If we understand cognition as an embodied system, enmeshed at every level with a larger ecosystem, it follows that we must test hypotheses and models in a naturalistic feedback loop. Our experimental setup from chapter 5 is an example of testing animals in such a behavioral feedback loop. It was designed to evoke and characterize the optokinetic reflex (OKR) of mice in order to infer properties of the visual system. The setup is now commercially available as "OptoDrum"[2]. It is primarily used to quantify the effect of pharmaceutical treatments on the visual system.

In some animals, or if eye movement is measured directly, the reflex is triggered reliably and constantly, as soon as the stimulus is presented, so that automated tracking and evaluation is feasible. In zebrafish for example, methods for tracking the OKR are well established and software is available as open source (Scheetz et al., 2018). However, mice have a rich repertoire of high level behavior, which often superimposes on the low level OKR. They groom, explore or try to escape from the platform. Instead of tracking eye movement, the aim was no-invasive tracking

[2]https://stria.tech/optodrum

of head movement of freely behaving mice. Therefore, isolation and quantification of the OKR required elaborate processing. Assessing the tracking behavior of the animals included the detection of the animal sitting still and an analysis of the head movement fitting the stimulus speed and direction.

The goal of our setup was to isolate a low level reflex. However, hypotheses on cognitive models need to recognize the embodied and integrated character of biological systems. Similar setups with visual feedback have been used to investigate any level of neuronal processing, like early sensory processing of optic flow in zebrafish (Wang et al., 2019), the influence of visual cues on navigational strategies employed by freely moving ants (Murray et al., 2020) or rats (Hölscher et al., 2005), or how language cues influence the hierarchical representation of space (Schick et al., 2019). Such setups enable rich options for stimulus manipulation and can generate stimuli which are impossible in real world experiences. Manipulited stimuli can serve as perturbances, engineered to test hypotheses on cognitive models. Especially in conjunction with neuroimaging methods, testing animals in the perception-action cycle has the potential to reveal more about the relationship between cognitive models and the neuronal principles from which they emerge.

References

Baars, B. J., & Gage, N. M. (2010). *Cognition, brain, and consciousness: Introduction to cognitive neuroscience* (2nd ed) [OCLC: ocn455870625]. Academic Press/Elsevier.

Banino, A., Barry, C., Uria, B., Blundell, C., Lillicrap, T., Mirowski, P., Pritzel, A., Chadwick, M. J., Degris, T., Modayil, J., Wayne, G., Soyer, H., Viola, F., Zhang, B., Goroshin, R., Rabinowitz, N., Pascanu, R., Beattie, C., Petersen, S., … Kumaran, D. (2018). Vector-based navigation using grid-like representations in artificial agents. *Nature*. https://doi.org/10.1038/s41586-018-0102-6

Barlow, H. (1987). Cerebral Cortex as Model Builder [Reporter: Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience]. In L. M. Vaina (Ed.), *Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience* (pp. 395–406). Springer Netherlands. https://doi.org/10.1007/978-94-009-3833-5_18

Basten, K., & Mallot, H. A. (2010). Simulated visual homing in desert ant natural environments: Efficiency of skyline cues. *Biological Cybernetics*, *102*(5), 413–425. https://doi.org/10.1007/s00422-010-0375-9

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008b). Speeded-up robust features (SURF). *Computer vision and image understanding*, *110*(3), 346–359. https://doi.org/10.1016/j.cviu.2007.09.014

Bethge, M. (2006). Factorial coding of natural images: How effective are linear models in removing higher-order dependencies? [Publisher: The Optical Society]. *Journal of the Optical Society of America A*, *23*(6), 1253. https://doi.org/10.1364/josaa.23.001253

Braitenberg, V. (1986). *Vehicles: Experiments in synthetic psychology*. MIT press.

