Temporal slowness as an unsupervised learning principle: self-organization of complex-cell receptive fields and application to pattern recognition

DISSERTATION

zur Erlangung des akademischen Grades
doctor rerum naturalium
(Dr. rer. nat.)
im Fach Biophysik

eingereicht an der
Mathematisch-Naturwissenschaftlichen Fakultät I
der Humboldt-Universität zu Berlin

von
Herrn Dipl.-Math. Pietro Berkes
geboren am 4.5.1977 in Mendrisio (CH)

Präsident der Humboldt-Universität zu Berlin:
Prof. Dr. Jürgen Mlynek

Dekan der Mathematisch-Naturwissenschaftlichen Fakultät I:
Prof. Thomas Buckhout, PhD

Gutachter:

1. Laurenz Wiskott
2. Peter König
3. Andreas Herz

eingereicht am: 30.3.2005
Tag der mündlichen Prüfung: 23.6.2005

Acknowledgments

I am aware that this work would not have been possible without the help of many people, to which I am greatly indebted. My first thought goes to my advisor, Laurenz Wiskott, whose mathematical intuition never ceased to amaze me. In these years he taught me too many things to mention, mostly by his example, and I am extremely grateful for this. I hope I will be able to work with you again, sometime in the future.

Tobias Blaschke was already working in Laurenz' group when I first arrived. He introduced me both to the secrets of the ITB and to Berlin. Thank you for all the technical discussions we had and even more for a lot of insider tips about Berlin nightlife. Special thanks to Tiziano Zito, colleague and friend, with whom I wrote a nice piece of software (MDP) and heard the guitar of Max Cavalera, live at the Columbiahalle. I shared the office also with Irina Erchova, Mathias Franzius, and Henning Sprekeler, whom I thank for all the *espresso* we brew with our *Krups* espresso-machine and for discussions and nonsenses.

I would like to thank everybody else at the ITB for the nice atmosphere they created, and especially Samuel Glauser, Aki Naito, Christian Petereit, Roland Schaette, Susanne Schreiber, Raphael Ritz, and Thomas Voegtlin. Many thanks also to Prof. Andreas Herz for having provided such a nice research environment, and to Andreas Hantschmann and Christian Waltermann for the administration of the computer network, a vital part of our job. I thank Tim Gollish and Martin Stemmler for useful comments on my first manuscript, and Thomas Neukircher at the mathematical department for helping me with some insidious details of the mathematics of manifolds. This work has been supported by a grant from the Volkswagen Foundation. Thank you for that.

A kiss to Eva, with whom I divided everything during our stay in Berlin. This thesis is dedicated to you. I am grateful to my mother, Daniela, and to my father, Hans, for their constant emotional and financial support during my studies and my PhD.

Contents

1 Introduction 1
1.1 Computational models of the sensory cortex . . . 1
1.2 Overview . . . 4

2 The temporal slowness principle 7
2.1 Motivation . . . 7
2.2 Models based on temporal slowness . . . 10
2.3 Slow feature analysis . . . 13
    2.3.1 Problem statement . . . 13
    2.3.2 The linear case . . . 15
    2.3.3 The general case . . . 16
    2.3.4 The SFA algorithm . . . 16
2.4 Technical remarks to Chapter 2 . . . 18
    2.4.1 Alternative procedure to perform SFA . . . 18

3 Temporal slowness as a model for self-organization of receptive fields in the primary visual cortex 19
3.1 Introduction . . . 19
3.2 Methods . . . 20
    3.2.1 Input data . . . 20
    3.2.2 Analysis methods . . . 21
3.3 Results . . . 27
3.4 Control experiments . . . 35
    3.4.1 Control Experiment 1 . . . 35
    3.4.2 Control Experiment 2 . . . 37
    3.4.3 Control Experiment 3 . . . 37
    3.4.4 Control Experiment 4 . . . 37
3.5 Discussion . . . 38
    3.5.1 Other theoretical studies . . . 38
    3.5.2 Function space and equivalent network architecture . . . 39
    3.5.3 Relation to other learning rules . . . 41
    3.5.4 Conclusion . . . 43
3.6 Technical remarks to Chapter 3 . . . 44
    3.6.1 Dimensionality reduction by PCA . . . 44
    3.6.2 Receptive field localization . . . 44
    3.6.3 Inhibition in cortical neurons . . . 44
    3.6.4 Frequency tuning, digitalization, and dimensionality reduction . . . 45

3.6.5 Direction selectivity and velocity distribution 45
3.6.6 Tonic cells 45
3.6.7 Derivation of the relation to temporal coherence 45

4 Analysis and interpretation of inhomogeneous quadratic forms as receptive fields 47
4.1 Introduction 47
4.2 Model system 48
4.3 Visualization of coefficients and eigenvectors 49
4.4 Optimal stimuli 51
4.5 Invariances 56
4.6 Significant invariances 61
4.7 Relative contribution of the quadratic, linear, and constant term 62
4.8 Quadratic forms without linear term 63
4.9 Decomposition of a quadratic form in a neural network 65
4.10 Conclusion 66

5 Pattern recognition with slow feature analysis 69
5.1 Introduction 69
5.2 SFA for pattern recognition 69
5.3 Example application 71
5.3.1 Methods 71
5.3.2 Results 73
5.3.3 Extensions 77
5.4 Conclusions 77
5.5 Technical remarks to Chapter 5 78
5.5.1 Input sequences in the limit case 78
5.5.2 Dimensionality of a space of polynomials 78
5.5.3 The solutions of Equation 5.11 78

6 Conclusion and future directions 81

A Additional material 83

B Zusammenfassung in deutscher Sprache 84
B.1 Einleitung 84
B.2 Das Prinzip der zeitlichen Langsamkeit 84
B.3 Zeitliche Langsamkeit als Modell für rezeptive Felder in V1 85
B.4 Analyse und Interpretation von quadratischen Formen als rezeptive Felder 85
B.5 Mustererkennung mit Slow Feature Analysis 85
B.6 Schlussfolgerung 85

List of Figures

1.1 Quality of the sensory input 2
1.2 Function and implementation 2

2.1 The temporal slowness principle 8
2.2 Invariance manifold 9
2.3 Temporal coherence 13
2.4 Optimal free responses 14

3.1 Classical models of simple- and complex-cells 20
3.2 Natural image sequences 21
3.3 Optimal stimuli 23
3.4 Test and response images 24
3.5 Response images for the classical models of simple- and complex-cells 25
3.6 Response images of neurons in the visual system 26
3.7 Beta values 27
3.8 Active inhibition 28
3.9 Population statistics 30
3.10 Frequency and orientation bandwidth 32
3.11 End and side inhibition 33
3.12 Direction selectivity 34
3.13 Control Experiments 1 and 2 36
3.14 Control Experiment 3 38
3.15 Equivalent neural network 41
3.16 Units activity 42

4.1 Quadratic form coefficients 50
4.2 Eigenvectors of the quadratic term 51
4.3 Optimal stimuli 55
4.4 Geodetics on a sphere 58
4.5 Visualization of the invariances 59
4.6 Invariance movies 59
4.7 Significant invariances 62
4.8 Relative contribution of the quadratic, linear, and constant term 63
4.9 Interpretation of the invariances 65
4.10 Neural networks related to inhomogeneous quadratic forms 66
4.11 Random rotations of the positive and negative subunits 67

5.1 SFA for pattern recognition 70

5.2 Output signals 72
5.3 Δ-values 73
5.4 Test and training error 74
5.5 Classification errors 75
5.6 Dimensionality reduction 75

List of Tables

4.1 Definition of some important terms 48
4.2 Algorithm to compute the optimal stimuli 54
4.3 Algorithm to compute and visualize the invariances 60

5.1 Performance comparison 76

Chapter 1

Introduction

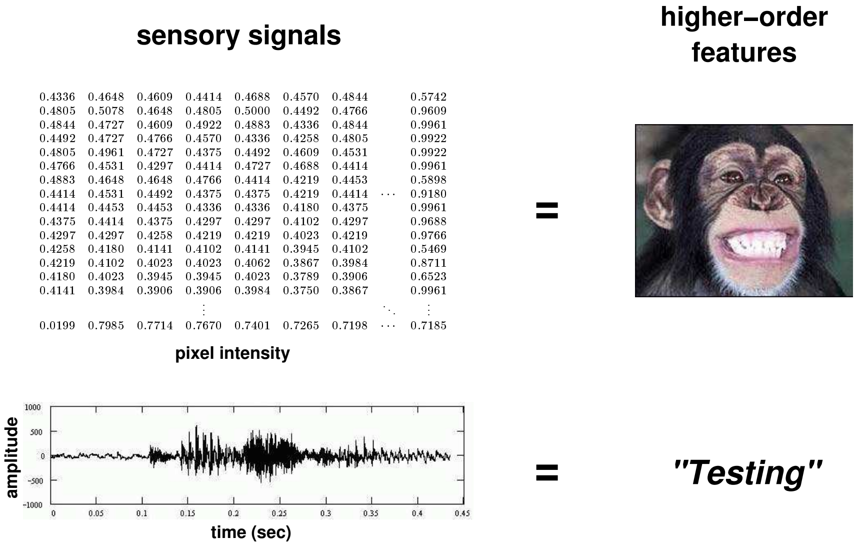
1.1 Computational models of the sensory cortex

We experience the world in a highly structured and consistent way. We perceive it as composed by separate entities, and these entities and the relations between them vary continuously in time and space. Yet if we consider the signals received by the sensory cortex we see that they are of a completely different quality. The information about the salient features of the environment seems to be nonlinearly mixed and dispersed among the receptors and to be lost in a huge amount of detailed and direct measurements of some aspects of our physical reality (Fig. 1.1). How does the cortex manage to recover the high-level, stable, and behaviorally relevant features of the environment from the sensory signals?

Physiological and anatomical studies have provided us with a great amount of data about the mechanisms by which the sensory input is processed by the cortex, especially in the first processing stages. These studies give us mainly a phenomenological account of the properties of single neurons (optimal stimuli, response tuning curves, adaptation characteristics, etc.) and of the communications between neurons (connections between different areas, inhibitory and excitatory connections, etc.).

An alternative approach is to look for a possible computational principle explaining the organization of the sensory cortex [Attneave, 1954, Barlow, 1961, Field, 1994, Zetzsche and Krieger, 2001, Olshausen, 2003, Simoncelli, 2003], which would provide particular advantages for further encoding and processing. The focus here is on the functional aspect of the cortex rather than on the mechanisms underlying the computation, which represent one of the (potentially many) possible implementations of the principle (Fig. 1.2). The importance of this approach has grown in the last couple of decades in response to the need of a theoretical framework to motivate experiments and to interpret their results [Olshausen, 2003]. Its role is best illustrated with the famous example by Barlow [1961]: "The situation may be likened to trying to understand how birds fly without knowledge of the principles of aerodynamics: no amount of experimentation or measurements made on the bird itself will reveal the secret. The key experiment – measuring air pressure above and below the wing as air is passed over it – would not seem obvious were it not for a theory suggesting why the pressures might be different." [Olshausen, 2003].

The computational point of view has a long tradition in psychology in terms of understanding perception as the internalization of the regularities of the environment [see Barlow, 2001, for a review]. It was however not before the classic papers by Attneave [1954] and Barlow [1961] that these ideas were placed in an information theoretical framework, which allowed to quantify these regularities. These first ideas were based on the concepts of efficient coding and redundancy reduction, which are similar to the modern *compactness* and *independence* principles (see below). Other computational principles that have been proposed later are *sparseness* and *temporal slowness*. These basic concepts were further formalized and investigated



$$\begin{matrix}
0.4336 & 0.4648 & 0.4609 & 0.4414 & 0.4688 & 0.4570 & 0.4844 & & 0.5742 \\
0.4805 & 0.5078 & 0.4648 & 0.4805 & 0.5000 & 0.4492 & 0.4766 & & 0.9609 \\
0.4844 & 0.4727 & 0.4609 & 0.4922 & 0.4883 & 0.4336 & 0.4844 & & 0.9961 \\
0.4492 & 0.4727 & 0.4766 & 0.4570 & 0.4336 & 0.4258 & 0.4805 & & 0.9922 \\
0.4805 & 0.4961 & 0.4727 & 0.4375 & 0.4492 & 0.4609 & 0.4531 & & 0.9922 \\
0.4766 & 0.4531 & 0.4297 & 0.4414 & 0.4727 & 0.4688 & 0.4414 & & 0.9961 \\
0.4883 & 0.4648 & 0.4648 & 0.4766 & 0.4414 & 0.4219 & 0.4453 & & 0.5898 \\
0.4414 & 0.4531 & 0.4492 & 0.4375 & 0.4375 & 0.4219 & 0.4414 & \cdots & 0.9180 \\
0.4414 & 0.4453 & 0.4453 & 0.4336 & 0.4336 & 0.4180 & 0.4375 & & 0.9961 \\
0.4375 & 0.4414 & 0.4375 & 0.4297 & 0.4297 & 0.4102 & 0.4297 & & 0.9688 \\
0.4297 & 0.4297 & 0.4258 & 0.4219 & 0.4219 & 0.4023 & 0.4219 & & 0.9766 \\
0.4258 & 0.4180 & 0.4141 & 0.4102 & 0.4141 & 0.3945 & 0.4102 & & 0.5469 \\
0.4219 & 0.4102 & 0.4023 & 0.4023 & 0.4062 & 0.3867 & 0.3984 & & 0.8711 \\
0.4180 & 0.4023 & 0.3945 & 0.3945 & 0.4023 & 0.3789 & 0.3906 & & 0.6523 \\
0.4141 & 0.3984 & 0.3906 & 0.3906 & 0.3984 & 0.3750 & 0.3867 & & 0.9961 \\
 & & & \vdots & & & & \ddots & \vdots \\
0.0199 & 0.7985 & 0.7714 & 0.7670 & 0.7401 & 0.7265 & 0.7198 & \cdots & 0.7185
\end{matrix}$$

Figure 1.1 Quality of the sensory input This figure illustrates the qualitative difference between the sensory input received by the cortex and what we experience. The sensory receptors make raw, direct measurements of the environment. This is comparable to reading the byte values in a file in a digital camera or observing the pressure wave generated by someone speaking (left). Although the information about the salient features is present (since for example the light intensity in a point of a visual scene depends on the identity and position of objects and lights), it is nonlinearly mixed and dispersed over the input signals. The quality of this representation is very different from what we experience (right).

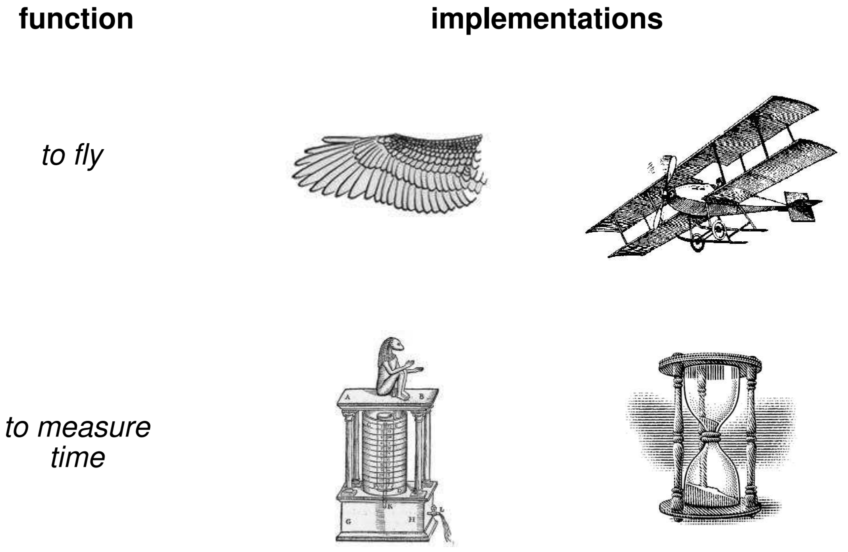


Figure 1.2 Function and implementation This figure illustrates the difference between a computational approach to the brain, where the function of neural system is investigated, and a more phenomenological approach, where the mechanisms underlying the computation are studied. The function is usually distinct from its implementation and it can be realized by different mechanisms.

with mathematical models and computer simulations, and have been applied with some success especially to the first stages of visual processing: retina [Atick and Redlich, 1992, Atick et al., 1992], lateral geniculate nucleus (LGN) [Dong and Atick, 1995a], and primary visual cortex (V1) [Hancock et al., 1992, Field, 1994, Olshausen and Field, 1996, Bell and Sejnowski, 1997, van Hateren and van der Schaaf, 1998, Hyvärinen and Hoyer, 2000, Zetzsche and Röhrbein, 2001, Einhäuser et al., 2002, Berkes and Wiskott, 2005b, see Sect. 3.5 for a complete overview]. In this thesis we focus on the *temporal slowness* principle, which is presented in detail in Chapter 2. In the following we give an overview of other computational principles that have been proposed as a model for self-organization in the sensory system.

The idea behind *compactness* or *compact coding* is to represent the likely input with a small number of components with minimal loss in the reconstruction, obtaining in this way a compressed and economical code. The redundancy of the input signals is reduced by eliminating the directions with low variance. This hypothesis, however, does not fit the available physiological and anatomical data and the resulting representation might make some of the task solved by the cortex difficult, like for example detecting associations (since the neurons are activated by most of the input stimuli) [Barlow, 2001]. In the linear case the mathematical formulation of compact coding is equivalent to principal component analysis (PCA). PCA components of natural images are Fourier filters with increasing frequencies [Hancock et al., 1992, Field, 1994], where Fourier filters with equal frequencies can be mixed without changing the compactness objective. Most of the solutions do not resemble the receptive fields of neurons in V1 [Field, 1994]. For these reasons the compact coding hypothesis has been rejected at least for the visual cortex.

According to the *independence* principle the code used by the cortex is such that the responses of the units that form the cortical representation are statistically independent [Bell and Sejnowski, 1997, van Hateren and van der Schaaf, 1998, Hoyer and Hyvärinen, 2000, Hyvärinen and Hoyer, 2000, Szatmáry and Lõrincz, 2001]. This can be motivated as a way to recover the independent sources that generate the input, or as an efficient code in terms of a reduction of the redundancy among different units, in the sense that the information coded by a given unit is not duplicated. In addition, in an independent representation it is possible to efficiently compute the joint probabilities of multiple events happening simultaneously, since it is just the product of the probabilities of the single events alone [Olshausen, 2003]. Note that the total redundancy does not decrease as in the compact coding case. The goal here is to *model* the redundancy instead of reducing it [Barlow, 2001]. It has been noticed that in linear models the units learned with this principle are not completely statistically independent. The remaining higher-order dependencies can be transformed with an appropriate nonlinear function into second-order dependencies and optimized again in a second stage [Zetzsche and Krieger, 2001, Zetzsche and Röhrbein, 2001, Karklin and Lewicki, 2003].

A closely related principle is that of *sparseness* [Field, 1994, Olshausen and Field, 1996, Olshausen, 2002]. According to this principle the transformation performed by the cortex is such that the resulting code is optimally sparse, i.e. such that the number of units responding to a particular input is minimized. Every unit has the same probability of responding, but for each individual unit the probability to respond is low. In this way events in the world are described by a small number of simultaneously active units [Olshausen, 2003]. Sparse codes are advantageous because they increase the signal-to-noise ratio, improve the detection of "suspicious coincidences", and allow effective storage in associative memories [Field, 1994]. Recent physiological experiments [Vinje and Gallant, 2000] seem to confirm that the response of neurons in V1 during natural vision is sparse. However, this might be the result of the high selectivity of the neurons, while their response need not necessarily be *maximally* sparse. Although their motivation is slightly different, sparseness and independence are found to be equivalent in some important cases [Hyvärinen et al., 2001]. Intuitively, this is because sparse signals are highly non-Gaussian and non-Gaussianity is commonly used as an objective to perform independent component analysis. The independence and sparseness principle have been shown to reproduce simple-cell receptive fields when applied to natural images (see Sect. 3.5).

An appealing aspect of computational models of sensory processing is that they are likely to have an impact on technological applications. To cite Olshausen [2003]: "Neuroscientists are interested in understanding how the cortex extracts certain properties of the visual environment [...] from the data stream coming from the retina. Similarly, engineers are interested in designing algorithms capable of extracting structure contained in images or sound, for example to identify and label parts within the body from medical imaging data. These problems at their core are one and the same, and progress in one domain will likely lead to new insights in the other.” The independence principle, for example, has been successfully applied to image denoising, image classification, analysis of EEG recordings, telecommunications, etc. [Hyvärinen et al., 2001, Lee and Lewicki, 2002]. The temporal slowness principle has been applied to nonstationary time series to extract underlying driving forces [Wiskott, 2003a], which might have applications for example in the analysis of financial data. In Chapter 5 we present an engineering application of this principle to pattern recognition.

1.2 Overview

In this thesis we investigate the relevance of *temporal slowness* as a principle for the self-organization of the sensory cortex and for technical applications.

In Chapter 2 we introduce and discuss the temporal slowness principle, and we give an overview of previous work on the subject. This principle is then put in mathematical terms in Section 2.3. In the same section we also define the slow feature analysis (SFA) algorithm, which solves the mathematical problem for multidimensional, discrete time series in a finite dimensional function space. The algorithm is presented in its formulation as a generalized eigenvalue problem that was introduced in [Berkes and Wiskott, 2003]. With respect to its original formulation in [Wiskott and Sejnowski, 2002] it is more efficient in terms of memory requirements and speed (see Sect. 2.4.1).

In Chapter 3 we apply temporal slowness as a learning principle of receptive fields in the primary visual cortex. Using SFA we learn the input-output functions that, when applied to natural image sequences, vary as slowly as possible in time and thus optimize the slowness objective. The resulting functions can be interpreted as nonlinear spatio-temporal receptive fields. We find that they reproduce many properties of complex cells in the primary visual cortex (V1), not only the two basic ones, namely a Gabor-like optimal stimulus and phase-shift invariance, but also secondary ones like direction selectivity, non-orthogonal inhibition, end-inhibition, and side-inhibition. This is illustrated both qualitatively and by quantitative comparisons with the population statistics of complex cells in V1. These results demonstrate that a single unsupervised learning principle can account for a rich repertoire of receptive field properties. We also perform a set of control experiments to investigate the role of the statistics of natural images and of the spatial transformations used to create the image sequences.

In order to analyze the nonlinear functions learned by SFA in our model, we developed a set of mathematical and numerical tools to characterize quadratic forms as receptive field. We expanded them to be of more general interest for second-order approximations and theoretical models of physiological receptive fields, and present them here in a separate chapter (Chap. 4).

We conclude this thesis by showing the application of the temporal slowness principle to pattern recognition (Chap. 5). We reformulate the SFA algorithm such that it can be applied to pattern recognition problems that lack a temporal structure and present the optimal solutions in this context. We then apply this system to a standard handwritten digits database.

At the end of each chapter we added a section with additional notes that concern more technical aspects which might be useful to complete and deepen the general picture, but are not necessary to understand the main results.

Original scientific contributions of this thesis include: i) The reformulation of the slow feature analysis algorithm as a generalized eigenvalue problem; ii) The investigation of temporal slowness as a learning principle of receptive fields in the primary visual cortex; iii) The development of mathematical and numerical tools for the analysis and interpretation of quadratic forms as receptive fields; iv) The application of the temporal slowness principle to pattern recognition to learn a feature space representation suitable to perform classification with simple methods.

Chapter 2

The temporal slowness principle

2.1 Motivation

The models presented in this thesis are based on the computational principle of *temporal slowness*. In this chapter we illustrate the motivation behind slowness as a learning principle for the sensory cortex and as a way to solve the technical problem of learning invariances from time series. We then give an overview of different definitions used in other studies (Sect. 2.2). Finally, in Section 2.3 we introduce slow feature analysis (SFA), an unsupervised algorithm that optimizes the slowness objective that will be used for the simulations in this thesis.

The temporal slowness principle is based on the observation that the environment, sensory signals, and internal representations of the environment vary on different time scales. The relevant variables in the environment change usually on a relatively slow time scale, like for example the identity or position of the objects that surround us. Sensory signals on the other hand consist of raw, direct measurements of our physical reality, and are thus in general very sensitive to changes in the environment or the observer. As a consequence, they vary on a faster time scale. For example, even a small eye movement or shift of a textured object may lead to a rapid variation of the light intensity received by a receptor neuron and as a consequence to a rapid variation of its output. The internal representations of the environment, finally, should vary on a time scale similar to that of the environment itself, that is on a slow time scale. This is illustrated in Figure 2.1. If we succeed in extracting slowly varying features from the quickly varying sensory signals, these features are likely to reflect the properties of the environment and are in addition invariant or at least robust to frequent transformations of the sensory input, such as visual translation, rotation, or zoom. For these reasons the extraction of slowly varying signals out of the input sensory signals has been proposed as a possible computational principle for the cortex [Mitchison, 1991, Földiák, 1991, Stone and Bray, 1995, Wiskott and Sejnowski, 2002, Hurri and Hyvärinen, 2003a, Körding et al., 2004, Berkes and Wiskott, 2005a]. In Chapter 3 we investigate temporal slowness as a learning principle for receptive fields in the primary visual cortex. The functions that vary most slowly in response to natural image sequences are found to reproduce many properties of complex cells in V1.