Cartwright, B. A., & Collett, T. S. (1987b). Landmark maps for honeybees. *Biological cybernetics*, *57*(1-2), 85–93. https://doi.org/10.1007/BF00318718

Cartwright, B. A., & Collett, T. S. (1983). Landmark learning in bees. *Journal of Comparative Physiology*, *151*(4), 521–543. https://doi.org/10.1007/BF00605469

Cutsuridis, V., Hussain, A., & Taylor, J. G. (Eds.). (2011). *Perception-Action Cycle*. Springer New York. https://doi.org/10.1007/978-1-4419-1452-1

Eichhorn, J., Sinz, F., & Bethge, M. (2009). Natural Image Coding in V1: How Much Use Is Orientation Selectivity? (L. Zhaoping, Ed.) [Number: 4 Reporter: PLoS Computational Biology]. *PLoS Computational Biology*, *5*(4), e1000336. https://doi.org/10.1371/journal.pcbi.1000336

Falconbridge, M. S., Stamps, R. L., & Badcock, D. R. (2006). A Simple Hebbian / Anti-Hebbian Network Learns the Sparse, Independent Components of Natural Images [Publisher: MIT Press - Journals]. *Neural Computation*, *18*(2), 415–429. https://doi.org/10.1162/089976606775093891

Földiák, P. (1990). Forming sparse representations by local anti-Hebbian learning [Publisher: Springer Science and Business Media LLC]. *Biological Cybernetics*, *64*(2), 165–170. https://doi.org/10.1007/bf02331346

Földiák, P. (1989). Adaptive network for optimal linear feature extraction [Publisher: Citeseer].

Franz, M. O., & Mallot, H. A. (2000). Biomimetic robot navigation. *Robotics and autonomous Systems*, *30*(1), 133–153. https://doi.org/10.1016/S0921-8890(99)00069-X

Fuentes-Pacheco, J., Ruiz-Ascencio, J., & Rendón-Mancha, J. M. (2015). Visual simultaneous localization and mapping: A survey. *Artificial Intelligence Review*, *43*(1), 55–81. https://doi.org/10.1007/s10462-012-9365-8

Fuster, J. M. (2004). Upper processing stages of the perception–action cycle. *Trends in Cognitive Sciences*, *8*(4), 143–145. https://doi.org/10.1016/j.tics.2004.02.004

Haider, B., Krause, M. R., Duque, A., Yu, Y., Touryan, J., Mazer, J. A., & McCormick, D. A. (2010). Synaptic and Network Mechanisms of Sparse and Reliable Visual Cortical Activity during Nonclassical Receptive Field Stimulation. *Neuron*, *65*(1), 107–121. https://doi.org/10.1016/j.neuron.2009.12.005

Hanuschkin, A., Yim, M. Y., & Wolfart, J. (2018). A Network Model Reveals That the Experimentally Observed Switch of the Granule Cell Phenotype During Epilepsy Can Maintain the Pattern Separation Function of the Dentate Gyrus. *Springer Series in Computational Neuroscience* (pp. 779–805). Springer International Publishing. https://doi.org/10.1007/978-3-319-99103-0_23

Hölscher, C., Schnee, A., Dahmen, H., Setia, L., & Mallot, H. A. (2005). Rats are able to navigate in virtual environments. *Journal of Experimental Biology*, *208*(3), 561–569. https://doi.org/10.1242/jeb.01371

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. [Publisher: Proceedings of the National Academy of Sciences]. *Proceedings of the National Academy of Sciences*, *79*(8), 2554–2558. https://doi.org/10.1073/pnas.79.8.2554

Hübner, W., & Mallot, H. A. (2007b). Metric embedding of view-graphs: A vision and odometry-based approach to cognitive mapping. *Autonomous Robots*, *23*(3), 183–196. https://doi.org/10.1007/s10514-007-9040-0

Hunter, D. W., & Hibbard, P. B. (2015). Distribution of independent components of binocular natural images [Number: 13 Reporter: Journal of Vision]. *Journal of Vision*, *15*(13), 6. https://doi.org/10.1167/15.13.6