From a more technical point of view, the temporal slowness principle can also be regarded as an unsupervised learning principle to learn to represent salient features of an input stream of data in a way invariant to frequent transformations. This is illustrated in Figure 2.2. Consider a particular feature of an input data set, for example an object in a visual scene. Often the same feature is represented by a large set of different instances: For example in Figure 2.2 the image on the left and that on the right are different, but they represent to the same object. Each individual instance represents one point in a multidimensional input space. We can imagine that all points corresponding to the same feature form a surface, sometimes called *invari-*

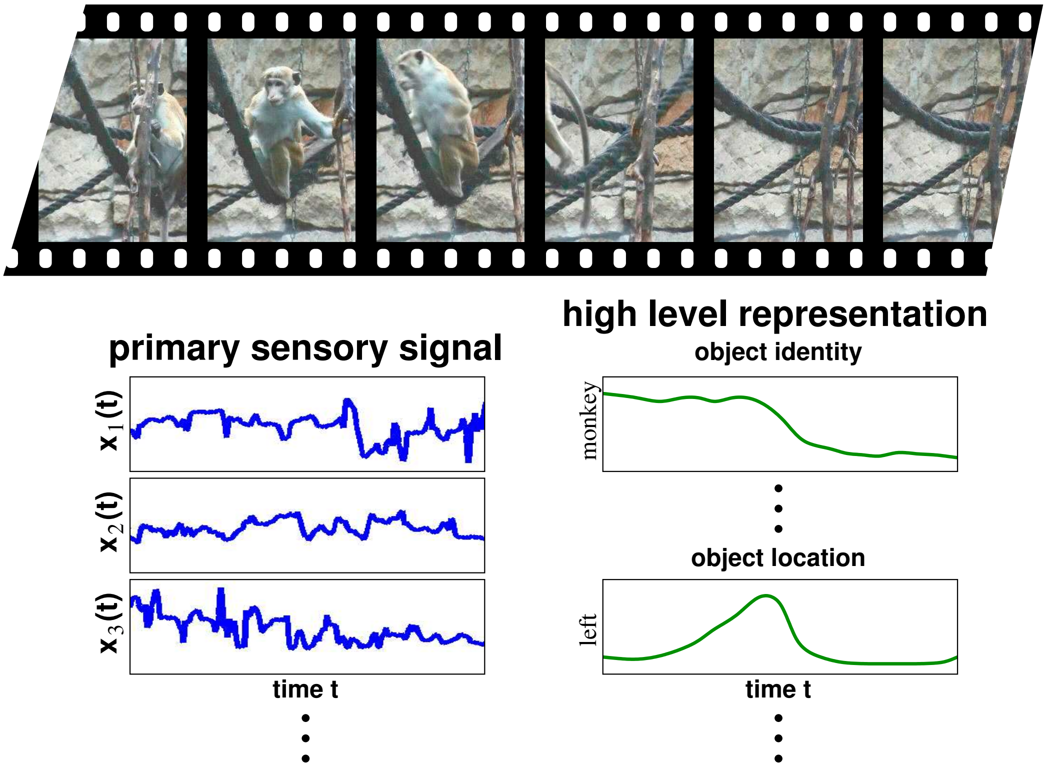


Figure 2.1 The temporal slowness principle This figure illustrates the temporal slowness principle. The image sequence at the top shows a visual scene representing the environment. The light intensity at one specific point in the scene is very sensitive to small changes in the environment (e.g. the translation of a textured object) or in the observer (e.g. a movement of the head). As a consequence the output signals of light-intensity receptors is sensitive to these changes, too, as shown in the plots at the bottom left. The relevant variables in the environment, however, change on a much slower timescale (bottom right). If we succeed in extracting slowly-varying signals out of the input data, we are likely to recover some salient features of the environment.

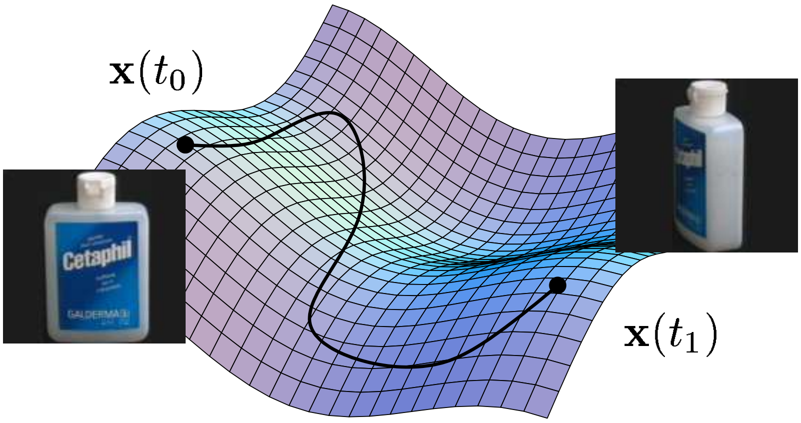


Figure 2.2 Invariance manifold The relevant features of the input space are often represented by a large number of different instances. For example the same object can be represented by different images (in the squared panels). We can imagine that the set of all image instances of an object form a surface in the input space. A function that is invariant to the difference between individual instances will respond similarly to all points on the surface. If the input has a temporal structure such that the underlying features are stable in time (illustrated here by the path on the surface), an invariant function would change slowly in time.

ance manifold [Wiskott, 1998] (although it does not need to be a manifold in the mathematical sense). In many cases it is useful to have a representation that is invariant to the differences between the instances of a given feature. The functions g_i that transform the input into such a representation would need to respond in the same way to all points on the surface (i.e. the surface would be a level set of the functions, or a subset thereof). More realistically, the functions would need to change as little as possible on the surface. If the input data $\mathbf{x}$ has a temporal structure, the sequence of points $\mathbf{x}(t)$, $t \in [t_0, t_1]$ forms a trajectory in the input space. If we assume that neighboring points in time contain the same features they would lie on the invariance manifold. An invariant function g_i would thus change slowly in time when applied to the time series. Conversely, if we require a function to vary as slowly as possible in time, it will need to be invariant to frequent transformations of the instances. On the other hand, we also want the functions to distinguish between different features, which is achieved in practice by imposing additional constraints, for example by forcing the functions g_i to have unit variance. This more technical point of view is applied in Chapter 5, where we present the application of the temporal slowness principle to pattern recognition.

A secondary advantage of slowly varying signals is that they can be transmitted by channels with a lower bandwidth [Körding et al., 2004]. It has also been suggested that learning is easier in a system where variables change slowly [Körding et al., 2004]. However, this is not necessarily the case, as proved by the fact that many learning algorithms do not consider the temporal structure of their input and one could thus rearrange the input data to vary fast in time without affecting the efficiency of learning.

2.2 Models based on temporal slowness

The temporal slowness objective has been formulated in different ways by different authors. In this section we give an overview of the different definitions and of the results of other studies based on this principle.

In the rest of this thesis we will drop references to time in equations if they are not necessary for the understanding. The input and output vectors $\mathbf{x}$ and $\mathbf{y}$ are always thought to form time series $\mathbf{x}(t)$ and $\mathbf{y}(t)$, unless otherwise stated. Their components will be indicated with x_i and y_i. w_{ij} is the weight coefficient between x_j and y_i, assuming a linear model $y_i = \sum_j w_{ij} x_j$ or a model with fixed nonlinearities h_j, $y_i = \sum_j w_{ij} h_j(\mathbf{x})$. $\langle\cdot\rangle_t$ indicates the mean over time.

An early description of temporal slowness was given by Hinton [1989, p. 208] as a principle for self-supervised learning in backpropagation neural networks. He suggested that this might be "a sensible principle if the input vectors are generated by a process whose underlying parameters change more slowly than the input vectors themselves". This method was meant to be applied to the units on the hidden layer of a neural network as a way to simplify supervised learning in successive layers. An analogous idea was applied by Peng et al. [1998] to a digit recognition problem using an objective that mixed a term related to slowness and one related to sparseness. Similarly, the pattern recognition system presented in Chapter 5 makes use of temporal slowness in order to learn a feature space representation suitable to easily perform the successive classification step.

In that paper Hinton did not present a mathematical formulation or results based on simulations. This was first done in the models by Mitchison [1991] and Földiák [1991]. Földiák's model was motivated by the fact that the visual system has to recognize objects in a way invariant to viewpoint, position, size, deformation, etc. His work was inspired by Fukushima's and LeCun's object recognition networks [Fukushima, 1980, LeCun et al., 1998], which are based on a hierarchical system of alternating layers of feature-detecting units (S-units) and invariance units (C-units) that pool over the feature detectors. In their model translation invariance is achieved by a weight-sharing mechanism for the S-units and by C-units that pool over S-units detecting the same feature at different positions in an image. Földiák's rule was intended as a way to *learn* how to connect a C-unit to the S-units in order to be invariant to frequent transformations. He proposed a modified Hebbian rule in which the weight updates are proportional to a trace of the outputs y_i with decay factor δ (i.e. to a low-pass filtered version of y_i):

$$\Delta w_{ij} = \alpha \bar{y}_i \left(x_j - w_{ij} \right) , \tag{2.1}$$

where

$$\bar{y}_i(t) = (1-\delta)\bar{y}_i(t-1) + \delta y_i(t) . \tag{2.2}$$

The input components x_j correspond in this model to the output of the S-units. The second term in (2.1) was added to keep the weight vector bounded and a winner-take-all strategy was applied in order to have the units learn different invariances. Wallis and Rolls [1997] used a hierarchical network similar to that proposed by Fukushima [1980] and the trace update rule of Equation (2.1) to perform invariant face and object recognition.

Mitchison [1991] proposed a learning rule to remove systematic temporal variations from the input of a unit. To make the unit vary as little as possible in time, he proposed to minimize the objective function

$$E = \langle \dot{y}^2 \rangle_t \tag{2.3}$$

by gradient descent. For a linear unit $y = \sum w_i x_i$ he obtained the anti-Hebbian rule

$$\Delta w_i = -\alpha \Delta x_i \Delta y . \tag{2.4}$$

To avoid the weights converging to zero (which would minimize Eq. 2.3), the length of the weight vector was normalized to a fixed norm K at each iteration. As noticed in [Wiskott and Sejnowski, 2002] this is a scale-sensitive operation, so that the optimal solution would depend on the range of the input variables. It seems thus more reasonable to normalize the weights according to the variance of the output. Mitchinson proved that the solution $\mathbf{w}$ that optimizes Equation (2.3) is equal to the eigenvector with the smallest eigenvalue of the covariance matrix of $\mathbf{x}$ (cf. Sect. 2.3). In order to learn a set of different units, a bias term ζ was introduced that drives the unit toward a given response:

$$E = \langle \dot{y}^2 \rangle_t + \lambda \langle (\zeta - y)^2 \rangle_t \,, \tag{2.5}$$

where $\lambda > 0$. This leads to the gradient-descent rule

$$\Delta w_i = -\alpha \Delta x_i \Delta y + \beta x_i (\zeta - y) \,, \tag{2.6}$$

where $\beta = \alpha\lambda$. This additional term presumes of course a certain knowledge about desired solutions. In case such knowledge is not available, a better solution could be to use a decorrelation constraint (cf. Sect. 2.3.1).

Stone and Bray [1995] proposed an objective function that joins the necessity for a unit to vary in order to code for some hidden variable (avoiding an explicit normalization of the weights as in the previous model) and the fact that it should vary slowly in time according to the temporal slowness principle. The variability and slowness of a unit are expressed by the short-term variance U, which should be minimized, and the long-term variance V, which should be maximized. The proposed objective function is

$$\begin{aligned} E &= \frac{1}{2} \log \frac{V}{U} && (2.7) \\ &= \frac{1}{2} \log \left(\frac{\sum_t (y(t) - \bar{y}(t))^2}{\sum_t (y(t) - \tilde{y}(t))^2} \right) , && (2.8) \end{aligned}$$

where $\tilde{y}$ and $\bar{y}$ are a short-term and a long-term average defined by a trace rule like that of Equation 2.2 with different decay factors. The corresponding gradient-descent rule contains a Hebbian and an anti-Hebbian term:

$$\frac{\partial E}{\partial w_j} = \frac{1}{V} \langle (y - \bar{y})(x_j - \bar{x}_j) \rangle_t - \frac{1}{U} \langle (y - \tilde{y})(x_j - \tilde{x}_j) \rangle_t \tag{2.9}$$

This rule was first illustrated on the problem of learning to represent the position of one active element in a one- or two-dimensional grating by the value of its coordinates [Stone and Bray, 1995]. In [Stone, 1996] the same setting was tested in a two-layer neural network to extract the disparity information from random-dot stereograms and from images with artificially generated disparity. In [Stone, 2001] the problem was then reformulated as a generalized eigenvalue problem and applied to blind source separation, i.e. to the reconstruction of linearly mixed input sources.

Our work is based on the formulation of slowness given by [Wiskott, 1998, Wiskott and Sejnowski, 2002] and presented in more details in Section 2.3. The objective function is the same as in [Mitchison, 1991] (Eq. 2.3), with the addition of a unit variance and decorrelation constraint that permits to learn a whole set of different slowly-varying functions. Wiskott extended the slowness problem to nonlinear functions and introduced the slow feature analysis (SFA) algorithm that finds the global solutions by solving two eigenvalue problems. We successively reformulated the algorithm as a generalized eigenvalue problem [Berkes and Wiskott, 2003]. SFA is presented and discussed in Sections 2.3.2–2.3.4. In [Wiskott and Sejnowski, 2002] SFA was applied to the extraction of disparity from artificial simple-cells output and to a hierarchical network of SFA-modules. One-dimensional random objects were presented to the bottom layer, representing a 1D-retina. On the top layer the system learned to represent the *what* and *where* information of arbitrary 1D-objects (i.e. their identity and position) in an invariant fashion. These results have also been derived analytically in [Wiskott, 2003b], where an analytical and algebraic approach to the analysis of the solutions of the temporal slowness principle was introduced. SFA was also applied to the estimation of hidden variables underlying nonstationary time series [Wiskott, 2003b] and to blind source separation [Blaschke et al., 2004].

Kayser et al. [2001] and Körding et al. [2004] were the first to apply the temporal slowness principle to natural image sequences in order to make a comparison with receptive fields in the primary visual cortex (V1). In their papers they showed that neural networks adapted to optimize a slowness objective on natural image sequences reproduce the basic characteristics of complex cells in V1, i.e. a Gabor-like optimal stimulus and phase-shift invariance. The objective function they used is the sum of a slowness and a decorrelation term:

$$\begin{aligned} E &= E_{slowness} + E_{decorrelation} && (2.10)\\ &= \sum_i \langle \dot{y_i}^2 \rangle_t + \sum_{i \neq j} \langle y_i y_j \rangle_t && (2.11) \end{aligned}$$

which has to be minimized. The variance of the output signals of the units was normalized to be one, in order to avoid the trivial constant solution and to make the temporal variation of different signals comparable. The decorrelation term $E_{decorrelation}$ was responsible for the learning of different units. The slowness term $E_{slowness}$ is similar to the one used in this work (cf. Eq. 2.13), except that the *average* of the temporal variation of all units is optimized. Under some conditions, the resulting solutions are unique only up to an orthogonal transformation of the units (see Sect. 3.5.3). These and other studies modeling complex and simple cells are discussed in Section 3.5.

Hurri and Hyvärinen [2003a] proposed a definition of slowness, called *temporal coherence*, that has different properties with respect to the ones in the models cited above (see also Sect. 3.5.3 and Sect. 3.6.7). The basic idea is to make the *energy* of the output signal vary slowly by maximizing

$$\sum_i \left\langle f(y_i(t)) f(y_i(t + \Delta t)) \right\rangle_t , \qquad (2.12)$$

where f is an even and strictly convex nonlinearity. In the simulations f was set to $f_1(y) = y^2$, which measures the energy of the signal, or to $f_2(y) = \ln\cosh(y)$, which is a robust version of f_1. The output signals y_i is normalized such that they have unit variance and are decorrelated. Due to the even nonlinearity, strongly oscillating solutions can be optimal, since the sign is canceled out. This is illustrated in Figure 2.3: The two signals have zero mean and unit variance. Their energy varies equally smoothly in time (as indicated by the gray dashed curve for $f = f_1$), so that they are equally temporal coherent with respect to Equation (2.12). On the other hand, considering for example the objective of Equation (2.3), the second curve would vary faster than the first since the derivatives are larger. Also in the case of non-symmetric, rectifying nonlinearities the solutions found by (2.12) are different from the ones extracted using Equation (2.3).

Ideas and learning rules related to temporal slowness can also be found in [Becker and Hinton, 1992, O'Reilly and Johnson, 1994, Becker, 1996]. In [Becker and Hinton, 1992] two (or more) units received as input different aspects of the same input. A measure of mutual information between their outputs was maximized, so that the units had to ignore the particular aspect they were considering and concentrate on hidden variables common to both inputs, for example a common disparity signal underlying non-overlapping parts of random-dot stereograms [Becker and Hinton, 1992]. Their model could be expressed in a temporal form if the units would receive input from different points in time [Mitchison, 1991].

In the visual domain temporal and spatial slowness are closely related concepts. For example, temporal slowness could be reformulated as a spatial one by adapting each unit to respond in a similar way to neighboring visual regions. The slowness objective could thus be reformulated as a spatial optimization criterion [Stone, 1996, Wiskott and Sejnowski, 2002]. However, the former seems more natural to us and easier to implement in a biological system.

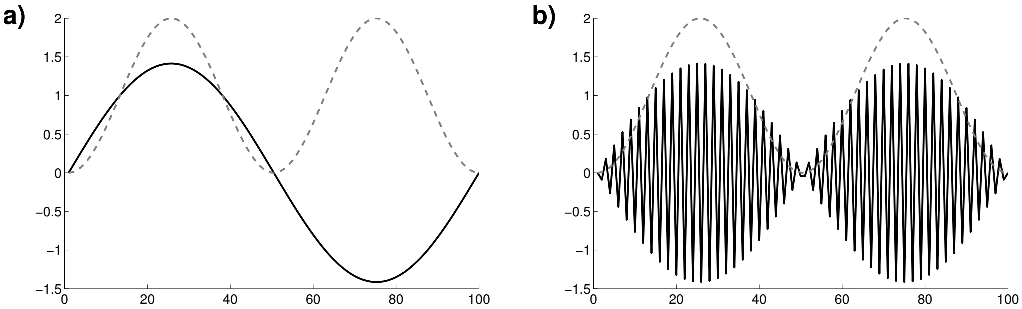


Figure 2.3 Temporal coherence The two signals in (a) and (b) have zero mean and unit variance. Their energy vary equally smoothly in time (as indicated by the gray dashed curve for $f = f_1$), so that they are equally temporal coherent with respect to Equation (2.12). On the other hand, considering the objective of Equation (2.3), the second curve would vary faster than the first since the the derivatives are larger.

2.3 Slow feature analysis

2.3.1 Problem statement

In the previous section we have seen that the temporal slowness principle has been formulated in different ways by different authors. In the rest of the chapter we give the mathematical formulation of the slowness objective that will be used throughout this thesis and introduce slow feature analysis (SFA), an unsupervised algorithm to determine the functions that optimize this objective.

As previously discussed our learning task is to find (scalar) functions $g_j(\mathbf{x})$ which generate output signals $y_j(t) = g_j(\mathbf{x}(t))$ from the input signal $\mathbf{x}(t)$ that vary as slowly as possible but carry significant information. The latter is ensured by requiring the output signals to have unit variance and be mutually uncorrelated. It is important to note that even though the objective is the slowness of the output signals the process by which the output is computed from the input is very fast or in the mathematical idealization even instantaneous. Slowness can therefore not be achieved simply by low-pass filtering. Thus only if the input signal has some underlying slowly varying causes does the system have a chance of extracting slowly varying output signals at all. It is exactly this apparent paradox of instantaneous processing on the one hand and the slowness objective on the other hand which guarantees that the extracted output signals represent relevant features of the underlying causes that gave rise to the input signal.

In mathematical terms the problem can be stated as follows [Wiskott and Sejnowski, 2002]: Given an input signal $\mathbf{x}(t) = (x_1(t), \ldots, x_N(t))^T$, $t \in [t_0, t_1]$, find a set of real-valued functions $g_1(\mathbf{x}), \ldots, g_M(\mathbf{x})$ lying in a function space $\mathcal{F}$ so that for the output signals $y_j(t) := g_j(\mathbf{x}(t))$

$$\Delta(y_j) := \langle \dot{y}_j^2 \rangle_t \quad \text{is minimal} \tag{2.13}$$

under the constraints

$$\begin{aligned} \langle y_j \rangle_t &= 0 \quad \text{(zero mean)}\,, && (2.14)\\ \langle y_j^2 \rangle_t &= 1 \quad \text{(unit variance)}\,, && (2.15)\\ \forall i < j, \quad \langle y_i y_j \rangle_t &= 0 \quad \text{(decorrelation and order)}\ \ , && (2.16) \end{aligned}$$

with $\langle . \rangle_t$ and $\dot{y}$ indicating time averaging and the time derivative of y, respectively. Equation (2.13) introduces a measure of the temporal variation of a signal (the Δ-*value* of a signal) equal to the mean of the squared derivative of the signal. This quantity is large for quickly-varying signals and zero for constant signals. We will also use a more intuitive measure of slowness, the *β-value*, defined as

$$\beta(y_j) = \frac{1}{2\pi}\sqrt{\Delta(y_j)}\,. \tag{2.17}$$

A sine wave with period T and unit variance has a β-value of $1/T$ when averaged over an integer number of oscillations. The zero-mean constraint (2.14) is present for convenience only, so that (2.15) and (2.16) take a simple form. Constraint (2.15) means that each signal should carry some information and avoids the trivial solution $g_j(\mathbf{x}) = 0$. Alternatively, one could drop this constraint and divide the right side of (2.13) by the variance $\langle y_j^2 \rangle_t$. Constraint (2.16) forces different signals to be uncorrelated and thus to code for different aspects of the input. It also induces an order, the first output signal being the slowest one, the second being the second slowest, etc.

This optimization problem is in general extremely difficult to solve analytically. Wiskott [2003b] was able to derive the analytical solution for free output signals, i.e. the solution obtained without the constraints coming from the input signals and the function space. The optimal free responses for cyclic or free boundary conditions on the output signal have been found to be sine and cosine waves of increasing frequency (Fig. 2.4). In the same paper, Wiskott introduced an algebraic framework that allowed him to derive analytically the results obtained in [Wiskott and Sejnowski, 2002]. Otherwise, for arbitrary, multidimensional time series and a finite-dimensional function space the solution to the slowness problem can be found numerically with the SFA algorithm, that we will define in the next three sections.

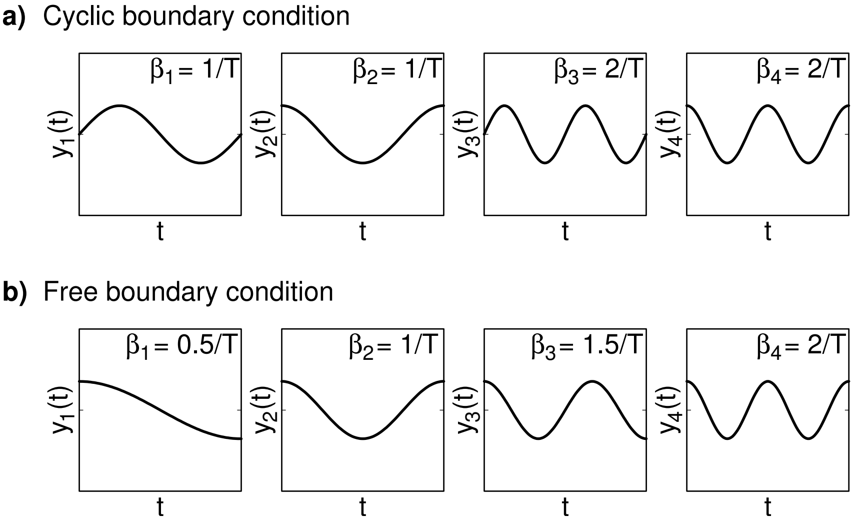


Figure 2.4 Optimal free responses This image shows the optimal free responses for the slowness optimization problem (Eqs. 2.13–2.16), i.e. the solutions obtained without the constraints coming from the input signals and the function space. y-axes range from -4 to +4; t-axes are of length T. The expressions in the plots indicate the β-value of the signals. **(a)** The first 4 optimal responses for cyclical boundary conditions. **(b)** The first 4 optimal responses for free boundary conditions. (Images courtesy of Laurenz Wiskott.)