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications [Number: 4 Reporter: Neural networks]. *Neural networks*, *13*(4), 411–430. https://doi.org/10.1016/S0893-6080(00)00026-5

King, P. D., Zylberberg, J., & DeWeese, M. R. (2013). Inhibitory Interneurons Decorrelate Excitatory Cells to Drive Sparse Code Formation in a Spiking Model of V1 [Number: 13 Reporter: Journal of Neuroscience]. *Journal of Neuroscience*, *33*(13), 5475–5485. https://doi.org/10.1523/JNEUROSCI.4188-12.2013

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, *43*(1), 59–69. https://doi.org/10.1007/BF00337288

Kubo, F., Hablitzel, B., Dal Maschio, M., Driever, W., Baier, H., & Arrenberg, A. B. (2014). Functional Architecture of an Optic Flow-Responsive Area that Drives Horizontal Eye Movements in Zebrafish. *Neuron*, *81*(6), 1344–1359. https://doi.org/10.1016/j.neuron.2014.02.043

Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, *2*, 1150–1157. https://doi.org/10.1109/ICCV.1999.790410

Majaj, N. J., Hong, H., Solomon, E. A., & DiCarlo, J. J. (2015). Simple Learned Weighted Sums of Inferior Temporal Neuronal Firing Rates Accurately Predict Human Core Object Recognition Performance. *Journal of Neuroscience*, *35*(39), 13402–13418. https://doi.org/10.1523/JNEUROSCI.5181-14.2015

Mallot, H., Bülthoff, H., Georg, P., Schölkopf, B., & Yasuhara, K. (1995). View-based cognitive map learning by an autonomous robot. *Proceedings of ICANN*, *95*, 381–386.

McLaughlin, T., & O'Leary, D. D. (2005). Molecular Gradients and Development of Retinotopic Maps. *Annual Review of Neuroscience*, *28*(1), 327–355. https://doi.org/10.1146/annurev.neuro.28.061604.135714

Milford, M., Schulz, R., Prasser, D., Wyeth, G., & Wiles, J. (2007). Learning spatial concepts from RatSLAM representations. *Robotics and Autonomous Systems*, *55*(5), 403–410. https://doi.org/10.1016/j.robot.2006.12.006

Möller, R., Maris, M., & Lambrinos, D. (1999). A neural model of landmark navigation in insects. *Neurocomputing*, *26-27*, 801–808. https://doi.org/10.1016/S0925-2312(98)00150-7

Muller, M., & Wehner, R. (1988). Path integration in desert ants, Cataglyphis fortis [Publisher: Proceedings of the National Academy of Sciences]. *Proceedings*

of the National Academy of Sciences, *85*(14), 5287–5290. https://doi.org/10.1073/pnas.85.14.5287

Müller, M., & Wehner, R. (2007). Wind and sky as compass cues in desert ant navigation. *Naturwissenschaften*, *94*(7), 589–594. https://doi.org/10.1007/s00114-007-0232-4

Murray, T., Kócsi, Z., Dahmen, H., Narendra, A., Le Möel, F., Wystrach, A., & Zeil, J. (2020). The role of attractive and repellent scene memories in ant homing (*Myrmecia croslandi*). *The Journal of Experimental Biology*, *223*(3), jeb210021. https://doi.org/10.1242/jeb.210021

O'Keefe, J., & Speakman, A. 1. (1987). Single unit activity in the rat hippocampus during a spatial memory task. *Experimental brain research*, *68*(1), 1–27. https://doi.org/10.1007/BF00255230

Olshausen, B., & Field, D. (2004). Sparse coding of sensory inputs [Number: 4 Reporter: Current Opinion in Neurobiology]. *Current Opinion in Neurobiology*, *14*(4), 481–487. https://doi.org/10.1016/j.conb.2004.07.007

Olshausen, B. A. (2003). Principles of Image Representation in Visual Cortex. *The visual neurosciences, LM Chalupa, JS Werner, Eds* (pp. 1603–1615). MIT Press.

Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images [Number: 6583 Reporter: Nature]. *Nature*, *381*(6583), 607–609. https://doi.org/10.1038/381607a0

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? [Number: 23 Reporter: Vision Research]. *Vision Research*, *37*(23), 3311–3325. https://doi.org/10.1016/s0042-6989(97)00169-7

Ponulak, F., & Hopfield, J. J. (2013). Rapid, parallel path planning by propagating wavefronts of spiking neural activity. *Frontiers in Computational Neuroscience*, *7*. https://doi.org/10.3389/fncom.2013.00098

Restat, J. D., Steck, S. D., Mochnatzki, H. F., & Mallot, H. A. (2004). Geographical Slant Facilitates Navigation and Orientation in Virtual Environments. *Perception*, *33*(6), 667–687. https://doi.org/10.1068/p5030

Ringach, D. L. (2002). Spatial Structure and Symmetry of Simple-Cell Receptive Fields in Macaque Primary Visual Cortex [Publisher: American Physiolog-

ical Society]. *Journal of Neurophysiology*, *88*(1), 455–463. https://doi.org/10.1152/jn.2002.88.1.455

Rolls, E. T. (2013). The mechanisms for pattern completion and pattern separation in the hippocampus. *Frontiers in Systems Neuroscience*, *7*. https://doi.org/10.3389/fnsys.2013.00074

Rozell, C. J., Johnson, D. H., Baraniuk, R. G., & Olshausen, B. A. (2008). Sparse Coding via Thresholding and Local Competition in Neural Circuits [Number: 10 Reporter: Neural Computation]. *Neural Computation*, *20*(10), 2526–2563. https://doi.org/10.1162/neco.2008.03-07-486

Scheetz, S. D., Shao, E., Zhou, Y., Cario, C. L., Bai, Q., & Burton, E. A. (2018). An open-source method to analyze optokinetic reflex responses in larval zebrafish. *Journal of Neuroscience Methods*, *293*, 329–337. https://doi.org/10.1016/j.jneumeth.2017.10.012

Schick, W., Halfmann, M., Hardiess, G., Hamm, F., & Mallot, H. A. (2019). Language cues in the formation of hierarchical representations of space. *Spatial Cognition & Computation*, 1–30. https://doi.org/10.1080/13875868.2019.1576692

Scholkopf, B., & Mallot, H. A. (1995). View-Based Cognitive Mapping and Path Planning. *Adaptive Behavior*, *3*(3), 311–348. https://doi.org/10.1177/105971239500300303

Simoncelli, E. P., & Olshausen, B. A. (2001). Natural Image Statistics and Neural Representation [Publisher: Annual Reviews]. *Annual Review of Neuroscience*, *24*(1), 1193–1216. https://doi.org/10.1146/annurev.neuro.24.1.1193

Sivic, J., & Zisserman, A. (2003b). Video Google: A text retrieval approach to object matching in videos. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 1470–1477. https://doi.org/10.1109/ICCV.2003.1238663

Tinbergen, N., & Kruyt, W. (1938). Über die orientierung des Bienenwolfes (Philanthus triangulum Fabr.) *Zeitschrift für vergleichende Physiologie*, *25*(3), 292–334.

Tsotsos, J. K., & Rothenstein, A. L. (2011). The Role of Attention in Shaping Visual Perceptual Processes. *Perception-Action Cycle* (pp. 5–21). Springer.