2.3.2 The linear case

Consider first the linear case

$$g_j(\mathbf{x}) = \mathbf{w}_j^T \mathbf{x} \tag{2.18}$$

for some input $\mathbf{x}$ and weight vectors $\mathbf{w}_j$. In the following we assume $\mathbf{x}$ to have zero mean (i.e. $\langle \mathbf{x} \rangle_t = \mathbf{0}$) without loss of generality. This implies that Constraint (2.14) is always fulfilled, since

$$\langle y_j \rangle_t = \langle \mathbf{w}_j^T \mathbf{x} \rangle_t = \mathbf{w}_j^T \langle \mathbf{x} \rangle_t = 0\,. \tag{2.19}$$

We can rewrite Equations (2.13), (2.15) and (2.16) as

$$\begin{aligned} \Delta(y_j) &= \langle \dot{y_j}^2 \rangle_t & (2.20)\\ &= \langle (\mathbf{w}_j^T \dot{\mathbf{x}})^2 \rangle_t & (2.21)\\ &= \mathbf{w}_j^T \langle \dot{\mathbf{x}} \dot{\mathbf{x}}^T \rangle_t \mathbf{w}_j & (2.22)\\ &=: \mathbf{w}_j^T \mathbf{A} \mathbf{w}_j & (2.23) \end{aligned}$$

and

$$\begin{aligned} \langle y_i y_j \rangle_t &= \langle (\mathbf{w}_i^T \mathbf{x})(\mathbf{w}_j^T \mathbf{x}) \rangle_t & (2.24)\\ &= \mathbf{w}_i^T \langle \mathbf{x} \mathbf{x}^T \rangle_t \mathbf{w}_j & (2.25)\\ &=: \mathbf{w}_i^T \mathbf{B} \mathbf{w}_j\,. & (2.26) \end{aligned}$$

If we integrate Constraint (2.15) in the objective function (2.13), as suggested in the previous section, we obtain:

$$\Delta(y_j) \underset{(2.13,2.15)}{=} \frac{\langle \dot{y_j}^2 \rangle_t}{\langle y_j^2 \rangle_t} \underset{(2.23,2.26)}{=} \frac{\mathbf{w}_j^T \mathbf{A} \mathbf{w}_j}{\mathbf{w}_j^T \mathbf{B} \mathbf{w}_j}\,. \tag{2.27}$$

It is known from linear algebra that the weight vectors $\mathbf{w}_j$ that minimize this equation correspond to the eigenvectors of the generalized eigenvalue problem

$$\mathbf{A}\mathbf{W} = \mathbf{B}\mathbf{W}\mathbf{\Lambda}\,, \tag{2.28}$$

where $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_N)$ is the matrix of the generalized eigenvectors and $\mathbf{\Lambda}$ is the diagonal matrix of the generalized eigenvalues $\lambda_1, \ldots, \lambda_N$ [see e.g. Gantmacher, 1959, chap. 10.7, Theorem 8, 10 and 11]. The vectors $\mathbf{w}_j$ can be normalized such that $\mathbf{w}_i^T \mathbf{B} \mathbf{w}_j = \delta_{ij}$, which implies that Constraints (2.15) and (2.16) are fulfilled:

$$\begin{aligned} \langle y_j^2 \rangle_t &= \mathbf{w}_j^T \mathbf{B} \mathbf{w}_j = 1\,, & (2.29)\\ \langle y_i y_j \rangle_t &= \mathbf{w}_i^T \mathbf{B} \mathbf{w}_j = 0\,, \quad i \neq j\,. & (2.30) \end{aligned}$$

Note that, by substituting Equation (2.28) into Equation (2.27) one obtains

$$\Delta(y_j) = \lambda_j\,, \tag{2.31}$$

so that by sorting the eigenvectors by increasing eigenvalues we induce an order where the most slowly-varying signals have lowest indices (i.e. $\Delta(y_1) \leq \Delta(y_2) \leq \ldots \leq \Delta(y_N)$), as required by Constraint (2.16).

2.3.3 The general case

In the more general case of a nonlinear, finite-dimensional function space $\mathcal{F}$, consider a basis $h_1, \ldots, h_M$ of $\mathcal{F}$. For example, in the standard case where $\mathcal{F}$ is the space of all polynomials of degree d, the basis will include all monomials up to order d.

Defining the *expanded input*

$$\mathbf{h}(\mathbf{x}) := (h_1(\mathbf{x}), \ldots, h_M(\mathbf{x})) \tag{2.32}$$

every function $g \in \mathcal{F}$ can be expressed as

$$g(\mathbf{x}) \;=\; \sum_{k=1}^{M} w_k h_k(\mathbf{x}) \;=\; \mathbf{w}^T \mathbf{h}(\mathbf{x})\,. \tag{2.33}$$

This leads us back to the linear case if we assume that $\mathbf{h}(\mathbf{x}(t))$ has zero mean (again, without loss of generality), which can be easily obtained in practice by subtracting the mean over time $\langle \mathbf{h}(\mathbf{x}) \rangle_t =: \mathbf{h}_0$ from the expanded input signal.

For example, in the case of 3 input dimensions and polynomials of degree 2 we have

$$\mathbf{h}(\mathbf{x}) = (x_1^2,\, x_1x_2,\, x_1x_3,\, x_2^2,\, x_2x_3,\, x_3^2,\, x_1,\, x_2,\, x_3)^T - \mathbf{h}_0 \tag{2.34}$$

and

$$\begin{aligned} g(\mathbf{x}) \underset{(2.33)}{=} & \; w_1x_1^2 + w_2x_1x_2 + w_3x_1x_3 + w_4x_2^2 + w_5x_2x_3 + w_6x_3^2 \\ & + w_7x_1 + w_8x_2 + w_9x_3 - \mathbf{w}^T\mathbf{h}_0 \end{aligned} \tag{2.35}$$

Every polynomial of degree 2 in the 3 input variables can then be expressed by an appropriate choice of the weights w_i.

2.3.4 The SFA algorithm

We can now formulate the slow feature analysis algorithm [Wiskott and Sejnowski, 2002] (see also Sect. 2.4.1):

Non-linear expansion: Expand the input data and compute the mean over time $\mathbf{h}_0 := \langle \mathbf{h}(\mathbf{x}) \rangle_t$ to obtain the expanded signal

$$\mathbf{z} := \mathbf{h}(\mathbf{x}) - \mathbf{h}_0 \tag{2.36}$$

$$= (h_1(\mathbf{x}), \ldots, h_M(\mathbf{x}))^T - \mathbf{h}_0\,. \tag{2.37}$$

Slow feature extraction: Solve the generalized eigenvalue problem

$$\mathbf{AW} \;=\; \mathbf{BW\Lambda} \tag{2.38}$$

$$\text{with} \quad \mathbf{A} \;:=\; \langle \dot{\mathbf{z}}\dot{\mathbf{z}}^T \rangle_t \tag{2.39}$$

$$\text{and} \quad \mathbf{B} \;:=\; \langle \mathbf{z}\mathbf{z}^T \rangle_t\,. \tag{2.40}$$

The K eigenvectors $\mathbf{w}_1, \ldots, \mathbf{w}_K$ $(K \leq M)$ corresponding to the smallest generalized eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_K$ define the nonlinear input-output functions $g_1(\mathbf{x}), \ldots, g_K(\mathbf{x}) \in \mathcal{F}$:

$$g_j(\mathbf{x}) \;:=\; \mathbf{w}_j^T(\mathbf{h}(\mathbf{x}) - \mathbf{h}_0) \tag{2.41}$$

which satisfy Constraints (2.14)–(2.16) and minimize (2.13).

In other words to solve the optimization problem (2.13) it is sufficient to compute the covariance matrix of the signals and that of their derivatives in the expanded space and then solve the generalized eigenvalue problem (2.38). In the simulations presented in this thesis, the derivative of $\mathbf{z}(t)$ is computed by the linear approximation $\dot{\mathbf{z}}(t) \approx (\mathbf{z}(t+\Delta t) - \mathbf{z}(t))/\Delta t$ ($\Delta t = 1$ throughout the thesis). We performed some simulations of the model described in Chapter 3 with cubic interpolation and obtained equivalent results.

2.4 Technical remarks to Chapter 2

2.4.1 Alternative procedure to perform SFA

As originally defined in [Wiskott and Sejnowski, 2002] the SFA algorithm performs the slow feature extraction step by solving two eigenvalue problems. First, the data in the expanded space is whitened with a whitening matrix S, so that every orthonormal basis fulfills Constraints (2.14–2.16). Second, the covariance matrix of the derivatives is computed and the eigenvectors $\mathbf{w}_j$ corresponding to its smallest eigenvalues are extracted. The input-output function are defined as

$$\begin{aligned} g_j(\mathbf{x}) &:= \mathbf{w}_j^T \mathbf{S}(\mathbf{h}(\mathbf{x}) - \mathbf{h}_0) && (2.42) \\ &= (\mathbf{S}^T \mathbf{w}_j)^T (\mathbf{h}(\mathbf{x}) - \mathbf{h}_0) && (2.43) \end{aligned}$$

The advantage of the formulation as a generalized eigenvalue problem is that it is possible to use efficient algorithms that allow to compute just the set of eigenvectors we are interested in, while in the original definition it is necessary to solve one complete eigenvalue problem (for the whitening step). Since the dimensionality of the expanded space is usually very high, this leads an important reduction of memory requirements and computation time.

Chapter 3

Temporal slowness as a model for self-organization of receptive fields in the primary visual cortex

3.1 Introduction

As discussed in the two introductory chapters, the working hypothesis in this thesis is that the sensory cortex organizes according to the temporal slowness principle in order to build a consistent internal representation of the environment. In this chapter we verify this hypothesis for the primary visual cortex.

Primary visual cortex (V1) is the first cortical area dedicated to visual processing. This area has been intensively studied neurophysiologically since the seminal work by Hubel and Wiesel [1962], who also introduced the standard classification of neurons in V1 into two main groups: *simple* and *complex cells*. These neurons are conceived as edge or line detectors: simple cells respond to bars having a specific orientation and position in the visual field; complex cells also respond to oriented bars but are insensitive to their exact position. Idealized simple and complex cells can be described by Gabor wavelets [Pollen and Ronner, 1981, Adelson and Bergen, 1985, Jones and Palmer, 1987], which have the shape of sine gratings with a Gaussian envelope function. A single Gabor wavelet used as a linear filter (Fig. 3.1a) is similar to a simple cell, since the response depends on the exact alignment of a stimulus bar on an excitatory (positive) subfield of the wavelet. Taking the square sum of the responses of two Gabor wavelets with identical envelope function, frequency, and orientation but with a 90° phase difference (Fig. 3.1b) yields a model of a complex cell that is insensitive to the exact location of the bar (following a rule similar to the relation $\sin(x)^2 + \cos(x)^2 = 1$) while still being sensitive to its orientation. We will refer to these models as the *classical models* of simple and complex cells. This idealized picture, however, is clearly not complete. In particular complex cells in V1 show a much richer repertoire of receptive field properties than can be explained with the classical model. For example, they show end-inhibition, side-inhibition, direction selectivity, and sharpened or broadened tuning to orientation or frequency [Hubel and Wiesel, 1962, Sillito, 1975, Schiller et al., 1976a,b,c, De Valois et al., 1982a,b, Dobbins et al., 1987, Versavel et al., 1990, Skottun et al., 1991, DeAngelis et al., 1994, Shevelev, 1998, Walker et al., 1999, Ringach et al., 2002].

In order to verify the slowness hypothesis we consider a large set of natural image sequences, extract with SFA the functions that compute the slowest features from a space of nonlinear input-output functions (the space of all polynomials of degree 2), and compare their properties to those of complex cells in V1 described in the literature.

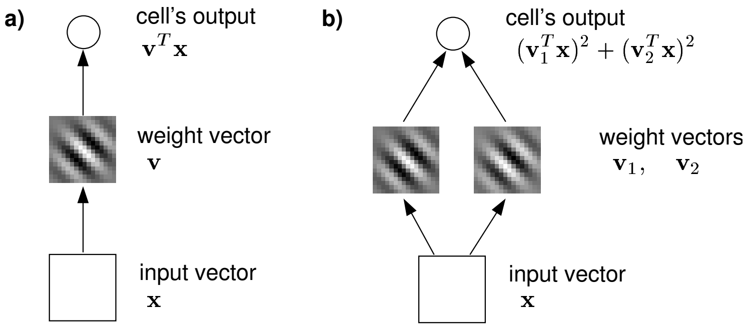


Figure 3.1 Classical models of simple- and complex-cells **(a)** Simple cells respond best to oriented bars at a specific position in the visual field, and are well modeled by a linear Gabor filter [Jones and Palmer, 1987]. **(b)** Complex cells respond to oriented bars but are insensitive to their local position. The classical model (energy model) consists of two linear Gabor filters having the same shape except for a 90° phase difference. The square sum of the response of the two filters yields the output [Adelson and Bergen, 1985]. Orientation, frequency, and size of the subunits in this figure have been fitted to those of the optimal excitatory stimulus of the unit shown in Fig. 3.8a for comparison.

The following section presents the input data set and the methods used to analyze the results. Section 3.3 describes the simulation results and compares the learned functions with neurons reported in the physiological literature. In Section 3.4 we investigate the role of spatial transformations, the statistics of the input images, dimensionality reduction, and asymmetric decorrelation in our results with a set of control experiments. The chapter concludes with a discussion in Section 3.5.

3.2 Methods

3.2.1 Input data

Our data source consisted of 36 gray-valued natural images extracted from the Natural Stimuli Collection of van Hateren (available online at `http://hlab.phys.rug.nl/archive.html`). The images were chosen to contain a variety of natural contents, including trees, flowers, animals, water etc. We avoided highly geometrical human artifacts. The images were preprocessed as suggested in [van Hateren and van der Schaaf, 1998] by block averaging (block size 2×2) and by taking the logarithm of the pixel intensities. This procedure corrects possible calibration problems and reshapes the contrast of the images. After preprocessing, the images were 768×512 pixels large. An extensive discussion of the images and of the preprocessing can be found in [van Hateren and van der Schaaf, 1998].

We constructed image sequences by moving a quadratic window over the images by translation, rotation, and zoom and subsequently rescaling the frames (to compensate for the zoom) to a standard size of 16×16 pixels. The input window was not masked or weighted in any way. The initial position, orientation, and zoom for each sequence were chosen at random. The transformations were performed simultaneously, so that each frame differed from the previous one by position, orientation, and scale. If the window moved out of the image, the sequence was discarded and a new one was started from scratch. Each individual sequence was 100 frames long with a total of 250,000 frames per simulation. (The length of the sequences is irrelevant to the algorithm as long as the total number of input vectors is preserved.) Each image contributed an equal number of frames. Figure 3.2 shows one example sequence. The displacements per frame in horizontal and vertical direction were Gaussian distributed with zero mean and standard deviation 3.56 pixels. The angular speed measured in radians/frame and the magnification difference (defined as the difference between the magnification factor of two successive frames) followed a Gaussian distribution with mean 0 and standard deviation 0.12 and 0.03, respectively. Other simulations showed qualitatively similar results within a reasonable range of parameters, although the distribution of some unit properties might vary. See Control Experiment 1 (Sect. 3.4.1) for a study of the influence of the individual transformations. To include temporal information, the input vectors to SFA were formed by the pixel intensities of two consecutive frames at times t and $t + \Delta t$, so that the second frame in one input vector was equal to the first frame in the next, as illustrated in Figure 3.2. (The time difference Δt was the same used to compute the time derivative.) Note that with two frames as an input, processing in not strictly instantaneous anymore, but slowness can still not be achieved by low-pass filtering.

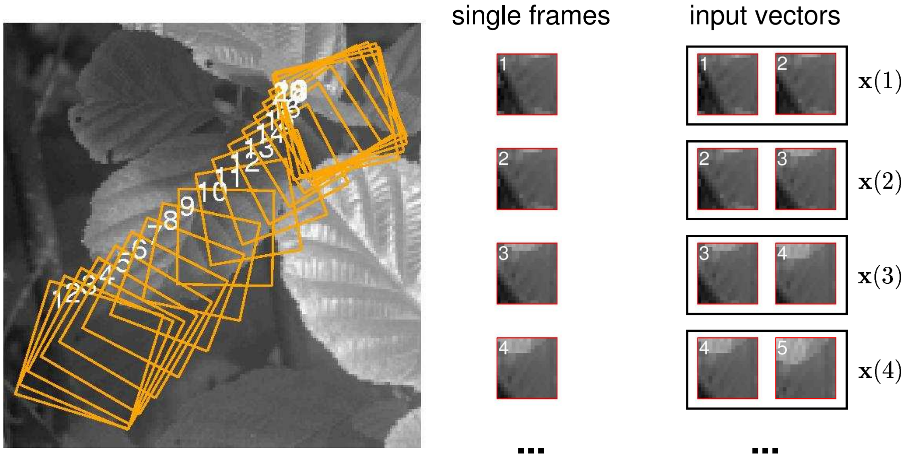


Figure 3.2 Natural image sequences A closeup of one of the natural images used in the simulations (left). The numbered squares show the position, size and orientation of the input window for a short sequence of 20 frames. The content of the window is then rescaled to 16×16 pixels (center). The input to SFA consists of pairs of successive frames (right), to include temporal information.

The function space $\mathcal{F}$ on which SFA is performed is chosen here to be the set of all polynomials of degree 2, as discussed extensively in Section 3.5.2. A run with SFA requires the computation of two large covariance matrices having in the order of $\mathcal{O}(M^2)$ elements, where M is the dimension of the considered function space. In the case of polynomials of degree 2 this corresponds to a number of elements in the order of $\mathcal{O}(N^4)$, where N is the input dimension. Since this is computationally expensive, we performed a standard preprocessing step using principal component analysis (PCA) to reduce the dimensionality of the input vectors from $16 \times 16 \times 2 = 512$ to $N = 100$, capturing 93% of the total variance (see Sect. 3.6.1 for additional remarks). In Control Experiment 3 (Sect. 3.4.3) we present the results of a simulation performed with smaller patches (10×10 pixels) and no dimensionality reduction.

3.2.2 Analysis methods

SFA learns the set of units that applied to our input visual stimuli have the most slowly varying output (which does not imply that processing is slow, see Sect. 2.3.1). They are ordered by slowness (the first one being the slowest) and their outputs are mutually uncorrelated. The units receive pairs of image patches as input vectors and can thus be interpreted as nonlinear spatio-temporal receptive fields and be tested with input stimuli much like in neurophysiological experiments.

Since the sign of a unit's response is arbitrary in the optimization problem, we have chosen it here such that the strongest response to an input vector with a given norm is positive (i.e. such that the magnitude of the response to the optimal excitatory stimulus, $\mathbf{x}^+$, is greater than that to the optimal inhibitory stimulus, $\mathbf{x}^-$; see below). The units can have a spontaneous firing rate, i.e. a non-zero response to a blank visual input. As in physiological experiments we interpret an output lower than the spontaneous one as active inhibition. The absolute value of the spontaneous firing rate is fixed by the zero mean constraint (Eq. 2.14) and has no direct interpretation.

In the simulations presented here the function space $\mathcal{F}$ considered by SFA is the space of polynomials of degree 2. Every polynomial of degree 2 can be written as an inhomogeneous quadratic form $\frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} + c$, where $\mathbf{H}$ is an $N \times N$ matrix, $\mathbf{f}$ is an N-dimensional vector, and c is a constant. For example for $N = 2$

$$
\begin{aligned}
& w_1x_1^2 + w_2x_1x_2 + w_3x_2^2 \quad + \quad w_4x_1 + w_5x_2 \quad + \quad c \\
= {} & \frac{1}{2}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}^T \underbrace{\begin{pmatrix} 2w_1 & w_2 \\ w_2 & 2w_3 \end{pmatrix}}_{\mathbf{H}} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \underbrace{\begin{pmatrix} w_4 \\ w_5 \end{pmatrix}}_{\mathbf{f}}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + c \quad .
\end{aligned}
\tag{3.1}
$$

This class of nonlinear functions has been used in some recent theoretical models and second-order approximations of cortical receptive fields, but their analysis is complicated by a lack of appropriate tools. In the course of this work we developed various methods and algorithms to study quadratic forms in a way similar to receptive fields in physiology. Because of the complexity of the methods and since they are of a more general interest, we present them in a separate chapter (Chap. 4). In the following we will introduce the ones that are relevant for the simulations in this chapter.

In order to analyze the units, we first compute for each of them the optimal excitatory stimulus $\mathbf{x}^+$ and the optimal inhibitory stimulus $\mathbf{x}^-$, i.e. the input that elicits the strongest positive and strongest negative output from the unit, respectively, given a constant norm r of the input vector, i.e. given a fixed energy constraint (Fig. 3.3). We choose r to be the mean norm of the training vectors, since we want $\mathbf{x}^+$ and $\mathbf{x}^-$ to be representative of the typical input. This is in analogy to the physiological practice of characterizing a neuron by the stimulus to which the neuron responds best [e.g. Dayan and Abbott, 2001, Chap. 2.2]. Since in our model we have an explicit definition of the input-output functions of our units, $\mathbf{x}^+$ and $\mathbf{x}^-$ can be computed analytically, as shown in Section 4.4. From the two $\mathbf{x}^+$ patches we compute the size and position of the receptive field and by Fourier analysis the preferred frequency, orientation, speed, and direction of a unit. In some units the preferred parameters for the patch at time t and that at time $t + \Delta t$ are slightly different, in which case we take the mean of the two.

Although the optimal stimuli are very informative they only give a partial view of the behavior of a unit, since these are nonlinear. To gain further insight into the response properties we use an appropriate pair of *test images* (one at time t and one at time $t + \Delta t$) and compute for each unit the corresponding *response image* (Fig. 3.4). The response image is computed by cutting a 16×16 window at each point of the test images, using it as the input to the unit and plotting its output at the corresponding point. To study the response to a range of frequencies and orientations, we use a test image that consists of a circular pattern of sine waves with frequency increasing from the circumference to the center. The frequency increase is logarithmic (i.e. an equal distance along the radius corresponds to an equal frequency difference in octaves) to make the comparison with physiological data easier. We let the ring patterns move outward at the preferred speed of the unit (Fig. 3.4a). These images contain information not only about the whole range of frequencies and orientations to which the unit responds or by which it is inhibited, but also about the sensitivity of the unit to the phase of the grating. If a unit is sensitive, the response image shows os-

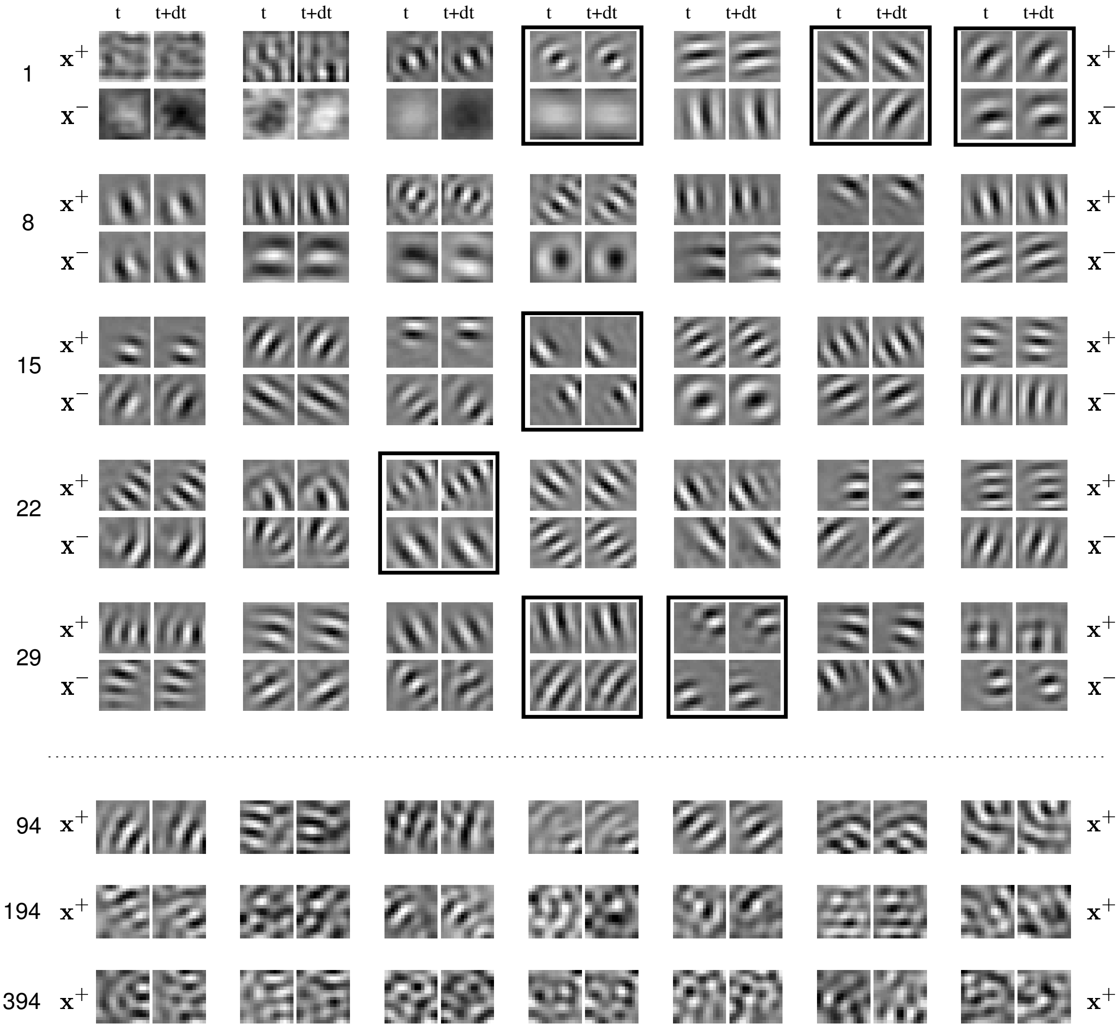


Figure 3.3 Optimal stimuli Top five rows: Optimal excitatory stimuli ($\mathbf{x}^+$) and optimal inhibitory stimuli ($\mathbf{x}^-$) of the first 35 units of the simulation described in the text. For most units $\mathbf{x}^+$ and $\mathbf{x}^-$ look like Gabor wavelets in agreement with physiological data. $\mathbf{x}^+$ gives information about the preferred frequency and orientation of a unit and about the size and position of its receptive field. A comparison between the patches at time t and at time $t + \Delta t$ hints at the temporal structure of the receptive field, e.g. its preferred speed and direction. The units surrounded by a black frame are analyzed in more detail in Figs. 3.8, 3.11, and 3.12. Bottom three rows: Optimal excitatory stimuli for Units 94–100, 194–200, and 394–400. The Gabor-like shape of $\mathbf{x}^+$ begins to degrade around Unit 100, and becomes unstructured for successive units corresponding to functions with quickly varying output. (More optimal stimuli can be found online as indicated in Additional Material.)

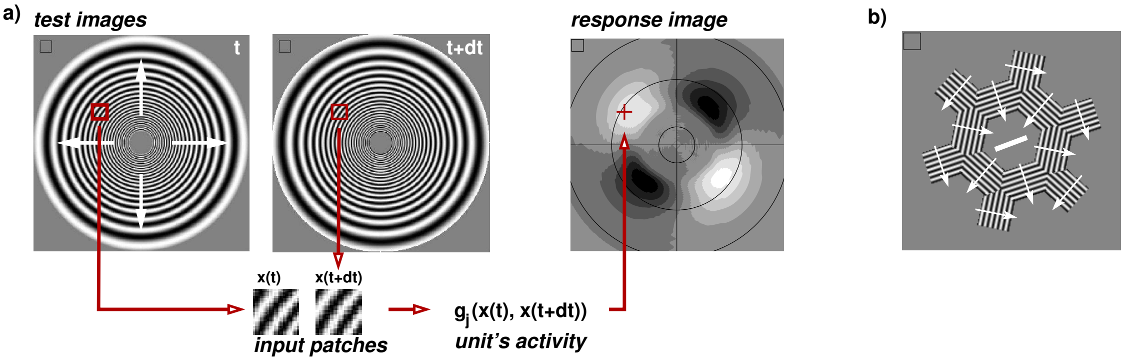


Figure 3.4 Test and response images **(a)** This image illustrates how the response images are computed. Given a test image at time t and at time $t + \Delta t$, at every position two 16×16 input patches are cut out and used as the input to the considered unit. Its output is then plotted at the corresponding point of the response image. Gray values are normalized such that white corresponds to the maximal response and black indicates inhibition. The square at the upper left corner of the response image indicates the size of the input patches. The circular test image shown on the left is used to investigate the response of a unit to a range of frequencies and orientations. The gratings move outward at the preferred speed of the considered unit, as indicated by the white arrows. **(b)** Test image used to investigate end- and side-inhibition. The hexagonal shape is oriented such that it is aligned to the preferred orientation of the considered unit, indicated by the central bar. The gratings are tuned to the preferred frequency and move at the preferred speed of the unit in the direction shown by the thin arrows.

cillations in the radial direction, while if there are no oscillations the unit is phase-invariant. Moreover, at two opposite points of a ring the orientation is equal but the grating is moving in opposite directions, and different responses indicate selectivity to the direction of motion. An illustrative example for the classical simple and complex cell model is shown in Figure 3.5. Since the gratings are curved, an additional factor due to curvature selection might be present in the response images. However, we find that in general this effect is negligible.

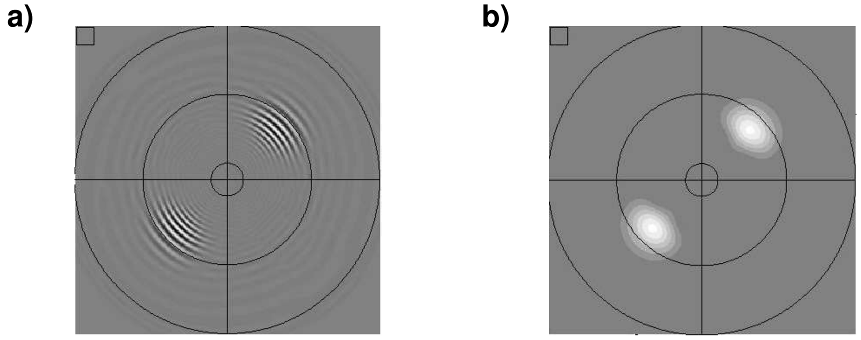


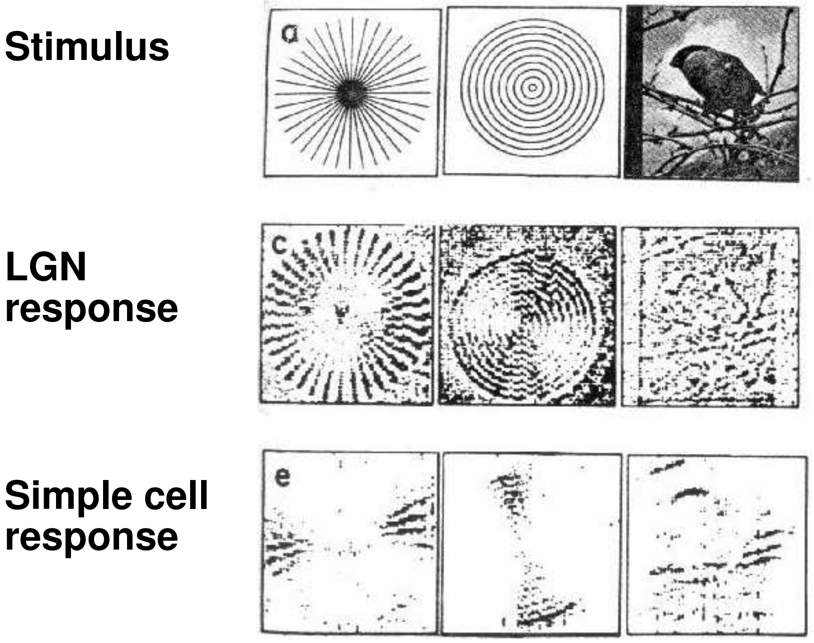
Figure 3.5 Response images for the classical models of simple- and complex-cells (a) Response image for the classical model of simple cells (Fig. 3.1a). The model shows a narrow orientation- and frequency-tuning and oscillations due to phase sensitivity. **(b)** Response image for the classical model of complex cells (Fig. 3.1b). The orientation- and frequency-tuning are the same as in a), but the oscillations have disappeared because the model is phase-insensitive.

The circular response images are similar to the Fourier representation of neural responses used by Ringach et al. [2002] (with the radial axis inverted), but they are more informative in that they contain also information about phase shift behavior and direction selectivity. Experimental readers might be more familiar with orientation and frequency tuning curves. The circular response images contain this information, too (a slice of the response image along the radial direction would give a frequency-tuning curve, while a circular slice would give an orientation-tuning curve), and in addition it shows how the unit behaves at non-optimal parameters.

We use hexagonal-shaped test images (Fig. 3.4b) to investigate end- and side-inhibition. The hexagon is oriented such that two of the branches are aligned to the preferred orientation as indicated by the central bar. The gratings are set to the preferred frequency and move at the preferred speed as shown by the arrows. On the border of the branches of the hexagon the receptive field is only partially filled while in the middle the grating occupies the whole input patch. If the response drops between border and center, the unit is end- or side-inhibited. Of particular interest are the branches at the preferred orientation, on the additional four branches it is possible to see if the inhibition is effective also at other orientations. The central hexagonal part contains angles at various orientations, and is useful to study the curvature selectivity of the units. If the preferred speed is not zero, in the second image there is one junction for each branch where the sine gratings of two branches do not coincide anymore. This might in principle distort the response in those areas. (Note that the hexagonal response image shown in Fig. 3.11b has preferred speed zero, and is thus not affected.)

Test images with geometrical patterns and complex visual scenes have been used to study LGN and V1 neurons by Creutzfeldt and Nothdurft [1978], as shown in Figure 3.6. We are not aware of any successive physiological study that used this technique.

We additionally performed experiments with drifting sine gratings in order to compute various unit



(Creutzfeldt and Nothdurft, 1978, Fig. 3)

Figure 3.6 Response images of neurons in the visual system This image shows the response images of an on-center cell in the LGN (middle) and of a simple cell in V1 (bottom) when presented with two test images with a geometrical pattern (drawings were bright stimuli on dark background) and one visual scene. (Adapted from [Creutzfeldt and Nothdurft, 1978], Fig. 3a, c, and e, with kind permission of Springer Science and Business Media.)

properties and compare them with physiological results. The sine grating parameters were always set to the preferred ones of the considered unit. For example the polar plots of Figure 3.8a–c.3 were generated by presenting to a unit sine gratings at different orientations and with frequency, speed, position, and size fixed to the preferred ones. The plots show the response of the unit normalized by the maximum (radial axis) vs. the orientation of the sine grating (angular direction). Unless stated otherwise, comparisons are always made with experimental data of complex cells only.

3.3 Results

We now describe units obtained in a single simulation and make a comparison with corresponding cells reported in the experimental literature. For each simulation SFA extracts a complete basis of the considered function space ordered by decreasing slowness. In our case this corresponds to 5150 polynomials of degree 2. Of course, the last functions are actually the ones with the most *quickly* varying output signal and will not be considered. We use the mean β-value of the pixel intensities as a reference, since we do not want functions that compute a signal varying more quickly than their input. Figure 3.7 shows the β-values of the first 400 units when tested on training data or on previously unseen sequences with a total of 400,000 frames. The units remain slowly varying also on test data, and their order is largely preserved. At about unit number 100 the shape of the optimal excitatory stimulus begins to degrade (Fig. 3.3, bottom rows) and the β-values come close to that of the input signal ($> 90\%$). For these reasons we consider here only the first 100 functions. Although for illustrative purposes we mainly concentrate on individual units, we also find a good qualitative and quantitative match on population level between the considered units and complex cells in V1. We did not find any unit among the first 100 whose properties were in contradiction with those of neurons in V1.

Gabor-like optimal stimuli and phase invariance: The optimal stimuli of almost all units look like Gabor wavelets (Fig. 3.3) in agreement with physiological data. This means that the units respond best to edge-like stimuli. The response of all these units is largely invariant to phase shift as illustrated by the lack of oscillations in the response images (Figs. 3.8, 3.11, 3.12). To quantify this aspect

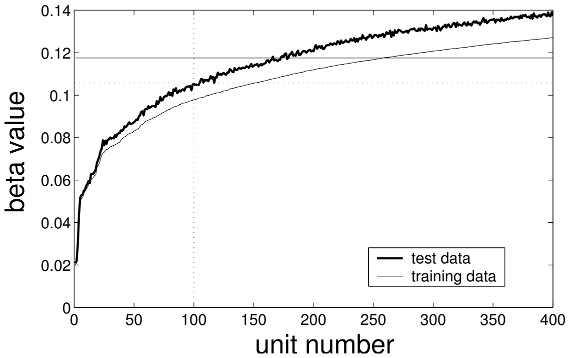


Figure 3.7 Beta values The β-value of the first 400 units when applied to the training data (thin line) or to the test data (novel sequences with a total of 400,000 frames, thick line). The units remain slow and ordered also on test data. The horizontal solid line corresponds to the mean β-value of the input signals. The horizontal dotted line corresponds to the β-value of unit number 100, which is slightly higher than 90% of that of the input signals.

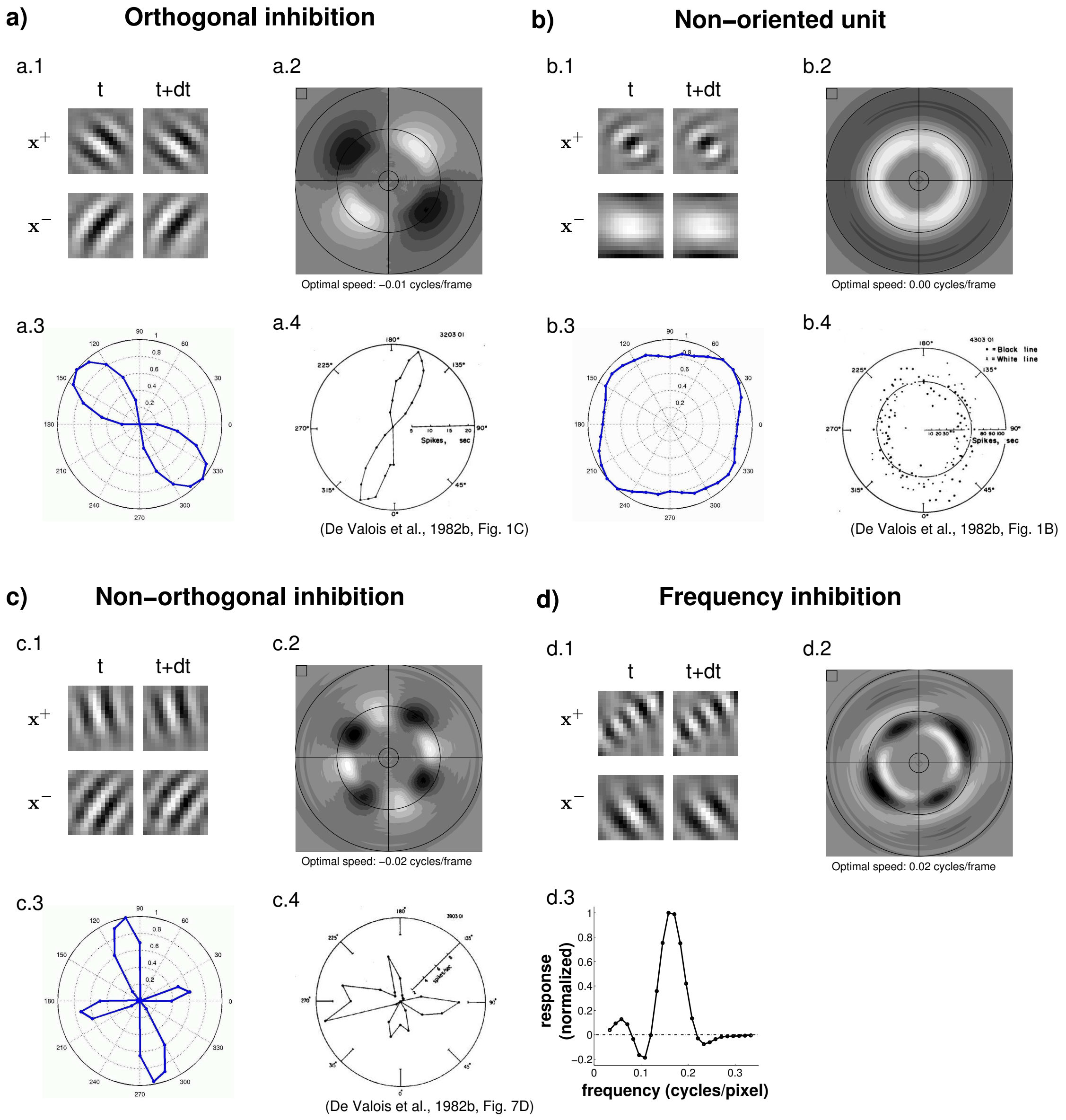
we presented to each unit a sine grating tuned to the preferred orientation, frequency, speed, length, width, and position as revealed by $\mathbf{x}^+$ and computed the relative modulation rate F_1/F_0, i.e. the ratio of the amplitude of the first harmonic to the mean response. Neurons are classified as complex if their F_1/F_0 ratio is less than 1.0, otherwise they are classified as simple, as defined in [Skottun et al., 1991]. All units have a modulation rate considerably smaller then 1.0 (the maximum modulation rate is 0.16) and would thus be classified as complex cells in a physiological experiment. 98 units out of 100 had both a Gabor-like $\mathbf{x}^+$ and phase-shift invariance, the missing two cells are described under the paragraph *Tonic cells*. (See Sect. 3.6.2 for additional remarks.)

Active inhibition (orientation and frequency tuning): In the classical model complex cells have no inhibition and are correspondingly restricted in their functional properties [see e.g. MacLennan, 1991]. In physiological neurons, however, active inhibition is present and useful for example to shape the tuning to orientation and frequency [Sillito, 1975, De Valois et al., 1982b]. (See Sect. 3.6.3 for additional remarks.)

In our model inhibition is present in most units and typically makes an important contribution to the output. As a consequence $\mathbf{x}^-$ is usually well-structured and has the form of a Gabor wavelet as well (Fig. 3.3). Its orientation plays an important role in determining the orientation tuning. It can be orthogonal to that of $\mathbf{x}^+$ (Fig. 3.8a), but it is often not, which results in sharpened orientation tuning (since the response must decrease from a maximum to a minimum in a shorter interval along the orientation). On the other hand, we also find units that would be classified as complex by their response to edges and their phase-invariance but which are not selective for a particular orientation (Fig. 3.8b) (four units out of 100). Similar cells are present in V1 and known as *non-oriented cells* [De Valois et al., 1982b]. Figure 3.9a compares the distribution of the orientation bandwidth of our units with that of complex cells reported in [De Valois et al., 1982b, Gizzi et al., 1990]. Our units seem to have a slightly broader tuning, but the difference is not significant (one-sided Kolmogorov-Smirnov test, $P > 0.8$).

When the orientation of $\mathbf{x}^-$ is very close to that of $\mathbf{x}^+$, it is possible to observe a second peak of activity at an orientation orthogonal to the optimal one (Fig. 3.8c) known as *secondary response*

Figure 3.8 Active inhibition This figure illustrates the ways in which active inhibition can shape the receptive field of a unit. Subfigures (a–d.1) show the optimal excitatory and inhibitory stimuli $\mathbf{x}^+$ and $\mathbf{x}^-$ of the considered units. Subfigures (a–d.2) show the response images corresponding to the circular test image (Fig. 3.4a). The small square in the upper left corner represents the size of an input patch. Subfigures (a–c.3) show a polar plot of the response of the unit to a sine grating. The orientation of the grating varies along the angular direction, while the radial axis measures the response normalized to the maximum. In Subfigures (a–c.4) we show for comparison equivalent plots showing the response in spikes/sec of neurons of the primary visual cortex of the cat [De Valois et al., 1982b]. **(a)** This unit shows maximal inhibition at an orientation orthogonal to the optimal excitatory one while there is no inhibition in frequency. The unit has a relatively broad tuning to both orientation and frequency. (The cell shown in (a.4) has been classified as simple but seems to be representative for complex cells as well.) **(b)** Although this unit responds to edges (as shown by $\mathbf{x}^+$), the polar plot reveals that it is not selective for any particular orientation. The unit is thus classified as non-oriented. There is a slight inhibition at lower frequencies. **(c)** This unit has inhibitory flanks with an orientation near to the preferred one. In such a case it is sometimes possible to observe a second peak of activity appearing at the orthogonal orientation, known in the experimental literature as secondary response lobe. **(d)** $\mathbf{x}^+$ and $\mathbf{x}^-$ of this unit have the same orientation but a different frequency. This results in a very sharp frequency tuning. On the other hand the unit responds to a broad range of orientations.



a) Orthogonal inhibition

b) Non−oriented unit

c) Non−orthogonal inhibition

d) Frequency inhibition

Optimal speed: −0.01 cycles/frame

Optimal speed: 0.00 cycles/frame

Optimal speed: −0.02 cycles/frame

Optimal speed: 0.02 cycles/frame

(De Valois et al., 1982b, Fig. 1C)

(De Valois et al., 1982b, Fig. 1B)

(De Valois et al., 1982b, Fig. 7D)

(caption on the previous page)

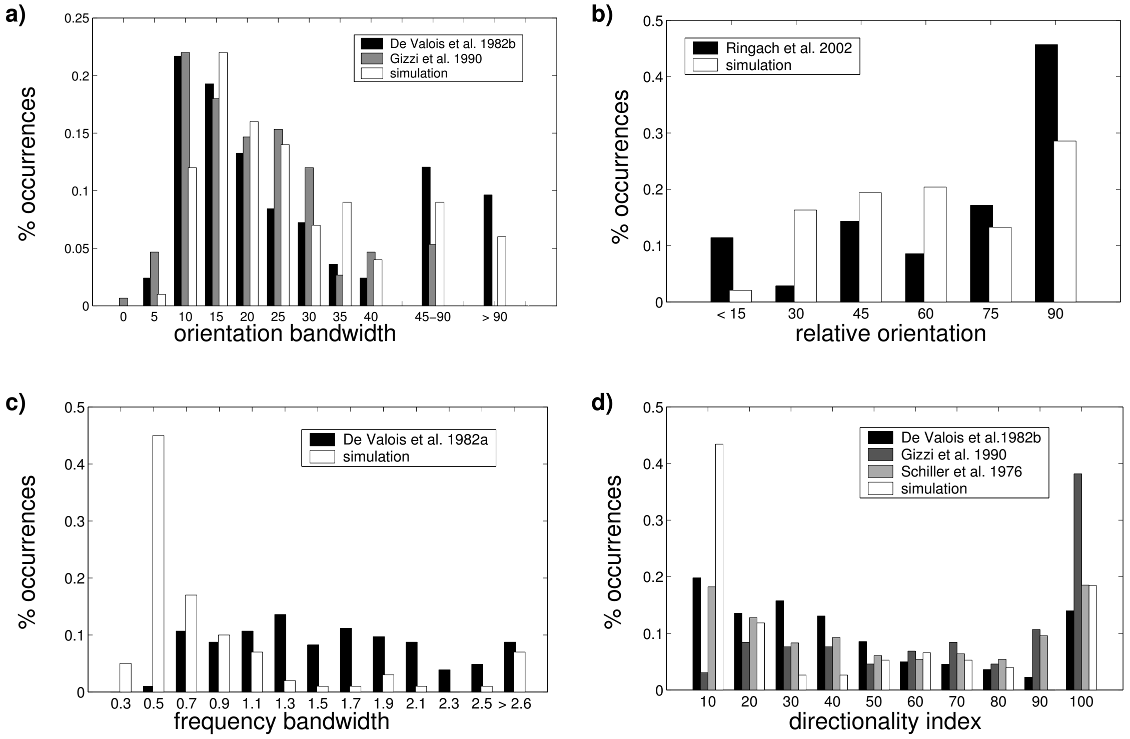


Figure 3.9 Population statistics **(a)** Distribution or half-height orientation bandwidth in complex cells and in our simulation. **(b)** Distribution of the angle between the orientation of maximal excitation and maximal inhibition. The data from [Ringach et al., 2002] contains simple cells as well as complex cells, which might explain the more pronounced peak at 90°. **(c)** Distribution of half-height frequency bandwidth in complex cells and in our simulation. The bandwidth is measured by the units' contrast sensitivity function as in [De Valois et al., 1982a]. Units whose contrast sensitivity does not drop below 50% of the maximum are classified in the last bin. **(d)** Distribution of the directionality index of complex cells and in our simulation (the data from [De Valois et al., 1982b] also contain simple cells, but in the paper it is stated that there was no significant difference between the two populations).

lobe. 9 units out of 100 had this characteristic. (We classify a unit as having a secondary response lobe if on the orientation tuning curve its output expressed in percentage of the maximal response decreases from 100% to less than 10% and then it increases again to more than 50% at the orthogonal orientation.) De Valois et al. [1982b] repeatedly stress that non-orthogonal inhibition is common among neurons in V1 and show some example cells (one of which is shown in Fig. 3.8c.4). The fraction of such cells in V1 is not reported. Ringach et al. [2002] measured the relative angle between maximal excitation and suppression. Figure 3.9b shows the histograms of the angle between maximal excitation and inhibition in our simulation and in their study. In the latter there is a more pronounced peak at orthogonal orientations. This difference might be due to the small fraction of complex cells in the population considered in the paper (24 complex cells out of 75). To draw the histogram the cells that responded to very low frequencies were discarded. It was not specified how many complex cells remained in the final set). To the extent simple cells are well described by linear functions, they are likely to show maximal suppression at 90°.

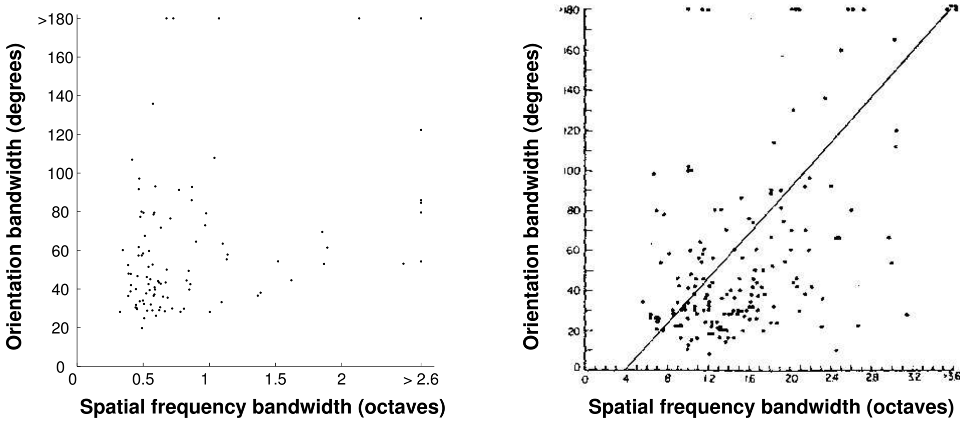
Analogous considerations also hold for the frequency domain: here again the tuning varies from a sustained response within a range of frequencies (Fig. 3.8a) to a sharp tuning due to active inhibition (Fig. 3.8d). Figure 3.9c shows the distribution of frequency bandwidth in octaves in our simulation and in complex cells as reported by De Valois et al. [1982a]. The bandwidth was computed from the units' contrast sensitivity, like in the cited paper. Complex cells have a rather flat distribution, while the bandwidth of our units is concentrated between 0.3 and 1.4 octaves with a peak at 0.5 . The reason for this difference is not yet entirely clear (but see Sect. 3.6.4).