Wang, K., Hinz, J., Haikala, V., Reiff, D. F., & Arrenberg, A. B. (2019). Selective processing of all rotational and translational optic flow directions in the zebrafish pretectum and tectum. *BMC Biology*, *17*(1). https://doi.org/10.1186/s12915-019-0648-2

Zylberberg, J., Murphy, J. T., & DeWeese, M. R. (2011). A Sparse Coding Model with Synaptically Local Plasticity and Spiking Neurons Can Account for the Diverse Shapes of V1 Simple Cell Receptive Fields (O. Sporns, Ed.) [Number: 10 Reporter: PLoS Computational Biology]. *PLoS Computational Biology*, *7*(10), e1002250. https://doi.org/10.1371/journal.pcbi.1002250

Chapter 7

Conclusion

The aim of this thesis is to add knowledge to our understanding of mental models and the underlying neuronal rules that lead to the emergence of these mental models. In the following, I summarize the conclusions from all four chapters with respect to this aim.

Sparse coding serves as a model for the formation of neuronal circuitry, especially in the visual cortex. Models for an implementation in neuronal substrate are mature. The essential elements are: feed forward hebbian learning of input relations and lateral anti-hebbian learning that enables delimitation of similar input. Sparse coding of sensory input is an unsupervised learning method of early sensory models. My thesis contains two examples, in which the representations obtained with sparse coding show a clear relationship to physical properties of the environment and may therefore serve as a sensory model.

In the first sparse coding study, presented in chapter 2, the Locally Competitive Algorithm (LCA), was applied to stereo image data. The units of the representation were selective to physical properties of the environment: to disparity of image structures, the cause of which is the distance of objects to the observer; to the absence of corresponding structures in stereo half images, the cause of which are occlusion boundaries of objects; and to the anisotropic visual compression of surface textures, the cause of which is the orientation of the surface with respect to the observer. We were able to show that the population code of the representation is sufficient for inferring a depth map from stereo images, which may serve as a model for the spatial structure of the environment.

In the second sparse coding study, presented in chapter 3, the LCA was applied to wide-field optic flow data, modeled after the visual system of zebrafish. It was

applied after the extraction of optic flow. Note that, because the nature of the optic flow problem is similar to the stereo problem—both are based on the extraction of disparities from consecutive images—this first processing step could also be accomplished with sparse coding. Again, the units of the representations were selective to physical properties of the environment: to ego-motion of the animal in six degrees of freedom. The representation may therefore serve as the basis for a model of the spatial relationship between the animal and the environment.

Our navigation model with dual population code, presented in chapter 4, can be classified as a topological navigation method. It therefore differs substantially from standard survey navigation schemes for robotic applications. Our model rests on the association of visual patterns with places and on the association of these places with motion instructions from one place to the other. The used information and their representation is parsimonious in our model. Places, as well as motion instructions, are not encoded explicitly, but in a population code. Their representation is noisy and ambiguous and individual units are not unique to the place fields they represent. Our successful evaluation confirms that simple rules of association, combined with a biologically plausible planning method, are sufficient for wayfinding in larger environments. The model is a step towards the outline of a plausible evolutionary path, which describes the emergence of navigational skills starting with recognition triggered responses.

In order to test hypotheses on mental models or other cognitive processing capabilities, it is important to recognize the studied subject as a living system, which is embodied and enmeshed with the environment on every level. Therefore, the real nature of cognitive processing can only unfold if tested in a naturalistic perception-action cycle. An example for such a setup is our automated visual performance testing in mice, presented in chapter 5. Because auf the manifold of interactions in cognitive processes, quantification and isolation of individual capabilities requires a holistic view. This was already apparent in our case, where the task was to quantify a simple reflex, but is even more important in the case of testing models of cognitive functions.

In summary, the first two studies contained in this thesis have contributed to our understanding of sensory representations, obtained with sparse coding, as early mental models of physical properties of the environment. Additionally, this thesis contains the outline of a mental model of environmental spatial relationships for

navigational purposes, which is parsimonious in nature and therefore a possible intermediate step that explains the evolutionary emergence of navigational skills. Finally, this thesis contains a study that establishes a setup to test animals in the perception-action cycle, which is important in order to reveal the real nature of cognitive processes in living systems.