Figure 3.10 shows the joint distribution of frequency and orientation bandwidth in V1 in our simulation and as reported by De Valois et al. [1982a]. Since the marginal distribution of frequency bandwidth is different, in our case the data points are more concentrated in the left part of the graph. However, the two distributions are similar in that they have a large scatter and no strong correlation between orientation and frequency bandwidth. (The data from [De Valois et al., 1982a] also contains simple cells. The distribution of frequency and orientation bandwidth was found to be similar in simple and in complex cells [De Valois et al., 1982a,b], but the correlation of the two variables within the two groups is not reported.)

End- and Side-Inhibition: Some of the complex cells in V1 are selective for the length (*end-inhibited* cells) or width (*side-inhibited* cells) of their input. While in normal cells the extension of a grating at the preferred orientation and frequency produces an increase of the response up to a saturation level, in these cells the response drops if the grating extends beyond a certain limit [DeAngelis et al., 1994].

End- and side-inhibition are present also in our simulation (Fig. 3.11). We computed for each unit a quantitative measure of its degree of end- or side-inhibition by presenting sine gratings of different length and width (keeping all other parameters equal to the preferred ones). We define the end- and side-inhibition index as in [DeAngelis et al., 1994] by the decrease of the response in percent between optimal and asymptotic length and width, respectively. 10 units out of 100 had an end-inhibition index greater than 20%, and 7 units out of 100 had a side-inhibition index greater than 20%. In contrast to [DeAngelis et al., 1994] we found only 2 units that showed large ($> 20\%$) end- and side-inhibition simultaneously.

End- and side-inhibited units can sometimes be identified by looking directly at the optimal stimuli. In these cases $\mathbf{x}^+$ fills only one half of the window while the missing half is covered by $\mathbf{x}^-$ with the same orientation and frequency (Figs. 3.11a.1,b.1). In this way, if we extend the stimulus into the missing half, the output receives an inhibitory contribution and drops. This receptive field organization is compatible with that observed in V1 by Walker et al. [1999] in that inhibition is asymmetric and is tuned to the same orientation and frequency as the excitatory part.



(De Valois et al., 1982a, Fig. 8)

Figure 3.10 Frequency and orientation bandwidth This figure compares the joint distribution of frequency and orientation bandwidth in our simulation (left) and in [De Valois et al., 1982a] (right). The two distributions are similar in that they have a large scatter and no strong correlation between orientation and frequency bandwidth. (The data set on the right contains simple and complex cells. The distribution of orientation bandwidth was found to be similar for both classes, but the correlation of the two variables within the two groups is not reported.)

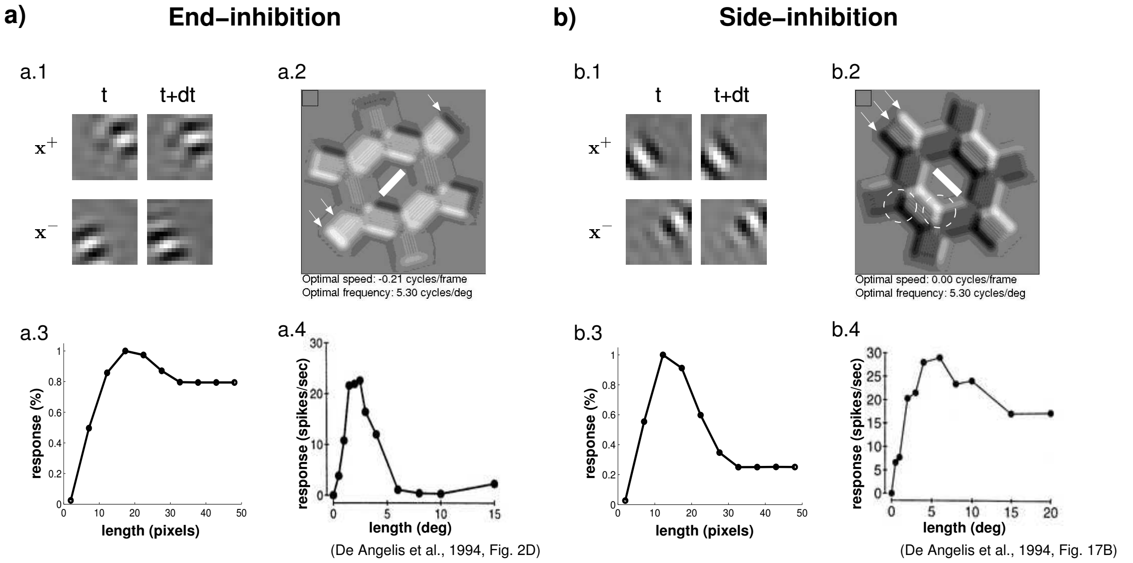
A secondary characteristic of end- and side-inhibited cells in V1 is that they are sometimes selective for different signs of curvature [Dobbins et al., 1987, Versavel et al., 1990]. This can be observed in our simulation, for example in Figure 3.11b.2: the dashed circles indicate two opposite curvatures. One of them causes the unit to respond strongly while the other one inhibits it.

Direction selectivity: Complex cells in V1 are sensitive to the motion of the presented stimuli. Some of them respond to motion in both directions while others are direction selective [Hubel and Wiesel, 1962, Schiller et al., 1976a, De Valois et al., 1982b, Gizzi et al., 1990]. Similarly, in our model some units are strongly selective for direction (Fig. 3.12) while others are neutral. In the latter case the optimal speed may be non-zero but the response is nearly equal for both directions.

We measure direction selectivity by the *directionality index* given by $DI = (1 - R_{np}/R_p) \cdot 100$ with R_p and R_{np} being the response in the preferred and in the non-preferred direction, respectively [Gizzi et al., 1990]. The index is 0 for bidirectional units and 100 for units that respond only in one direction of motion. Figure 3.9d shows the histogram of DI in our simulation compared to three distributions from the physiological literature. The distributions are quite similar, although there is a more pronounced peak for bidirectional units. (See Sect. 3.6.5 for additional remarks.)

Tonic cells: The first unit in every simulation codes for the mean pixel intensity and is thus comparable to the *tonic cells* found in V1 [Schiller et al., 1976a]. Tonic cells respond to either bright or dark stimuli but do not need a contour in order to respond. We find in addition one unit (the second one) that responds to the squared mean pixel intensity and therefore to both bright and dark stimuli. (See Sect. 3.6.6 for additional remarks.)

Response to complex stimuli: As can be inferred from the invariances (Sect. 4.5), some units give a near-optimal output ($> 80\%$ of the optimum) in response to corners and T-shaped stimuli and are related



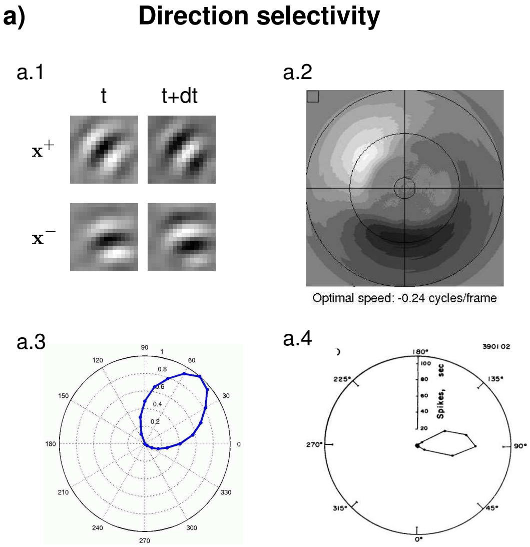
Optimal speed: -0.21 cycles/frame
Optimal frequency: 5.30 cycles/deg

Optimal speed: 0.00 cycles/frame
Optimal frequency: 5.30 cycles/deg

(De Angelis et al., 1994, Fig. 2D)

(De Angelis et al., 1994, Fig. 17B)

Figure 3.11 End and side inhibition This figure illustrates end- and side-inhibition in our simulation. Subfigures (a–b.1) show the optimal stimuli $\mathbf{x}^+$ and $\mathbf{x}^-$ of the considered units. Subfigures (a–b.2) show the response image corresponding to a hexagonal test image (Fig. 3.4b). The small square in the upper left corner represents the size of an input patch. Subfigures (a–b.3) show the response of the unit to a sine grating with varying length or width, respectively. For comparison, equivalent plots of the response in spikes/sec of end- and side-inhibited complex cells published in [DeAngelis et al., 1994] are shown in Subfigures (a–b.4). The four curves (a–b.3, a–b.4) have similar shapes and mainly differ in their inhibition index (the ratio between maximal and asymptotic response), which has a broad distribution in all four cases. **(a)** This unit is end-inhibited. The optimal stimuli indicate how the receptive field is organized: the optimal excitatory stimulus only fills one half of $\mathbf{x}^+$while the missing half is in $\mathbf{x}^-$. Inhibition is thus asymmetric and tuned to the same orientation and frequency as excitation in agreement with [Walker et al., 1999]. Going from left to right, the first arrow in the hexagonal response image indicates the point of maximal response, when only the right part of the input window is filled. In the region indicated by the second arrow the grating fills the whole input window; the response is decreased, which corresponds to end-inhibition. The third arrow indicates the region where only the left part of the input window is filled, and the unit is inhibited. **(b)** A side-inhibited unit. The interpretation of $\mathbf{x}^+$ and $\mathbf{x}^-$ and of the arrows is similar to that in a), except that the grating extends laterally, which corresponds to side-inhibition. The two dashed circles surround two regions with opposite curvature. The unit responds strongly in one case and is inhibited in the other, which indicates curvature selectivity.



(De Valois et al., 1982b, Fig. 1D)

Figure 3.12 Direction selectivity This figure shows a direction selective unit. See the caption of Fig. 3.8 for the description of the subfigures. The two wavelets in $\mathbf{x}^{+}$ at time t and at time $t + \Delta t$ are identical except for a phase shift. This means that the unit responds best to a moving edge. This is confirmed by the response image, which shows different responses at opposite points of a ring. At those points the orientation is equal but the grating is moving in opposite directions.

to the V1-cells described in [Shevelev, 1998]. In a physiological experiment these stimuli could be classified as the optimal ones if the contrast instead of the energy is kept constant (for example, a T-shaped stimulus has a larger energy than a bar with the same length and contrast). Other units respond to one sign of curvature only. These behaviors are often associated with end- or side-inhibition, as described above. In a few cases the two wavelets in the optimal stimuli have a slightly different orientation or frequency at time t and at time $t + \Delta t$, which indicates a more complex behavior in time, such as selectivity for rotation or zoom.

Relations between slowness and behavior: Although the precise shape and order of the units can vary in different simulations, there seem to be relations between the slowness of unit responses and the receptive field properties. The slowest units are usually less selective for orientation and frequency, have orthogonal inhibition, and their preferred speed is near zero. Units with non-orthogonal inhibition, direction selectivity, and end- or side-inhibition predominate in a faster regime. It would be interesting to see if similar relations can also be found experimentally by comparing the temporal variation of the response of a neuron stimulated by natural scenes and its receptive field properties. We could not find any relation between preferred orientation and slowness.

3.4 Control experiments

We performed a set of control experiments in order to assess the role of spatial transformations, the statistics of the input images, dimensionality reduction, and asymmetric decorrelation in our results.

3.4.1 Control Experiment 1

In this first set of simulations we investigated the role of the three spatial transformations used in our simulation: translation, rotation, and zoom. The settings for these experiments are identical to those of the main simulation described in Section 3.3 except for the fact that to achieve a reasonable total simulation time the input vectors consisted of single frames (instead of pairs of consecutive frame). The input dimensionality is reduced accordingly to $N = 50$, so that the proportion between input and reduced dimensions is the same as in the main simulation. We analyzed the first 50 units for each experiment. We performed a first simulation to be used as a reference using all three spatial transformations, followed by six simulations where only one or two transformations where used: translation only, rotation only, zoom only, translation and rotation, translation and zoom, rotation and zoom. The parameters used for the transformations are the same as in the main simulation.

Figure 3.13a–h shows the optimal excitatory stimuli of all seven simulations. It can be seen that optimal stimuli similar to Gabor wavelets appear only in simulations with translation, including the one with only translation. Sine grating experiments show that all units have phase shift invariance (all relative modulation rates are smaller than 0.27). Translation is thus a necessary and sufficient spatial transformation to obtain complex cell receptive fields from natural images with SFA. On the other hand, the optimal stimuli in the simulation with only translation seem to occupy the whole patch in contrast to the more localized optimal stimuli in the simulations where translation is combined with the other transformations. Zoom and especially rotation are necessary to obtain more localized receptive fields. (This is not directly evident from Fig. 3.13 because of the small number of optimal stimuli shown. Images containing more optimal stimuli can be found online; see Additional Material.) In simulations including translation but no rotation, the distribution of orientation bandwidth is skewed towards small bandwidths. High bandwidths therefore seem to be a consequence of the amount of rotation included in the simulation, as one would expect since it results in an improved tolerance to changes in orientation.

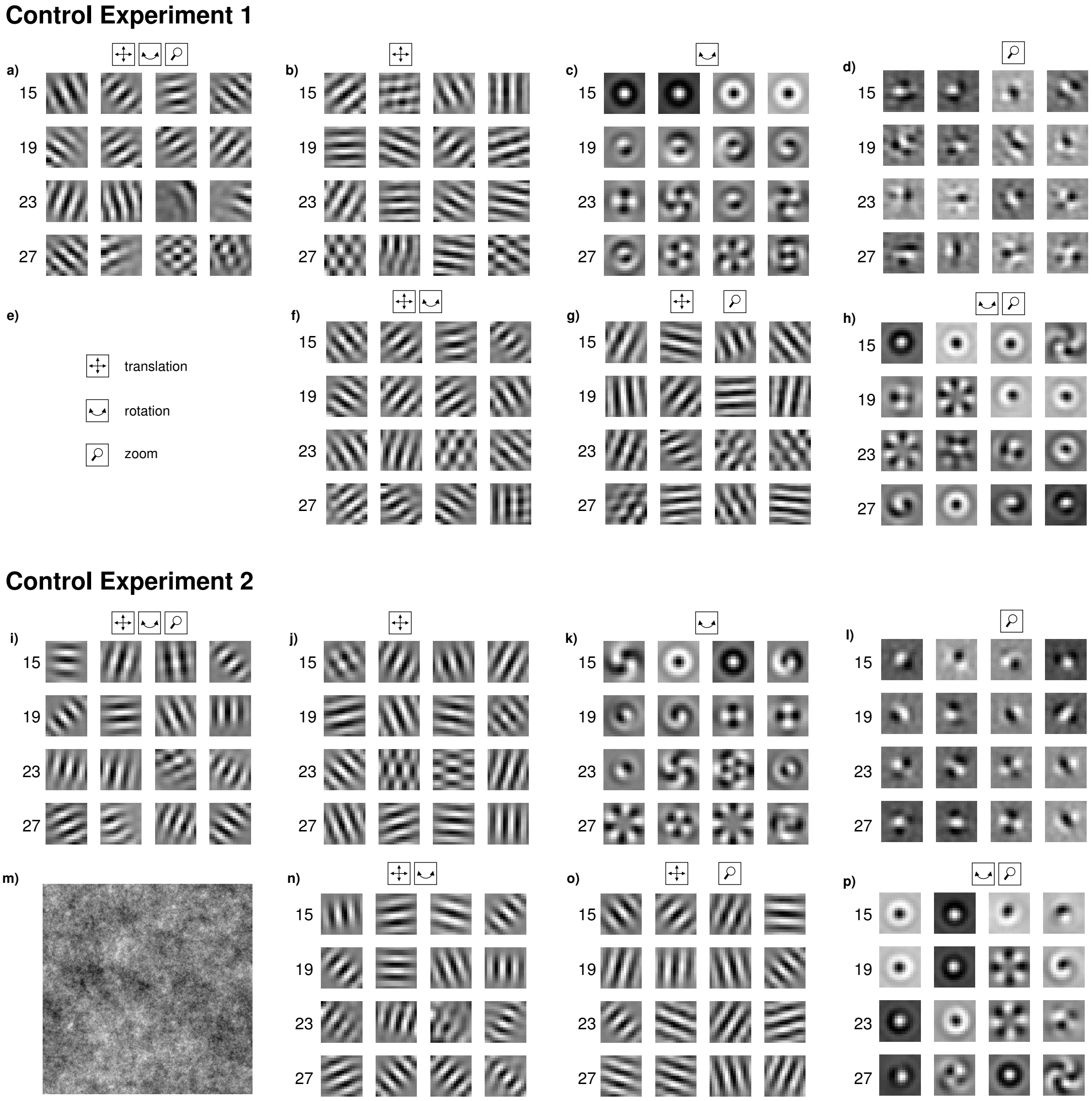


Figure 3.13 Control Experiments 1 and 2 This figure shows the optimal excitatory stimuli of Units 15–30 for Control Experiment 1 and 2. One or more icons representing the spatial transformations applied in a particular experiment are displayed on the top of each plot. A legend for the icons can be found in (e). **(a–h)** Optimal excitatory stimuli for Control Experiment 1. In this set of experiments we investigated the role of the three spatial transformations used in our simulation. The input image sequences were constructed from natural images, like in the main simulation, and all combinations of one, two, or three spatial transformations were applied. The results show that translation is a necessary and sufficient spatial transformation to obtain complex cell characteristics. **(i–p)** Optimal excitatory stimuli for Control Experiment 2. In this set of experiments we investigated the role of the spatial statistics of natural images in our results. The input images were replaced by colored noise images with a power spectrum equal to that of natural images. The results suggest that spatial second order statistics are sufficient to obtain complex cell characteristics.

Functions learned with rotation only and with both rotation and zoom show optimal stimuli with a circular structure [cf. Kohonen et al., 1997]. Functions learned with zoom only have optimal stimuli with small white/black spots in the center of the receptive field. These two last receptive field structures are not found in V1.

3.4.2 Control Experiment 2

In this set of experiments we investigated the role of the spatial statistics of natural images in our results. The settings are identical to those of Control Experiment 1 except that instead of natural images we used colored noise images with a $1/f^2$ power spectrum, similar to the one shown in Figure 3.13m. The statistical properties of such images are equivalent to those of natural images up to the second order [Ruderman and Bialek, 1994, Dong and Atick, 1995b]. For each experiment we generated 36 new noise images that replaced the natural ones. The results of these experiments were almost identical to those of Control Experiment 1. The experiments including translation show Gabor-like optimal stimuli (Fig. 3.13i–p) and phase shift invariance. All relative modulation rates are smaller than 0.14 except for two units (one in the experiment with all transformations and one in the experiment with translation and rotation), which have a modulation rate of 1.47 and 1.53, respectively. The distributions of the various parameters are very similar to those obtained in Control Experiment 1. This suggests that spatial second order statistics are sufficient to learn complex cell receptive fields. In principle our model considers spatial statistics up to the fourth order, since SFA uses the matrices $\mathbf{A}$ and $\mathbf{B}$ (Eq. 2.38), that contain products of monomials of degree 2.

3.4.3 Control Experiment 3

This experiment was performed in order to exclude an influence of the dimensionality reduction on our results. (See also Sect. 3.6.1.) The settings of the simulation are identical to those used in the main simulation except that the input vectors consisted in single frames only and, most importantly, that the input patches were 10×10 pixels large and no dimensionality reduction was performed. The translation speed was reduced by $10/16$th, so that the proportion with the patch size was preserved. The first 100 units were analyzed.

The first two units were classified as tonic units. All other units have Gabor-like optimal stimuli (Fig. 3.14) and phase shift invariance (maximum modulation rate 0.23) excluding Unit 9 and 10 which are described below. Units 4 and 11 have checkerboard-like optimal excitatory stimuli. Further analysis reveals that they are non-oriented units that only respond to very high frequencies. Units 9 and 10 have optimal excitatory stimuli with one bright region in a corner. The optimal inhibitory stimuli are similar, but their bright corner is opposite to the excitatory one. The role of these units is unclear, but it is possible that they give a nonlinear response to a luminance gradient along the diagonal.

This experiment shows that the learning of complex cells receptive fields is not a consequence of the dimensionality reduction step.

3.4.4 Control Experiment 4

In our mathematical formulation of the slowness principle the units are learned “one after the other” (Constraint 2.16) in the sense that in an online implementation of SFA the units would be learned using an asymmetric decorrelation term, i.e. unit j would be adapted to optimize the slowness objective (2.13) and to be decorrelated to all units i with $i < j$. In a biological system it might seem more realistic to use symmetric decorrelation, where each unit is adapted to optimize (2.13) and to be decorrelated to all other units.

In this control experiment we relax Constraint (2.16) and mix the units of the main simulation by an orthonormal transformation. The resulting units still satisfy Constraints (2.14–2.16) and span the slowest

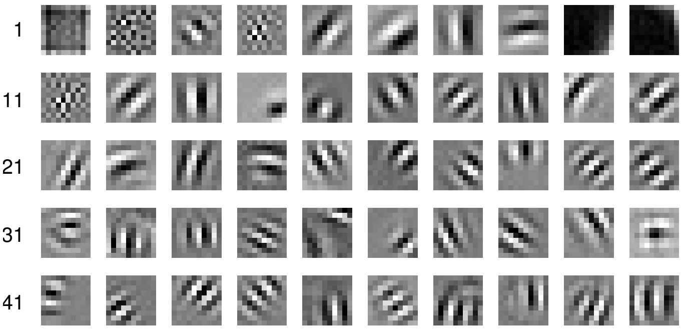


Figure 3.14 Control Experiment 3 This figure shows the optimal excitatory stimuli of Units 1–50 for Control Experiment 3. This experiment was performed without dimensionality reduction in order to exclude a possible influence on the main simulation's results. The results show that the learning of complex cell receptive fields is not a consequence of the dimensionality reduction step.

subspace, i.e. they minimize the *total* variance of the derivative in a 100-dimensional subspace. However, the asymmetry that is inherent in the algorithm and induces the order is no longer effective, so that none of the units is distinguished over the others anymore.

All resulting units have Gabor-like optimal stimuli and phase-shift invariance (maximum modulation rate 0.37) and would thus be classified as complex cells. However, their response images are more unstructured than in the main simulation and they sometimes show a few peaks at different orientations and frequencies, which is not consistent with the behavior of complex cells (see for example the plots reported in [Ringach et al., 2002]). Moreover, the distribution of orientation bandwidth is skewed towards small bandwidths, and in the distribution of the relative angle between excitation and suppression the peak at 90 degree is missing and there is a maximum at 45 degree. It thus seems that asymmetric decorrelation is necessary to obtain the more structured results found in the main simulation. Note, however, that every breaking in the symmetry of decorrelation would make the units converge to the asymmetric solution. Since perfect symmetry is difficult to enforce in a biological system, the asymmetric case might be more realistic.

3.5 Discussion

In this chapter we have shown that SFA applied to natural image sequences learns a set of functions that has a good qualitative and quantitative match with the population of complex cells in V1. In the following section we discuss other theoretical models of complex cells. In Section 3.5.2 we discuss the properties of the chosen function space, present a neural network architecture equivalent to a given polynomial of degree 2, and compare it with the neural networks used in other studies. We conclude with some remarks about other learning rules.

3.5.1 Other theoretical studies

Several theoretical studies have successfully reproduced the basic properties of simple cells [Olshausen and Field, 1996, Bell and Sejnowski, 1997, Hoyer and Hyvärinen, 2000, Szatmáry and Lõrincz, 2001, Olshausen, 2002, Einhäuser et al., 2002, Hurri and Hyvärinen, 2003a] or complex cells [Hyvärinen and Hoyer, 2000, Körding et al., 2004] in models based on the computational principles *sparseness* [Olshausen and Field, 1996, Olshausen, 2002], *statistical independence* [Bell and Sejnowski, 1997, Hoyer and Hyvärinen, 2000, Hyvärinen and Hoyer, 2000, Szatmáry and Lõrincz, 2001], or *slowness* [Einhäuser et al., 2002, Hurri and Hyvärinen, 2003a, Körding et al., 2004]. Among the simple cell models, two included direction selectivity [Olshausen, 2002, Szatmáry and Lõrincz, 2001], two color-selectivity [Hoyer and Hyvärinen, 2000, Einhäuser et al., 2002], and one disparity [Hoyer and Hyvärinen, 2000]. Most of these models focused on one particular aspect of the behavior of cells in V1. In particular the two complex cell models [Hyvärinen and Hoyer, 2000, Körding et al., 2004] learned units that were equivalent to the classical model and thus reproduced only the Gabor-like receptive fields and the phase shift invariance. One important limitation of these models was that they assumed linear or nonlinear but simple neural network architectures that belong to a function set much smaller than the one we consider (see Sect. 3.5.2). None of the nonlinear models included inhibition while many of the illustrated complex cell behaviors are impossible to obtain without it.

Hashimoto [2003] learned quadratic forms (without the linear term) from natural image sequences using three computational principles: independent component analysis (ICA), a gradient descent variant of SFA, and an objective function that maximizes the sparseness of the derivatives of the output. In the experiments performed using ICA she obtained a set of units corresponding to the squared output of simple cells. The results obtained with the gradient descent variant of SFA showed a few units with complex cell properties while most of them were not structured. Although it is difficult to make a direct comparison since only the largest eigenvectors of two of the quadratic forms are reported, these results are in contradiction with the ones presented here. It is possible that the size of the input patches (8×8 pixel) was too small in comparison with the transformations of the image sequences that were used, so that two consecutive frames would have had almost no correlation. It is also possible that the gradient descent converged to a local minimum. In this respect it would be interesting to compare the β-value of the quadratic forms. The experiments performed by maximizing the sparseness of the derivatives learned some units with complex cell properties (including a few with structured inhibition) and others with the characteristics of squared simple cells. These results seemed in general more structured than the ones obtained with the SFA variant. It would be interesting to explore the relation between these two objective functions further.

The studies mentioned up to now learned visual receptive fields directly from the pixel intensities of natural images or movies. Zetzsche and Röhrbein [2001] on the other hand considered as an input the response of a set of artificial simple cells (i.e. linear Gabor wavelets, whose outputs were split into an ON (positive) and an OFF (negative) pathway by half-wave rectification). The two pathways were then used as an input to PCA or to ICA. The main result of the paper is that PCA applied to the output of Gabor wavelets having the same orientation and frequency but different positions in the visual field learns units with simple- and others with complex-cell characteristics. Additionally, some units of both classes showed end-inhibition. Another experiment performed by applying ICA to the output of Gabor wavelets with same orientation, different positions, and even- and odd-symmetric filter properties produced simple cells, some of which were end-inhibited and others side-inhibited. It is known that the Gabor wavelets used to form the first layer can be learned directly from natural images, for example by ICA, so that these results could in principle be obtained directly from pixel intensities. In this case, however, an additional criterion should be provided to group the resulting wavelets, so that only the ones with equal orientation and frequency are connected to a second-layer unit.

To our knowledge the model presented here is the first one based directly on input images that is able to learn a population of units with a rich repertoire of complex cell properties, such as active inhibition, secondary response lobes, end-inhibition, side-inhibition, direction selectivity, tonic cell behavior, and curvature selectivity.

3.5.2 Function space and equivalent network architecture

We performed SFA on the function space $\mathcal{F}$ of all polynomials of degree 2 mainly because of limited computational resources. In principle one would like to consider a function space as large as possible. On the other hand, neurons have computational constraints, too, and thus considering too large a set could lead to unrealistic results. This leads to an interesting question: Which functions of its input can a neuron compute? In other words, in which function space does the input-output map of a neuron lie?

Lau et al. [2002] have fitted the weights of a nonlinear 2-layer neural network to the output of complex cells. They found that the relation between the linear output of the subunits and the output of the complex cell is approximately quadratic (mean exponent 2.3 ± 1.1). This result describes which functions the neurons do compute and not which ones it could compute, which would determine the function space. It suggests, however, that considering the space of polynomials of degree 2 might be sufficient. Kayser et al. [2003] adapted the weights and the exponents of a neural network similar to the classical model of complex cells using an objective function based on the slowness principle (but see Sect. 3.5.3 for some remarks regarding the definition of slowness). The exponent of most of the units converged to 2. This experiment also suggests that quadratic nonlinearities might be an appropriate choice in our context. Polynomials of degree 2 also correspond to a Volterra expansion up to the second order of the spatio-temporal receptive field of a neuron [see e.g. Dayan and Abbott, 2001, Sect. 2.2] when time is discretized in small steps. Such an approximation has been used with some success to describe complex cells [e.g. Touryan et al., 2002].

As shown later in Section 4.9, each polynomial of degree 2 can be written as a two layer neural network (Fig 3.15a). The first layer is formed by a set of N linear subunits $s_k(\mathbf{x}) = \mathbf{v}_k^T \mathbf{x}$ followed by a quadratic nonlinearity weighted by some coefficients $\mu_k/2$ ($\mathbf{v}_k$ and μ_k are the eigenvectors and eigenvalues of the quadratic term $\mathbf{H}$ of the corresponding quadratic form, cf. Sect. 3.2.2). The output neuron sums the contribution of all subunits plus the output of a direct linear connection from the input layer. The coefficient can be negative, so that some of the subunits give an inhibitory contribution to the output. The equivalent neural network clarifies the relation between the general case of polynomials of degree 2 (Fig. 3.15a) and the smaller neural networks used in in standard studies in the field [Hyvärinen and Hoyer, 2000, Hyvärinen and Hoyer, 2001, Körding et al., 2004, see also Sect. 3.5.1]. Those models usually rely either on linear networks, which lie in the space of polynomials of degree 1, or on networks with one layer of 2 to 25 linear units followed by a quadratic nonlinearity (Fig. 3.15b), which form a small subset of the space of polynomials of degree 2. These networks differ from polynomials of degree 2 in that they lack a direct linear contribution to the output and have fewer subunits: in our study each learned function has $N = 100$ subunits, which is much larger than the fixed numbers used in the studies mentioned above. The most important difference, however, is related to the normalization of the weights. In the theoretical studies cited above the weights are normalized to a fixed norm and the activity of the subunits is not weighted. In particular, since there are no negative coefficients, no inhibition is possible.

The equivalent neural network shows that the choice of the space of all polynomials of degree 2 is compatible with the hierarchical organization of the visual cortex first proposed by Hubel and Wiesel [Hubel and Wiesel, 1962] in the sense that every learned function can be implemented by a hierarchical model similar to the energy model (Fig. 3.1b). The additional excitatory or inhibitory subunits might introduce additional complex cell invariances, broaden or sharpen the orientation and frequency tuning, and provide end- or side-inhibition. The learning of the linear subunits would be modulated by the application of the slowness principle to the complex cell. According to this interpretation, the subunits would correspond to simple cells, and their receptive fields should thus look like Gabor wavelets. However, the neural network architecture is not unique and the subunits can assume different forms many of which might not be similar to simple cells (Sect. 4.9 and Fig. 4.11).

A possible alternative would be that simple cells are learned by a parallel computational principle and then grouped and weighted by the slowness principle in order to form complex cells. A similar distinct grouping step has been used in [Zetzsche and Röhrbein, 2001] and [Hurri and Hyvärinen, 2003b]. Computational principles that have led to simple cells are sparseness, statistical independence, and slowness (see Sect. 3.5.1).

Although the function space of polynomials of degree 2 is mathematically attractive and has proved

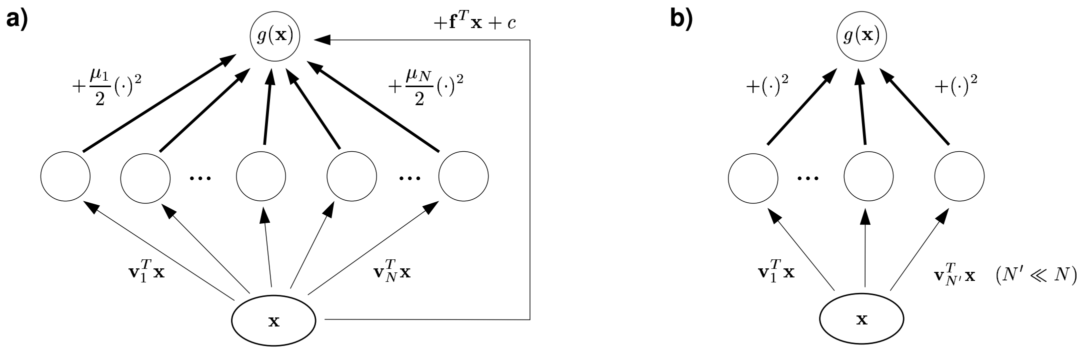

a)
$g(\mathbf{x})$
$+\mathbf{f}^T\mathbf{x}+c$
$+\frac{\mu_1}{2}(\cdot)^2$
$+\frac{\mu_N}{2}(\cdot)^2$
...
$\mathbf{v}_1^T\mathbf{x}$
$\mathbf{v}_N^T\mathbf{x}$
$\mathbf{x}$
b)
$g(\mathbf{x})$
$+(\cdot)^2$
$+(\cdot)^2$
...
$\mathbf{v}_1^T\mathbf{x}$
$\mathbf{v}_{N'}^T\mathbf{x}$ $(N' \ll N)$
$\mathbf{x}$

Figure 3.15 Equivalent neural network In the plots we assume that the norm of the subunits is normalized to 1, i.e. $\|\mathbf{v}_i\| = 1$. The ellipse in the input layer represents a multidimensional input. **(a)** Neural network architecture equivalent to a polynomial of degree 2. The first layer consists of linear subunits whose outputs are squared and weighted. The output neuron on the second layer sums the contribution of all subunits and the output of a direct linear connection from the input layer (cf. Sect. 4.9). **(b)** Simpler neural network used in some theoretical studies. The output of the linear subunits is squared but not weighted and can only give an excitatory (positive) contribution to the output. There is no direct linear connection between input and output layer.

to be appropriate in experimental and theoretical studies as discussed above, it is not able to encompass all input-output nonlinearities of visual neurons. Contrast gain control [Ohzawa et al., 1982], which has a divisive effect on the input, saturation effects, or pattern adaptation are some examples of nonlinear effects present in the visual cortex that cannot be realized.

3.5.3 Relation to other learning rules

As mentioned before in Section 2.2 the definition of *slowness* in the present work (Sect. 2.3.1) and that given in other models of the primary visual cortex [Einhäuser et al., 2002, Hurri and Hyvärinen, 2003a, Körding et al., 2004] are different to some extent. In [Körding et al., 2004] the weights of neural networks equivalent to the classical model of complex cells are adapted by gradient descent in order to optimize a decorrelation and a slowness term. The slowness term is defined by the *mean* of the Δ-values in Equation (2.13). If one fully enforces the decorrelation constraint (Eq. 2.16), the units found by this rule lie in the subspace of the most slowly varying functions, but they are unique only up to an orthogonal transformation, i.e. by mixing the resulting functions through a rotation (in the space of polynomials) one would find different but equally optimal units. In the cited paper, however, the architecture of the neural networks imposes additional constraints in the sense that the polynomials that the networks can compute form a subset and not a subspace of the space of polynomials of degree 2. This implies that an arbitrary rotation could lead to functions that do not lie in the subset and are thus not representable by such neural networks (Körding 2003, personal communication). This argument shows that the two objective functions are different in some aspects.

It is interesting to notice that in our model the learning rule at the level of the subunits in the equivalent neural network is similar to the one proposed by Hurri and Hyvärinen [2003a] plus some cross-correlation terms. As shown in Section 3.6.7 if we consider a neural network like that of Figure 3.15b and we expand the SFA objective function (Eq. 2.13) at the level of the subunits, we obtain the equivalent objective function

$$\sum_{i=1}^{N'} s_i(t)^2 s_i(t-1)^2 + \sum_{\substack{i,j=1\\ i\neq j}}^{N'} s_i(t)^2 s_j(t-1)^2\,, \tag{3.2}$$

which has to be maximized ($s_i(t)$ is the activity of subunit i at time t). The first term of Equation (3.2) is equal to the objective function proposed by Hurri and Hyvärinen [2003a] in the case $f(y) = y^2$ (Sect. 2.2), i.e. in the case where the energy of the output of a unit has to be similar at successive time steps. Hurri and Hyvärinen [2003a] showed that this learning rule applied to natural video sequences learn simple-cell receptive fields. The second term of Equation (3.2) maximizes the coherence of the energy of *different* subunits at successive time steps. As a consequence, the subunits are encouraged to code for frequently occurring transformations of the features represented by the others. According to this analysis, it is tempting to conclude that temporal slowness at the level of complex cells modulates temporal coherence at the level of simple cells.

As mentioned in Section 1.1, another proposed computational principle is based on the sparseness of the output of a cell or on the independence of the outputs of a set of cells, which turns out to be equivalent in this context [Hyvärinen et al., 2001, Chap. 21.2]. The sparseness of a code can be measured by its kurtosis, where higher kurtosis corresponds to a sparser code [Willmore and Tolhurst, 2001]. Interestingly, the kurtosis of our units (mean kurtosis 12.85 ± 3.46) is much higher than that of their input (mean kurtosis 0.42 ± 0.04). This is due to the selectivity characteristics of the units. They can therefore take advantage of the benefits of a sparse representation without being explicitly optimized for it. Figure 3.16 shows an excerpt of the activity trace of a unit and the distribution of its output in response to 400,000 test frames. In a complex cell in V1 the activity would be half-rectified at some threshold since the firing rate cannot be negative.

Statistical independence can be defined on the basis of higher order statistics, like in the studies cited above, or on the basis of second order temporal statistics of the input signals. In this case, the correlation between different input signals at different time delays is minimized [Molgedey and Schuster, 1994, Belouchrani et al., 1997, Ziehe and Müller, 1998]. Blaschke et al. [2004] investigated the relation between

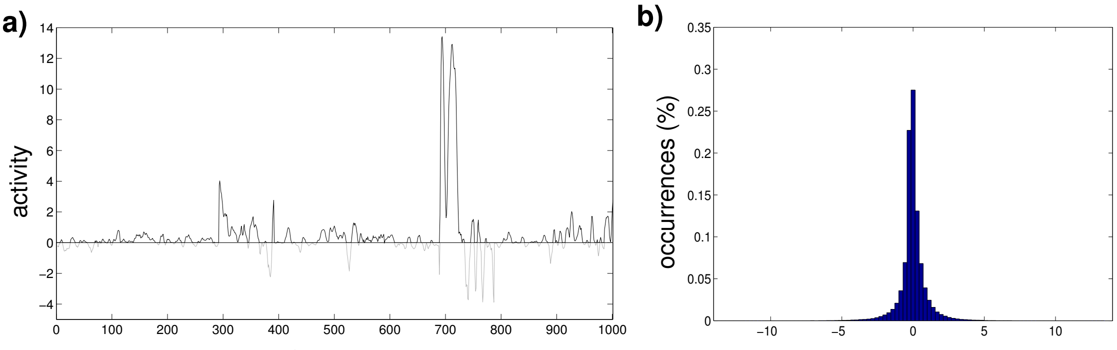


Figure 3.16 Units activity **(a)** Excerpt of the activity trace of Unit 5. In a complex cell in V1, the activity would be half-rectified at some threshold since the firing rate cannot be negative. In the plot, the spontaneous activity level has been put to zero, and negative values corresponding to inhibition have been plotted in gray. **(b)** Distribution of the activity of Unit 5 in response to 400,000 test frames. The kurtosis of the output of this unit is 19.74, which is much higher than that of its input (mean kurtosis 0.42 ± 0.04).

SFA and second order ICA and proved that in the linear case if only one time delay is considered the two methods are equivalent.

3.5.4 Conclusion

In summary we have shown that slowness leads to a great variety of complex cell properties found also in physiological experiments. Our results demonstrate that such a rich repertoire of receptive field properties can be accounted for by a single unsupervised learning principle. They also suggest a relation between the behavior of a neuron and the slowness of its output. It will be interesting to see whether this prediction will be confirmed experimentally. Earlier modeling studies with SFA [Wiskott and Sejnowski, 2002] have shown that translation, scale, and other invariances can also be learned for whole objects in a hierarchical network of SFA-modules: When trained with moving random 1D objects, such a network learns to represent the *what* and the *where* information (i.e. the identity and position) of novel objects in an invariant fashion, results that have been derived also analytically in [Wiskott, 2003b]. This suggests that slowness might be a rather general learning principle in the visual systems.

3.6 Technical remarks to Chapter 3

3.6.1 Dimensionality reduction by PCA

In our model the dimensionality of the input vectors is reduced by PCA. This corresponds to a low pass filtering of the input patches, since the principal components of natural images are linear filters of increasing frequency [Hancock et al., 1992, Field, 1994]. The exact form of the filters learned in the PCA step, however, is completely irrelevant (and thus not shown), since SFA is independent of any linear transformation of its input. An arbitrary linear mix of the principal components would lead to identical results. Due to the self-similar structure of natural images [Ruderman and Bialek, 1994, Dong and Atick, 1995b], it is in principle equivalent to work with low-pass filtered large patches or with small patches with no preprocessing. Large, low-pass filtered patches, however, are smoother and easier to analyze, especially in experiments with drifting sine gratings. In smaller patches, higher frequencies are represented as alternating positive and negative values. This raw sampling has undesired effects especially for diagonal orientations, where the highest frequencies assume a checkerboard-like appearance whose orientation is often ambiguous. Moreover, the anisotropy due to the square shape of the pixels has more influence on measurements. Control Experiment 3 (Sect. 3.4.3) shows that there is no major qualitative difference between results obtained with large, low-passed filtered patches or with smaller unprocessed patches.

3.6.2 Receptive field localization

The optimal stimuli are somewhat localized (Fig. 3.3), especially for end- or side-inhibited units. However, their size is necessarily relative to that of the input patches (i.e. by making the input patches larger we would expect to obtain larger optimal stimuli), since there is nothing in the algorithm nor in the statistics of natural images that would define an absolute scale. This is analogous to what happens in the linear case for PCA, which also produces a set of filters extending over the whole image patches when applied to natural images. In contrast, the wavelets learned by independent component analysis (ICA) [e.g. Bell and Sejnowski, 1997] are more localized and do not scale with input patch size (but might depend on the resolution used, since the frequencies of the learned filters seem to cluster around the highest possible frequency). The difference between PCA and ICA suggests that by replacing the decorrelation constraint (Eq. 2.16) (like in PCA) with an independence constraint (like in ICA) we might expect to find more localized filters with a fixed absolute scale.

3.6.3 Inhibition in cortical neurons

The exact shape and tuning of inhibition in cortical neurons is usually difficult to determine experimentally from the firing rate, which cannot be negative. Experiments studying inhibition must rely on the membrane potential, increase the neuron's firing by superimposing a "conditioning stimulus", or block inhibition with pharmacological manipulations. Each of these methods has specific drawbacks. For example adaptation to the conditioning stimulus influences the orientation tuning of the neurons [Dragoi et al., 2001]. Ringach et al. [2002] applied a new reverse correlation technique and found that suppression and enhancement have similar magnitudes and that peak enhancement tends to be slightly larger than suppression. This is compatible with our results (note that peak enhancement is larger by construction). They also showed a positive correlation between suppression and orientation selectivity. Since they used non-localized, oriented stimuli, it is impossible to say if inhibition was oriented or localized. It was also not possible to make precise statements on the feedback/feedforward or broadly/narrowly tuned structure of inhibition, although it was found to be compatible with a tuned, feedforward, additive inhibition like the one present in our model. Walker et al. [1999] showed that the inhibitory part of end- and side-inhibited cells in V1 is localized and oriented, which is compatible with our results (Sect. 3.3). In the two cells reported in [Walker et al., 1999] the inhibitory part has also the same orientation- and frequency-tuning than the excitatory part.

3.6.4 Frequency tuning, digitalization, and dimensionality reduction

The difference between the two distributions of frequency bandwidths in Figure 3.9c might be partly due to digitalization and dimensionality reduction. Our input patches are 16×16 pixels large, which means that the maximal bandwidth of our units is 3 octaves (from 1 to 8 cycles per patch, somewhat more on the diagonal). However, we reduced the number of input dimensions to 100 for both time steps using PCA, i.e. there are about 50 components per input patch (assuming that the two patches are independent). Since the principal components of natural images are linear filters of increasing frequency (see Sect. 3.6.1) we are only considering the 7×7 central Fourier components (because we have $50 = 7 \cdot 7 + 1$ principal components). This corresponds to frequencies from 1 to 4 cycles/patch, and thus to at most 2 octaves. The actual bandwidth would in general be much smaller because to reach the theoretical limit the response of a unit at the two extreme frequencies of 1 and 4 cycles/patch would need to be exactly half of the maximum response. Simulations with a higher number of input components could yield a broader distribution.

3.6.5 Direction selectivity and velocity distribution

We observed in other simulations (data not shown) that the distribution of the direction selectivity index depends on the distribution of velocities in the input sequences. Direction selectivity disappears for a velocity distribution including mostly very small translations and increases if it is skewed toward larger translations. A better estimation of the real-world distribution of velocities (both of the observer and of the environment) could improve the match between the histograms. One would perhaps need to increase the size of the input patches, since it limits the maximum velocity.

3.6.6 Tonic cells

The first two units in our simulation code for the mean pixel intensity and for the squared mean pixel intensity. It might be argued that the fact that the first two units code for such simple features argues against the slowness principle, because they might seem “uninteresting” when compared with successive units described in Section 3.3. However, although simple, the features coded by the first two cells might be fundamental ones, just like the first terms in a Taylor expansion.

3.6.7 Derivation of the relation to temporal coherence

As proved in [Blaschke et al., 2004], if the first derivative is approximated by the time difference $\dot{y}_j(t) \approx y_j(t) - y_j(t-1)$ it is equivalent to minimize Equation (2.13) or to maximize the expression

$$\langle y_j(t) y_j(t-1) \rangle_t \,, \tag{3.3}$$

since

$$\begin{aligned} \langle \dot{y}_j^2 \rangle_t &= \langle (y_j(t) - y_j(t-1))(y_j(t) - y_j(t-1)) \rangle_t \\ &= \langle y_j(t)^2 \rangle_t + \langle y_j(t-1)^2 \rangle_t - 2 \langle y_j(t) y_j(t-1) \rangle_t \\ &= 2 - 2 \langle y_j(t) y_j(t-1) \rangle_t \,, \end{aligned} \tag{3.4}$$

where in the last step we applied the unit variance constraint (2.15).

For a neural network like that shown in Figure 3.15b we can express Equation (3.3) at the level of the subunits by expanding the output y_j :

$$\begin{aligned}\langle y_j(t)y_j(t-1)\rangle_t &= \left\langle \left(\sum_{k=1}^{N'} s_k(t)^2\right)\left(\sum_{l=1}^{N'} s_l(t-1)^2\right)\right\rangle_t \\ &= \left\langle \sum_{k,l=1}^{N'} \left(s_k(t)^2 s_l(t-1)^2\right)\right\rangle_t \\ &= \left\langle \sum_{k=1}^{N'} \left(s_k(t)^2 s_k(t-1)^2\right)\right\rangle_t + \left\langle \sum_{\substack{k,l=1\\k\neq l}}^{N'} \left(s_k(t)^2 s_l(t-1)^2\right)\right\rangle_t ,\end{aligned} \tag{3.5}$$

where in the last step we split the sum over all terms into one sum over all terms with equal indices and one sum over all terms with different indices. N' is the number of the subunits.

As discussed in Section 3.5.2, the first term is equal to the objective function proposed by Hurri and Hyvärinen [2003a] and maximizes the correlation of the energy of the output of each subunit, while the second term maximizes the correlation of the energy of the output of different subunits.

Chapter 4

Analysis and interpretation of inhomogeneous quadratic forms as receptive fields

4.1 Introduction

Recent research in neuroscience has seen an increasing number of extensions of established linear techniques to their nonlinear equivalent, in both experimental and theoretical studies. This is the case, for example, for spatio-temporal receptive-field estimates in physiological studies [e.g. Touryan et al., 2002, Rust et al., 2004] and information theoretical models like principal component analysis (PCA) [Schölkopf et al., 1998] and independent component analysis (ICA) (see [Jutten and Karhunen, 2003] for a review). Additionally, new nonlinear unsupervised algorithms have been introduced, like, for example, slow feature analysis (SFA) [Wiskott and Sejnowski, 2002]. The study of the resulting nonlinear functions can be a difficult task because of the lack of appropriate tools to characterize them qualitatively and quantitatively.

To analyze the results of the model of self-organization of complex-cell receptive fields presented in Chapter 3 we have developed some of these tools to analyze quadratic functions in a high-dimensional space. Because of the complexity of the methods we describe them here in a separate chapter. The resulting characterization is in some aspects similar to that given by physiological studies, making it particularly suitable to be applied to the analysis of nonlinear receptive fields.

We are going to focus on the analysis of the inhomogeneous quadratic form

$$g(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} + c\,, \tag{4.1}$$

where $\mathbf{x}$ is an N-dimensional input vector, $\mathbf{H}$ an $N \times N$ matrix, $\mathbf{f}$ an N-dimensional vector, and c a constant. Although some of the mathematical details of this study are specific to quadratic forms only, it should be straightforward to extend most of the methods to other nonlinear functions while preserving the same interpretations. In other contexts it might be more useful to approximate the functions under consideration by a quadratic form using a Taylor expansion up to the second order and then apply the algorithms described here.

Table 4.1 lists some important terms and variables used throughout the chapter. We will refer to $\frac{1}{2}\,\mathbf{x}^T\mathbf{H}\mathbf{x}$ as the *quadratic term*, to $\mathbf{f}^T\mathbf{x}$ as the *linear term* and to c as the *constant term* of the quadratic form. Without loss of generality we assume that $\mathbf{H}$ is a symmetric matrix, since if necessary we can substitute

$\mathbf{H}$ in Equation (4.1) by the symmetric matrix $\frac{1}{2}\,(\mathbf{H}+\mathbf{H^T})$ without changing the values of the function g. We define $\mu_1,\ldots,\mu_N$ to be the eigenvalues to the eigenvectors $\mathbf{v}_1,\ldots,\mathbf{v}_N$ of $\mathbf{H}$ sorted in decreasing order $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_N$. $\mathbf{V}=(\mathbf{v}_1,\ldots,\mathbf{v}_N)$ denotes the matrix of the eigenvectors and $\mathbf{D}$ the diagonal matrix of the corresponding eigenvalues, so that $\mathbf{V}^T\mathbf{H}\mathbf{V}=\mathbf{D}$. Furthermore, $\langle\cdot\rangle_t$ indicates the mean over time of the expression included in the angle brackets.

N	Number of dimensions of the input space.
$\langle\cdot\rangle_t$	Mean over time of the expression between the two brackets.
$\mathbf{x}$	Input vector.
$g, \tilde{g}$	The considered inhomogeneous quadratic form and its restriction to a sphere.
$\mathbf{H}, \mathbf{h}_i$	$N \times N$ matrix of the quadratic term of the inhomogeneous quadratic form (Eq. 4.1) and i-th row of $\mathbf{H}$ (i.e. $\mathbf{H}=(\mathbf{h}_1,\ldots,\mathbf{h}_N)^T$). $\mathbf{H}$ is assumed to be symmetric.
$\mathbf{v}_i, \mu_i$	i-th eigenvector and eigenvalue of $\mathbf{H}$, sorted by decreasing eigenvalues (i.e. $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_N$).
$\mathbf{V}, \mathbf{D}$	The matrix of the eigenvectors $\mathbf{V}=(\mathbf{v}_1,\ldots,\mathbf{v}_N)$ and the diagonal matrix of the eigenvalues, so that $\mathbf{V}^T\mathbf{H}\mathbf{V}=\mathbf{D}$.
$\mathbf{f}$	N-dimensional vector of the linear term of the inhomogeneous quadratic form (Eq. 4.1).
c	Scalar value of the constant term of the inhomogeneous quadratic form (Eq. 4.1).
$\mathbf{x}^+, \mathbf{x}^-$	Optimal excitatory and inhibitory stimuli, $\lVert\mathbf{x}^+\rVert = \lVert\mathbf{x}^-\rVert = r$.

Table 4.1 Definition of some important terms This table lists the definition of the most important terms and the basic assumptions of this chapter.

In the next section we introduce the model system that we are going to use for illustration throughout this chapter. Section 4.3 describes two ways of analyzing a quadratic form by visualizing the coefficients of its quadratic and linear term directly and by considering the eigenvectors of its quadratic term. We then present in Section 4.4 an algorithm to compute the optimal excitatory and inhibitory stimuli, i.e. the stimuli that maximize and minimize a quadratic form, respectively, given a fixed energy constraint. In Section 4.5 we consider the invariances of the optimal stimuli, which are the transformations to which the function is most insensitive, and in the following section we introduce a test to determine which of these are statistically significant. In Section 4.7 we discuss two ways to determine the relative contribution of the different terms of a quadratic form to its output. Furthermore, in Section 4.8 we consider the techniques described above in the special case of a quadratic form without the linear term. In the end we present in Section 4.9 a two-layer neural network architecture equivalent to a given quadratic form.

4.2 Model system

To illustrate the analysis techniques presented here we make use of the quadratic forms that result from a simulation similar to that of Chapter 3 and presented in [Berkes and Wiskott, 2002]. The units learned in that simulation receive as input only one frame of the image sequence and are thus more appropriate to illustrate the analysis methods, since they are easier to visualize and interpret. In this section we summarize the settings and main results of this simulation.

We generated image sequences from a set of natural images by translation, rotation, and zoom as described in Section 3.2. The size of the input window was 16 × 16 pixels. The translation speed was chosen uniformly between 1 and 5 pixel/frame, the rotation speed between 0 and 0.1 rad/frame and the magnification difference between -0.02 and 0.02 per frame. Each sequence was 30 frames long, for a total of 150,000 frames. The functions space $\mathcal{F}$ used for SFA was the space of all polynomials of degree 2. (As shown in Section 3.2.2 every polynomial of degree 2 can be written as an inhomogeneous quadratic form.) The main difference from the simulation presented in Chapter 3 is that the input vectors to SFA were formed by the pixel intensities of one frame instead of two successive frames. The resulting units could thus not respond to changes in time.

The dimensionality of the input vectors $\mathbf{x}$ was reduced from 256 to 50 input dimensions including a whitening using principal component analysis (PCA) with

$$\mathbf{x}' = \mathbf{W}(\mathbf{x} - \langle \mathbf{x} \rangle_t)\,, \tag{4.2}$$

with $\mathbf{W}$ being a 50×256-matrix. Under an affine transformation of the coordinates $\mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{b}$ the quadratic form

$$g(\mathbf{x}') = \frac{1}{2}\mathbf{x}'^T\mathbf{H}'\mathbf{x}' + \mathbf{f}'^T\mathbf{x}' + c' \tag{4.3}$$

becomes

$$\begin{aligned}
g(\mathbf{A}\mathbf{x} + \mathbf{b}) &= \frac{1}{2}(\mathbf{A}\mathbf{x} + \mathbf{b})^T\mathbf{H}'(\mathbf{A}\mathbf{x} + \mathbf{b}) + \mathbf{f}'^T(\mathbf{A}\mathbf{x} + \mathbf{b}) + c' && (4.4)\\
&= \frac{1}{2}\mathbf{x}^T\mathbf{A}^T\mathbf{H}'\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{H}'\mathbf{A}\mathbf{x} + \frac{1}{2}\mathbf{b}^T\mathbf{H}'\mathbf{b} + \mathbf{f}'^T\mathbf{A}\mathbf{x} + \mathbf{f}'^T\mathbf{b} + c' && (4.5)\\
&= \frac{1}{2}\mathbf{x}^T(\mathbf{A}^T\mathbf{H}'\mathbf{A})\mathbf{x} + (\mathbf{A}^T\mathbf{H}'^T\mathbf{b} + \mathbf{A}^T\mathbf{f}')^T\mathbf{x} + (\frac{1}{2}\mathbf{b}^T\mathbf{H}'\mathbf{b} + \mathbf{f}'^T\mathbf{b} + c') && (4.6)\\
&=: \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} + c\,, && (4.7)
\end{aligned}$$

i.e. another quadratic form. For analysis we can thus project the learned functions back into the input space applying Equations (4.4–4.7) with $\mathbf{A} := \mathbf{W}$ and $\mathbf{b} := -\mathbf{W}\langle \mathbf{x}' \rangle_t$. Note that the rank of the quadratic term $\mathbf{H}$ after the transformation is the same as before and it has only 50 eigenvectors.

We applied SFA to the whitened data and determined the quadratic forms whose output signals vary as slowly as possible in time, sorted by decreasing slowness. The units were classified as complex cells by experiments with sine-gratings and had most of the properties described in Section 3.3, except direction selectivity which can not be computed from one input frame.

This model system is representative of the application domain considered here, which includes second-order approximations and theoretical models of physiological receptive fields.

4.3 Visualization of coefficients and eigenvectors

One way to analyze a quadratic form is to visualize its coefficients. The coefficients $f_1, \ldots, f_N$ of the linear term can be visualized and interpreted directly and give the shape of the input stimulus that maximizes the linear part given a fixed norm.

The quadratic term can be rewritten as a sum over the inner product of the j-th row $\mathbf{h}_j$ of $\mathbf{H}$ with the vector of the products $x_j x_i$ between the j-th variable x_j and all other variables:

$$\mathbf{x}^T \mathbf{H} \mathbf{x} = \sum_{j=1}^{N} x_j (\mathbf{h}_j^T \mathbf{x}) = \sum_{j=1}^{N} \mathbf{h}_j^T \begin{pmatrix} x_j x_1 \\ x_j x_2 \\ \vdots \\ x_j x_N \end{pmatrix} . \tag{4.8}$$

In other words, the response of the quadratic term is formed by the sum of the response of N linear filters $\mathbf{h}_j$ which respond to all combinations of the j-th variable with the other ones. If the input data has a two-dimensional spatial arrangement, like in our model system, the interpretation of the rows can be made easier by visualizing them as a series of images (by reshaping the vector $\mathbf{h}_j$ to match the structure of the input) and arranging them according to the spatial position of the corresponding variable x_j. In Figure 4.1 we show some of the coefficients of two functions learned in the model system. In both units, the linear term looks unstructured. The absolute values of its coefficients are small in comparison to those of the quadratic term so that its contribution to the output of the functions is very limited (cf. Sect. 4.7). The row vectors $\mathbf{h}_j$ of Unit 4 have a localized distribution of their coefficients, i.e. they only respond to combinations of the corresponding variable x_j and its neighbors. The filters $\mathbf{h}_j$ are shaped like a four-leaf clover and centered on the variable itself. Pairs of opposed leaves have positive and negative values, respectively. This suggests that the unit responds to stimuli oriented in the direction of the two positive leaves and is inhibited by stimuli with an orthogonal orientation, which is confirmed by successive analysis (cf. later in this section and Sect. 4.4). In Unit 28 the appearance of $\mathbf{h}_j$ depends on the spatial position of $\mathbf{x}_j$. In the bottom half of the receptive field the interaction of the variables with their close neighbors along the vertical orientation is weighted positively, with a negative flank on the sides. In the top half the rows have similar coefficients but with reversed polarity. As a consequence, the unit responds strongly to vertical edges in the bottom half, while vertical edges in the top half result in strong inhibition. Edges extending over the whole receptive field elicit only a weak total response. This unit is thus end-inhibited.

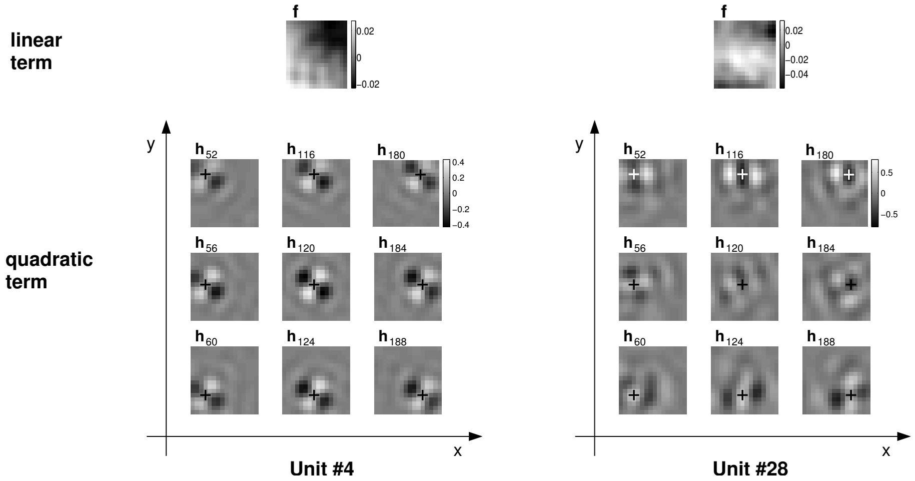


Figure 4.1 Quadratic form coefficients This figure shows some of the quadratic form coefficients of two functions learned in the model system. The top plots show the coefficients of the linear term $\mathbf{f}$, reshaped to match the two-dimensional shape of the input. The bottom plots show the coefficients of nine of the rows $\mathbf{h}_j$ of the quadratic term. The crosses indicate the spatial position of the corresponding reference index j.

Another possibility to visualize the quadratic term is to display its eigenvectors. The output of the quadratic form to one of the eigenvectors equals half of the corresponding eigenvalue, since $\frac{1}{2}\mathbf{v}_i^T\mathbf{H}\mathbf{v}_i = \frac{1}{2}\mathbf{v}_i^T(\mu_i\mathbf{v}_i) = \frac{1}{2}\mu_i$. The first eigenvector can be interpreted as the stimulus that among all input vectors with norm 1 maximizes the output of the quadratic term. The j-th eigenvector maximizes the quadratic term in the subspace that excludes the previous $j-1$ ones. In Figure 4.2 we show the eigenvectors of the two functions previously analyzed in Figure 4.1. In Unit 4 the first eigenvector looks like a Gabor wavelet. The second eigenvector has the same form except for a 90 degree phase shift of the sine grating. Since the two eigenvalues have almost the same magnitude, the response of the quadratic term is similar for the two eigenvectors and also for linear combinations with constant norm 1. As a consequence the quadratic term of this function has the main characteristics of complex cells in V1. The last two eigenvectors, which correspond to the stimuli that minimize the quadratic term, are Gabor wavelets with orientation orthogonal to the first two. This means that the output of the quadratic term is inhibited by stimuli at an orientation orthogonal to the preferred one. A similar interpretation can be given in the case of Unit 28, although in this case the first and the last two eigenvalues have the same orientation but occupy two different halves of the receptive field. This confirms that Unit 28 is end-inhibited. A direct interpretation of the remaining eigenvectors in the two functions is difficult (see also Sect. 4.8), although the magnitude of the eigenvalues shows that some of them elicit a strong response. Moreover, the interaction of the linear and quadratic terms to form the overall output of the quadratic form is not considered but cannot generally be neglected. The methods presented in the following sections often give a more direct and intuitive description of the quadratic forms.

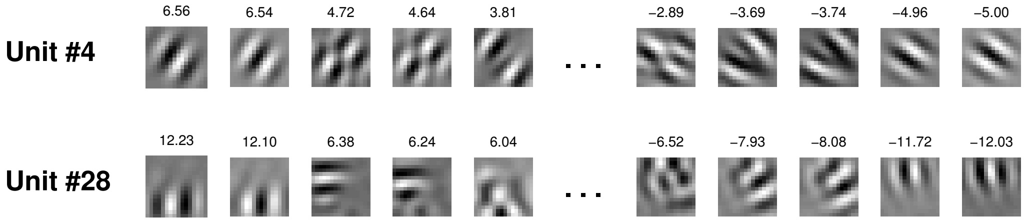


Figure 4.2 Eigenvectors of the quadratic term Eigenvectors of the quadratic term of two functions learned in the model system sorted by decreasing eigenvalues as indicated above each eigenvector.

4.4 Optimal stimuli

Another characterization of a nonlinear function can be borrowed from neurophysiological experiments, where it is common practice to characterize a neuron by the stimulus to which it responds best [for an overview see Dayan and Abbott, 2001, chap. 2.2]. Analogously, we can compute the *optimal excitatory stimulus* of g, i.e. the input vector $\mathbf{x}^+$ that maximizes g given a fixed norm $\|\mathbf{x}^+\| = r$. Note that $\mathbf{x}^+$ depends qualitatively on the value of r: if r is very small the linear term of the equation dominates, so that $\mathbf{x}^+ \approx \mathbf{f}$, while if r is very large the quadratic part dominates, so that $\mathbf{x}^+$ equals the first eigenvector of $\mathbf{H}$ (see also Sect. 4.8). We usually choose r to be the mean norm of all input vectors, since we want $\mathbf{x}^+$ to be representative of the typical input. In the same way we can also compute the *optimal inhibitory stimulus* $\mathbf{x}^-$, which minimizes the response of the function.

The fixed norm constraint corresponds to a *fixed energy constraint* [Stork and Levinson, 1982] used in experiments involving the reconstruction of the Volterra kernel of a neuron [Dayan and Abbott, 2001, chap. 2.2]. During physiological experiments in the visual system one sometimes uses stimuli with fixed contrast instead. The optimal stimuli under these two constraints may be different. For example, with fixed contrast one can extend a sine grating indefinitely in space without changing its intensity, while with fixed norm its maximum intensity is going to dim as the extent of the grating increases. The fixed contrast constraint is more difficult to enforce analytically (for example because the surface of constant contrast is not bounded).

The problem of finding the optimal excitatory stimulus under the fixed energy constraint can be mathematically formulated as follows:

$$\begin{array}{ll} \text{maximize} & g(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} + c \\ \text{under the constraint} & \mathbf{x}^T\mathbf{x} = r^2 \,. \end{array} \tag{4.9}$$

This problem is known as the *Trust Region Subproblem* and has been extensively studied in the context of numerical optimization, where a nonlinear function is minimized by successively approximating it by an inhomogeneous quadratic form, which is in turn minimized in a small neighborhood. Numerous studies have analyzed its properties, in particular in the numerically difficult case where $\mathbf{H}$ is near to singular (see [Fortin, 2000] and references therein). We make use of some basic results and extend them where needed to fit our needs.

If the linear term is equal to zero (i.e. $\mathbf{f} = \mathbf{0}$), the problem can be easily solved (it is simply the first eigenvector scaled to norm r, see Sect. 4.8). In the following we consider the more general case where $\mathbf{f} \neq \mathbf{0}$. We can use a Lagrange formulation to find the necessary conditions for the extremum:

$$\mathbf{x}^T\mathbf{x} = r^2 \tag{4.10}$$

$$\text{and} \quad \nabla[g(\mathbf{x}) - \frac{1}{2}\lambda\mathbf{x}^T\mathbf{x}] = \mathbf{0} \tag{4.11}$$

$$\Leftrightarrow \quad \mathbf{H}\mathbf{x} + \mathbf{f} - \lambda\mathbf{x} = \mathbf{0} \tag{4.12}$$

$$\Leftrightarrow \quad \mathbf{x} = (\lambda\mathbf{I} - \mathbf{H})^{-1}\,\mathbf{f}\,, \tag{4.13}$$

where we inserted the factor $\frac{1}{2}$ for mathematical convenience. According to Theorem 3.1 in [Fortin, 2000], if an $\mathbf{x}$ that satisfies Equation (4.13) is a solution to (4.9), then $(\lambda\mathbf{I} - \mathbf{H})$ is positive semidefinite (i.e. all eigenvalues are greater than or equal to 0). This imposes a strong lower bound on the range of possible values for λ. Note that the matrix $(\lambda\mathbf{I} - \mathbf{H})$ has the same eigenvectors $\mathbf{v}_i$ as $\mathbf{H}$ with eigenvalues $(\lambda - \mu_i)$, since

$$(\lambda\mathbf{I} - \mathbf{H})\mathbf{v}_i = \lambda\mathbf{v}_i - \mathbf{H}\mathbf{v}_i \tag{4.14}$$

$$= \lambda\mathbf{v}_i - \mu_i\mathbf{v}_i \tag{4.15}$$

$$= (\lambda - \mu_i)\mathbf{v}_i\,. \tag{4.16}$$

For $(\lambda\mathbf{I} - \mathbf{H})$ to be positive semidefinite all eigenvalues must be nonnegative, and thus λ must be greater than or equal to the largest eigenvalue μ_1,

$$\mu_1 \leq \lambda\,. \tag{4.17}$$

An upper bound for lambda can be found by considering an upper bound for the norm of $\mathbf{x}$. First we note that matrix $(\lambda\mathbf{I} - \mathbf{H})^{-1}$ is symmetric and has the same eigenvectors as $\mathbf{H}$ with eigenvalues $1/(\lambda - \mu_i)$:

$$(\lambda\mathbf{I} - \mathbf{H})\mathbf{v}_i \underset{(4.16)}{=} (\lambda - \mu_i)\mathbf{v}_i \tag{4.18}$$

$$\Leftrightarrow \quad \frac{1}{(\lambda - \mu_i)}\mathbf{v}_i = (\lambda\mathbf{I} - \mathbf{H})^{-1}\mathbf{v}_i\,. \tag{4.19}$$

We also know that the inequality

$$\|\mathbf{A}\mathbf{v}\| \leq \|\mathbf{A}\|\|\mathbf{v}\| \tag{4.20}$$

holds for every matrix $\mathbf{A}$ and vector $\mathbf{v}$. $\|\mathbf{A}\|$ is here the spectral norm of $\mathbf{A}$, which for symmetric matrices is simply the largest absolute eigenvalue. With this we find an upper bound for $\|\mathbf{x}\|$:

$$\|\mathbf{x}\| = \|(\lambda\mathbf{I} - \mathbf{H})^{-1}\mathbf{f}\| \tag{4.21}$$

$$\underset{(4.20)}{\leq} \|(\lambda\mathbf{I} - \mathbf{H})^{-1}\| \, \|\mathbf{f}\| \tag{4.22}$$

$$\underset{(4.19)}{=} \max_i \left\{ \left| \frac{1}{\lambda - \mu_i} \right| \right\} \|\mathbf{f}\| \tag{4.23}$$

$$\underset{(4.17)}{=} \frac{1}{\lambda - \mu_1} \|\mathbf{f}\| \, . \tag{4.24}$$

We can therefore discard all values of λ for which the upper bound is smaller than r (in which case the condition $\|\mathbf{x}\| = r$ would be violated), i.e. λ must fulfill the equation

$$r \leq \frac{1}{\lambda - \mu_1} \|\mathbf{f}\| \tag{4.25}$$

$$\Leftrightarrow \quad \lambda \leq \frac{\|\mathbf{f}\|}{r} + \mu_1 \, . \tag{4.26}$$

The optimization problem (4.9) is thus reduced to a search over λ on the interval $\left[\mu_1, \left(\frac{\|\mathbf{f}\|}{r} + \mu_1\right)\right]$ until $\mathbf{x}$ defined by (4.13) fulfills the constraint $\|\mathbf{x}\| = r$ (Eq. 4.10). Vector $\mathbf{x}$ and norm $\|\mathbf{x}\|$ can be efficiently computed for each λ using the eigenvalue decomposition of $\mathbf{f}$:

$$\mathbf{x} \underset{(4.13)}{=} (\lambda\mathbf{I} - \mathbf{H})^{-1}\mathbf{f} \tag{4.27}$$

$$= (\lambda\mathbf{I} - \mathbf{H})^{-1} \sum_i \mathbf{v}_i \, (\mathbf{v}_i^T \mathbf{f}) \tag{4.28}$$

$$= \sum_i (\lambda\mathbf{I} - \mathbf{H})^{-1} \, \mathbf{v}_i \, (\mathbf{v}_i^T \mathbf{f}) \tag{4.29}$$

$$= \sum_i \frac{1}{\lambda - \mu_i} \mathbf{v}_i \, (\mathbf{v}_i^T \mathbf{f}) \tag{4.30}$$

and

$$\|\mathbf{x}\|^2 = \sum_i \left(\frac{1}{\lambda - \mu_i} \right)^2 (\mathbf{v}_i^T \mathbf{f})^2 \, , \tag{4.31}$$

where the terms $\mathbf{v}_i^T \mathbf{f}$ and $(\mathbf{v}_i^T \mathbf{f})^2$ are constant for each quadratic form and can be computed in advance. The last equation also shows that the norm of $\mathbf{x}$ is monotonically decreasing in the considered interval, so that there is exactly one solution and the search can be efficiently performed by a bisection method. $\mathbf{x}^-$ can be found in the same way by maximizing the negative of g. Table 4.2 contains the pseudo-code of an algorithm that implements all the considerations above (see also Additional Material).

If the matrix $\mathbf{H}$ is negative definite (i.e. all its eigenvalues are negative) there is a global maximum that may not lie on the sphere, which might be used in substitution for $\mathbf{x}^+$ if it lies in a region of the input space that has a high probability of being reached (the criterion is quite arbitrary, but the region could be chosen to include, for example, 75% of the input data with highest density). The gradient of the function disappears at the global extremum such that it can be found by solving a simple linear equation system:

$$\nabla g(\mathbf{x}) = \mathbf{H}\mathbf{x} + \mathbf{f} = \mathbf{0} \tag{4.32}$$

$$\Leftrightarrow \quad \mathbf{x} = -\mathbf{H}^{-1}\mathbf{f}. \tag{4.33}$$

```
input: H, f, c: quadratic form
       r: norm of the solution (> eps)
       eps: tolerance of norm(x) from r

output: x_max: optimal excitatory stimulus x+

 1-  compute the eigenvalues mu(1) ... mu(N)
 1-  and the eigenvectors v(1) ... v(N) of H
 2-  # compute the coefficients of the eigenvector decomposition
 2-  # of f (Eq. 4.28)
 2-  alpha(i) := v(i)'*f     (i = 1 ... N)

 3-  # compute the range of the parameter lambda (Eqs. 4.17 and 4.26)
 3-  lambda_left := max(mu)
 4-  lambda_right := norm(f)/r + max(mu)

     # search by bisection until norm(x)^2 = r^2
 5-  # norm_x_2 holds the value of norm(x)^2 at the current lambda
 5-  norm_x_2 := 0
 6-  while abs(sqrt(norm_x_2)-r) > eps:
 7-    # bisect the interval
 7-    lambda = (lambda_right-lambda_left)/2 + lambda_left
 8-    # compute the eigenvalues of (lambda*I - H)^-1 (Eq. 4.19)
 8-    beta(i) := 1/(lambda-mu(i))     (i = 1 ... N)
 9-    # compute norm(x)^2 at lambda (Eq. 4.31)
 9-    norm_x_2 = sum(beta(i)^2 * alpha(i)^2)
       # update the interval limits
10-    if norm_x_2 > r^2:
11-      lambda_left = lambda
12-    else:
13-      lambda_right = lambda

14-  # lambda found, compute the solution (Eq. 4.30)
14-  x_max = sum(beta(i)*v(i)*alpha(i))
```

Table 4.2 Algorithm to compute the optimal stimuli Pseudo-code of the algorithm that computes the optimal excitatory stimulus of the inhomogeneous quadratic form. In the code, $\mathbf{A}'$ means "$\mathbf{A}$ transpose". The algorithm can be used to compute the optimal inhibitory stimulus by calling it with the negative of the quadratic form.

In the same way a positive definite matrix $\mathbf{H}$ has a negative global minimum, which might be used in substitution for $\mathbf{x}^-$.

For general nonlinear functions, the optimal stimuli can be found by standard techniques like gradient descent or stochastic methods [see e.g. Bishop, 1995, chap. 7]. This is also useful for quadratic forms if there are additional constraints which are difficult to incorporate analytically (e.g. the non-negativity of the elements of $\mathbf{x}^+$). In Section 4.5 we provide an explicit formula for the gradient and the second derivative of g restricted to the sphere of vectors of fixed energy that can be used in such situations. A well-known drawback of online optimization algorithms is that they can get trapped in local extrema, depending on the point from which the algorithm started. In this context, the vector in the input data set that yields the largest output of g can be a good starting point.

In Figure 4.3 we show the optimal stimuli for the first 48 quadratic forms in the model system. In almost all cases $\mathbf{x}^+$ looks like a Gabor wavelet, in agreement with physiological data (cf. Sect. 3.3). Like in the simulation of Chapter 3, $\mathbf{x}^-$ is usually structured and looks like a Gabor wavelet, which suggests that inhibition plays an important role. $\mathbf{x}^+$ can be used to compute the position and size of the receptive fields as well as the preferred orientation and frequency of the units for successive experiments.

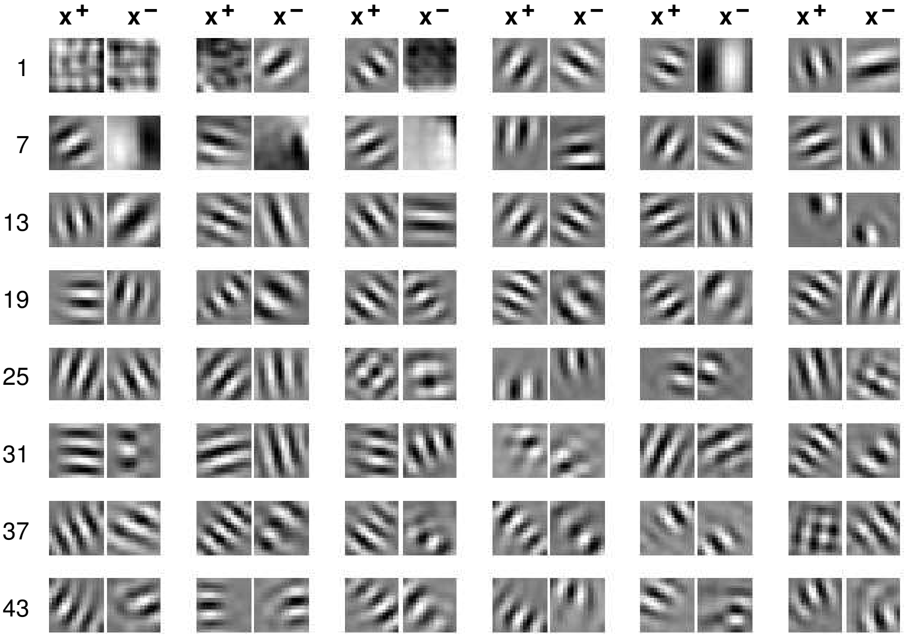


Figure 4.3 Optimal stimuli Optimal stimuli of the first 48 quadratic forms in the model system. $\mathbf{x}^+$ looks like a Gabor wavelet in almost all cases, in agreement with physiological data. $\mathbf{x}^-$ is usually structured and is also similar to a Gabor wavelet, which suggests that inhibition plays an important role.

Note that although $\mathbf{x}^+$ is the stimulus that elicits the strongest response in the function, it doesn't necessarily mean that it is representative of the class of stimuli that give the most important contribution to its output. This depends on the distribution of the input vectors: if $\mathbf{x}^+$ lies in a low-density region of the input space, it is possible that other kinds of stimuli drive the function more often. In that case they might be considered more relevant than $\mathbf{x}^+$ to characterize the function. Symptomatic for this effect would be if the output of a function when applied to its optimal stimulus would lie far outside the range of normal activity. This means that $\mathbf{x}^+$ can be an atypical, artificial input that pushes the function in an uncommon state. A similar effect has also been reported in some physiological papers comparing the response of neurons to natural stimuli and to artificial stimuli such as sine gratings [Baddeley et al., 1997]. Thus the characterization of a neuron or a nonlinear function as a feature detector via the optimal stimulus is at least incomplete [see also MacKay, 1985]. However, the optimal stimuli remain extremely informative in practice.

4.5 Invariances

Since the considered functions are nonlinear, the optimal stimuli do not provide a complete description of their properties. We can gain some additional insights by studying a neighborhood of $\mathbf{x}^+$ and $\mathbf{x}^-$. An interesting question is, for instance, to which transformations of $\mathbf{x}^+$ or $\mathbf{x}^-$ the function is invariant. This is similar to the common interpretation of neurons as detectors of a specific feature of the input which are invariant to a local transformation of that feature. For example, complex cells in the primary visual cortex are thought to respond to oriented bars and to be invariant to a local translation. In this section we are going to consider the function $\tilde{g}$, defined as g restricted to the sphere $\mathcal{S}$ of radius r, since like in Section 4.4 we want to compare input vectors having fixed energy. Notice that although $\tilde{g}$ and g take the same values on $\mathcal{S}$ (i.e. $\tilde{g}(\mathbf{x}) = g(\mathbf{x})$ for each $\mathbf{x} \in \mathcal{S}$) they are two distinct mathematical objects. For example, the gradient of $\tilde{g}$ in $\mathbf{x}^+$ is zero because $\mathbf{x}^+$ is by definition a maximum of $\tilde{g}$. On the other hand, the gradient of g in the same point is $\mathbf{H}\mathbf{x}^+ + \mathbf{f}$, which is in general different from zero.

Strictly speaking, there is no invariance in $\mathbf{x}^+$, since it is a maximum and the output of $\tilde{g}$ decreases in all directions (except in the special case where the linear term is zero and the first two or more eigenvalues are equal). In a general non-critical point $\mathbf{x}^*$ (i.e. a point where the gradient does not disappear) the rate of change in any direction $\mathbf{w}$ is given by its inner product with the gradient, $\nabla\tilde{g}(\mathbf{x}^*) \cdot \mathbf{w}$. For all vectors orthogonal to the gradient (which span an $N - 2$ dimensional space) the rate of change is thus zero. Note that this is not merely a consequence of the fact that the gradient is a first-order approximation of $\tilde{g}$. By the implicit function theorem [see e.g. Walter, 1995, Theorem 4.5], in each open neighborhood U of a non-critical point $\mathbf{x}^*$ there is an $N - 2$ dimensional level surface $\{\mathbf{x} \in U \subset \mathcal{S} \mid \tilde{g}(\mathbf{x}) = \tilde{g}(\mathbf{x}^*)\}$, since the domain of $\tilde{g}$ (the sphere $\mathcal{S}$) is an $N - 1$ dimensional surface and its range ($\mathbb{R}$) is 1 dimensional. Each non-critical point thus belongs to an $N - 2$ dimensional surface where the value of the $\tilde{g}$ stays constant. This is a somewhat surprising result: for an optimal stimulus there does not exist any invariance (except in some degenerate cases); for a general sub-optimal stimulus there exist many invariances.

This shows that although it might be useful to observe for example that a given function f that maps images to real values is invariant to stimulus rotation, one should keep in mind that in a generic point there are a large number of other transformations to which the function might be equally invariant but that would lack an easy interpretation. The strict concept of invariance is thus not useful for our analysis, since in the extrema we have no invariances at all, while in a general point they are the typical case, and the only interesting direction is the one of maximal change, as indicated by the gradient. In the extremum $\mathbf{x}^+$, however, since the output changes in all directions, we can relax the definition of invariance and look for the transformation to which the function changes as little as possible, as indicated by the direction with the smallest absolute value of the second derivative. (In a non-critical point this weak definition of invariance still does not help: if the quadratic form that represents the second derivative has positive as well as negative eigenvalues, there is still a $N - 3$ dimensional surface where the second derivative is zero.)

To study the invariances of the function g in a neighborhood of its optimal stimulus respecting the fixed energy constraint we have defined the function $\tilde{g}$ as the function g restricted to $\mathcal{S}$. This is particularly relevant here since we want to analyze the derivatives of the function, i.e. its change under small movements. Any straight movement in space is going to leave the surface of the sphere. We must therefore be able to define movements on the sphere itself. This can be done by considering a path $\boldsymbol{\varphi}(t)$ on the surface of $\mathcal{S}$ such that $\boldsymbol{\varphi}(0) = \mathbf{x}^+$ and then studying the change of g along $\boldsymbol{\varphi}$. By doing this, however, we add the rate of change of the path (i.e. its acceleration) to that of the function. Of all possible paths we must take the ones that have as little acceleration as possible, i.e. those that have just the acceleration that is needed to stay on the surface. Such a path is called a *geodetic*. The geodetics of a sphere are great circles and our paths are thus defined as

$$\boldsymbol{\varphi}(t) = \cos(t/r) \cdot \mathbf{x}^+ + \sin(t/r) \cdot r\mathbf{w} \tag{4.34}$$

for each direction $\mathbf{w}$ in the tangential space of $\mathcal{S}$ in $\mathbf{x}^+$ (i.e. for each $\mathbf{w}$ orthogonal to $\mathbf{x}^+$), as shown in Figure 4.4. The $1/r$ factor in the cosine and sine arguments normalizes the function such that $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\varphi}(0) = \mathbf{w}$ with $\|\mathbf{w}\| = 1$.

The first derivative of $\tilde{g}$ along $\boldsymbol{\varphi}$ is

$$\frac{\mathrm{d}}{\mathrm{d}t}(\tilde{g} \circ \boldsymbol{\varphi})(t) = \frac{\mathrm{d}}{\mathrm{d}t}\left[\frac{1}{2}\boldsymbol{\varphi}(t)^T \mathbf{H}\boldsymbol{\varphi}(t) + \mathbf{f}^T\boldsymbol{\varphi}(t) + c\right] \tag{4.35}$$

$$\begin{aligned} &= \frac{\mathrm{d}}{\mathrm{d}t}\Bigg[\frac{1}{2}\big(\cos(t/r)\ \mathbf{x}^+ + \sin(t/r)\ r\mathbf{w}\big)^T \mathbf{H}\big(\cos(t/r)\ \mathbf{x}^+ + \sin(t/r)\ r\mathbf{w}\big) \\ &\qquad\qquad + \mathbf{f}^T\big(\cos(t/r)\ \mathbf{x}^+ + \sin(t/r)\ r\mathbf{w}\big) + c\Bigg] \end{aligned} \tag{4.36}$$

$$\begin{aligned} &= \frac{\mathrm{d}}{\mathrm{d}t}\Bigg[\frac{1}{2}\cos(t/r)^2\ \mathbf{x}^{+T}\mathbf{H}\mathbf{x}^+ + \cos(t/r)\sin(t/r)\ r\ \mathbf{x}^{+T}\mathbf{H}\mathbf{w} + \frac{1}{2}\sin(t/r)^2\ r^2\ \mathbf{w}^T\mathbf{H}\mathbf{w} \\ &\qquad\qquad + \cos(t/r)\ \mathbf{f}^T\mathbf{x}^+ + \sin(t/r)\ r\ \mathbf{f}^T\mathbf{w} + c\Bigg] \end{aligned} \tag{4.37}$$

(since $\mathbf{w}\mathbf{H}\mathbf{x}^{+T} = \mathbf{x}^{+T}\mathbf{H}\mathbf{w}$, because $\mathbf{H}$ is symmetric)

$$\begin{aligned} &= -\frac{1}{r}\sin(t/r)\cos(t/r)\ \mathbf{x}^{+T}\mathbf{H}\mathbf{x}^+ + \cos(2t/r)\ \mathbf{x}^{+T}\mathbf{H}\mathbf{w} + \sin(t/r)\cos(t/r)\ r\ \mathbf{w}^T\mathbf{H}\mathbf{w} \\ &\qquad\qquad - \frac{1}{r}\sin(t/r)\ \mathbf{f}^T\mathbf{x}^+ + \cos(t/r)\ \mathbf{f}^T\mathbf{w}\,. \end{aligned} \tag{4.38}$$

Taking the second derivative we obtain

$$\begin{aligned} \frac{\mathrm{d}^2}{\mathrm{d}t^2}(\tilde{g} \circ \boldsymbol{\varphi})(t) = &-\frac{1}{r^2}\cos(2t/r)\ \mathbf{x}^{+T}\mathbf{H}\mathbf{x}^+ - \frac{2}{r}\sin(2t/r)\ \mathbf{x}^{+T}\mathbf{H}\mathbf{w} + \cos(2t/r)\ \mathbf{w}^T\mathbf{H}\mathbf{w} \\ &- \frac{1}{r^2}\cos(t/r)\ \mathbf{f}^T\mathbf{x}^+ - \frac{1}{r}\sin(t/r)\ \mathbf{f}^T\mathbf{w}\,. \end{aligned} \tag{4.39}$$

In $t = 0$ we have

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}(\tilde{g} \circ \boldsymbol{\varphi})(0) = \mathbf{w}^T\mathbf{H}\mathbf{w} - \frac{1}{r^2}(\mathbf{x}^{+T}\mathbf{H}\mathbf{x}^+ + \mathbf{f}^T\mathbf{x}^+)\,, \tag{4.40}$$

i.e. the second derivative of $\tilde{g}$ in $\mathbf{x}^+$ in the direction of $\mathbf{w}$ is composed of two terms: $\mathbf{w}^T\mathbf{H}\mathbf{w}$ corresponds to the second derivative of g in the direction of $\mathbf{w}$, while the constant term $-1/r^2 \cdot (\mathbf{x}^{+T}\mathbf{H}\mathbf{x}^+ + \mathbf{f}^T\mathbf{x}^+)$ depends on the curvature of the sphere $1/r^2$ and on the gradient of g in $\mathbf{x}^+$ orthogonally to the surface of the sphere:

$$\nabla g(\mathbf{x}^+) \cdot \mathbf{x}^+ = (\mathbf{H}\mathbf{x}^+ + \mathbf{f})^T\mathbf{x}^+ \tag{4.41}$$

$$= \mathbf{x}^{+T}\mathbf{H}\mathbf{x}^+ + \mathbf{f}^T\mathbf{x}^+\,. \tag{4.42}$$

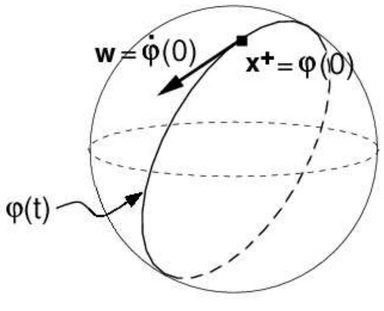


Figure 4.4 Geodetics on a sphere To compute the second derivative of the quadratic form on the surface of the sphere one can study the function along special paths on the sphere, known as geodetics. Geodetics of a sphere are great circles.

To find the direction in which $\tilde{g}$ decreases as little as possible we only need to minimize the absolute value of the second derivative (Eq. 4.40). This is equivalent to maximizing the first term $\mathbf{w}^T\mathbf{H}\mathbf{w}$ in (4.40), since the second derivative in $\mathbf{x}^+$ is always negative (because $\mathbf{x}^+$ is a maximum of $\tilde{g}$) and the second term is constant. $\mathbf{w}$ is orthogonal to $\mathbf{x}^+$ and thus the maximization must be performed in the space tangential to the sphere in $\mathbf{x}^+$. This can be done by computing a basis $\mathbf{b}_2, \ldots, \mathbf{b}_N$ of the tangential space (for example using the Gram-Schmidt orthogonalization on $\mathbf{x}^+, \mathbf{e}_1, \ldots, \mathbf{e}_{N-1}$ where $\mathbf{e}_i$ is the canonical basis of $\mathbb{R}^N$) and replacing the matrix $\mathbf{H}$ by

$$\tilde{\mathbf{H}} = \mathbf{B}^T\mathbf{H}\mathbf{B}\,, \tag{4.43}$$

where $\mathbf{B} = (\mathbf{b}_2, \cdots, \mathbf{b}_N)$. The direction of the smallest second derivative corresponds to the eigenvector $\tilde{\mathbf{v}}_1$ of $\tilde{\mathbf{H}}$ with the largest positive eigenvalue. The eigenvector must then be projected back from the tangential space into the original space by a multiplication with $\mathbf{B}$:

$$\mathbf{w}_1 = \mathbf{B}\tilde{\mathbf{v}}_1\,. \tag{4.44}$$

The remaining eigenvectors corresponding to eigenvalues of decreasing value are also interesting, as they point in orthogonal directions where the function changes with gradually increasing rate of change.

To visualize the invariances, we move $\mathbf{x}^+$ (or $\mathbf{x}^-$) along a path on the sphere in the direction of a vector $\mathbf{w}_i$ according to

$$\mathbf{x}(\alpha) = \cos(\alpha)\mathbf{x}^+ + \sin(\alpha)r\mathbf{w}_i \tag{4.45}$$

for $\alpha \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ as illustrated in Figure 4.5. At each point we measure the response of the function to the new input vector, and stop when it drops under 80% of the maximal response. In this way we generate for each invariance a movie like those shown in Figure 4.6 for some of the optimal stimuli (the corresponding animations are available online, see Additional Material). Each frame of such a movie contains a nearly-optimal stimulus. Using this analysis we can systematically scan a neighborhood of the optimal stimuli, starting from the transformations to which the function is most insensitive up to those that lead to a great change in response. Note that our definition of invariance applies only locally to a small neighborhood of $\mathbf{x}^+$. The path followed in (4.45) goes beyond such a neighborhood and is appropriate only for visualization. Table 4.3 contains the pseudo-code of an algorithm that computes and visualizes the invariances of the optimal stimuli (see also Additional Material).

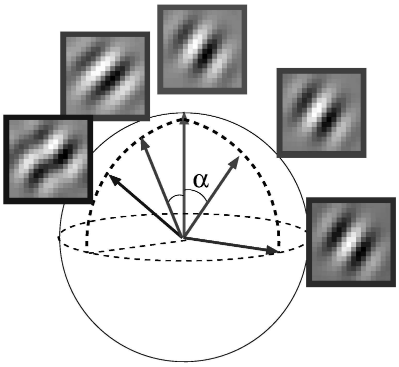


Figure 4.5 Visualization of the invariances This figure illustrates how the invariances are visualized. Starting from the optimal stimulus (top) we move on the sphere in the direction of an invariance until the response of the function drops below 80% of the maximal output or reaches 90 degrees. In the figure two invariances of Unit 4 are visualized. The one on the right represents a phase-shift invariance and preserves more than 80% of the maximal output until 90 degrees (the output at 90 degrees is 99.6% of the maximum). The one on the left represents an invariance to orientation change with an output that drops below 80% at 55 degrees.

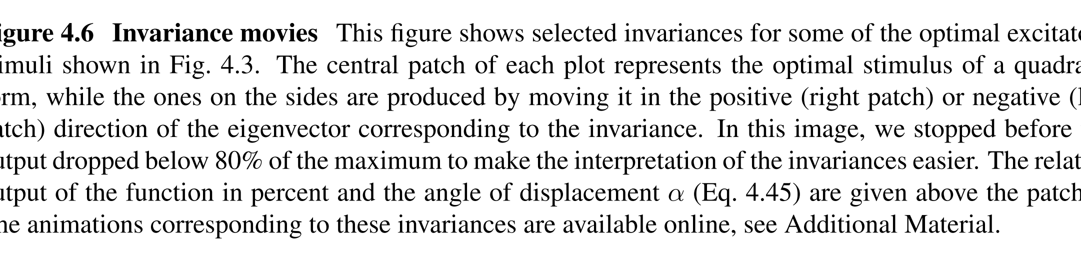

a) Unit #4, Inv. #1 - Phase shift
99.6% (−90°)
100% (0°)
99.5% (90°)
b) Unit #6, Inv. #3 - Position change
86% (−59°)
100% (0°)
84% (59°)
c) Unit #13, Inv. #4 - Size change
91% (−29°)
100% (0°)
94% (24°)
d) Unit #14, Inv. #5 - Frequency change
81% (−37°)
100% (0°)
84% (36°)
e) Unit #9, Inv. #3 - Orientation change
88% (−44°)
100% (0°)
85% (51°)
f) Unit #6, Inv. #5 - Curvature change
80% (−42°)
100% (0°)
89% (29°)

Figure 4.6 Invariance movies This figure shows selected invariances for some of the optimal excitatory stimuli shown in Fig. 4.3. The central patch of each plot represents the optimal stimulus of a quadratic form, while the ones on the sides are produced by moving it in the positive (right patch) or negative (left patch) direction of the eigenvector corresponding to the invariance. In this image, we stopped before the output dropped below 80% of the maximum to make the interpretation of the invariances easier. The relative output of the function in percent and the angle of displacement α (Eq. 4.45) are given above the patches. The animations corresponding to these invariances are available online, see Additional Material.

```
input: H, f, c: quadratic form
       x_max: optimal excitatory stimulus x+
       dalpha: precision (angular step in degrees on the sphere for
                          each frame of the invariance movies)

output: w(1) ... w(N-1) directions of the invariances, sorted by
                        increasing magnitude of the second derivative
        nu(1) ... nu(N-1) value of the second derivative in the
                          directions w(1) ... w(N-1)

 1-  # determine the radius of the sphere
 1-  r := norm(x_max)

 2-  # find a basis for the tangential plane of the sphere in x+
 2-  # e(1) ... e(N) is the canonical basis for R^N
 2-  perform a Gram-Schmidt orthogonalization on
 2-  x_max, e(1), ..., e(N-1) and save the results in b(1) ... b(N)

     # restrict the matrix H to the tangential plane
 3-  B := matrix with b(2) ... b(N) as columns
 4-  Ht := B'*H*B
     # compute the invariances
 5-  compute the eigenvalues nu(1) ... nu(N-1) and the eigenvectors
 5-  w(1) ... w(N-1) of Ht, sorted by decreasing nu(i)
 6-  # compute the second derivative in the direction of the
 6-  # eigenvectors (Eq. 4.40)
 6-  nu(i) := nu(i) - 1/r^2 * (x_max'*H*x_max + f'*x_max)
 7-  # project the eigenvectors back to R^N
 7-  w(i) := B*w(i)

 8-  # compute the value of the function at the maximum
 8-  out0 := 0.5*x_max'*H*x_max + f'*x_max + c
 9-  # minimal threshold value (80 percent of the maximum)
 9-  minout := 0.8*out0
     # visualize the invariances (Eq. 4.45)
10-  for i from 1 to N-1:
11-   out := out0
12-   alpha := 0
13-   x := x_max
14-   while out > minout and alpha < 90:
15-     visualize x
16-     alpha := alpha + dalpha
17-     x := cos(alpha)*x_max + sin(alpha)*r*w(i)
18-     out := 0.5*x'*H*x + f'*x + c

19-  repeat the visualization from step 10 with negative dalpha
```

Table 4.3 Algorithm to compute and visualize the invariances Pseudo-code of the algorithm that computes and visualizes the invariances of g in $\mathbf{x}^+$. In the code, $\mathbf{A}'$ means "$\mathbf{A}$ transpose".

4.6 Significant invariances

The procedure described above finds for each optimal stimulus a set of $N-1$ invariances ordered by the degree of invariance (i.e. by increasing magnitude of the second derivative). We would like to know which of these are statistically significant. An invariance can be defined to be significant if the function changes exceptionally little (less than chance level) in that direction, which can be measured by the value of the second derivative: the smaller its absolute value, the slower the function will change.

To test for their significance, we compare the second derivatives of the invariances of the quadratic form we are considering with those of random inhomogeneous quadratic forms that are equally adapted to the statistics of the input data. We therefore constrain the random quadratic forms to produce an output that has the same variance and mean as the output of the analyzed ones when applied to the input stimuli. Without loss of generality, we assume here zero mean and unit variance. These constraints are compatible with the ones that are usually imposed on the functions learned by many theoretical models, including the one presented in Chapter 3. Because of this normalization the distribution of the random quadratic forms depends on the distribution of the input data.

To understand how to efficiently build random quadratic forms under these constraints, it is useful to think in terms of a dual representation of the problem. A quadratic form over the input space is equivalent to a linear function over the space of the input expanded to all monomials of degree one and two using the function $\mathbf{h}((x_1,\ldots,x_n)^T) := (x_1x_1, x_1x_2, x_1x_3, \ldots, x_nx_n, x_1, \ldots, x_n)^T$, i.e.

$$\frac{1}{2}\mathbf{x}^T \underbrace{\begin{pmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{12} & h_{22} & & \\ \vdots & & \ddots & \vdots \\ h_{1n} & & \cdots & h_{nn} \end{pmatrix}}_{\mathbf{H}} \mathbf{x} + \underbrace{\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix}}_{\mathbf{f}}^T \mathbf{x} + c = \underbrace{\begin{pmatrix} \frac{1}{2}h_{11} \\ h_{12} \\ h_{13} \\ \vdots \\ \frac{1}{2}h_{nn} \\ f_1 \\ \vdots \\ f_n \end{pmatrix}}_{\mathbf{q}}^T \underbrace{\begin{pmatrix} x_1x_1 \\ x_1x_2 \\ x_1x_3 \\ \vdots \\ x_nx_n \\ x_1 \\ \vdots \\ x_n \end{pmatrix}}_{\mathbf{h}(\mathbf{x})} + c\,. \tag{4.46}$$

We can whiten the *expanded* input data $\mathbf{h}(\mathbf{x})$ by subtracting its mean $\langle \mathbf{h}(\mathbf{x}) \rangle_t$ and transforming it with a whitening matrix $\mathbf{S}$. In this new coordinate system, each linear filter with norm 1 fulfills the unit variance and zero mean constraints by construction. We can thus choose a random vector $\mathbf{q}'$ of length 1 in the whitened, expanded space and derive the corresponding quadratic form in the original input space:

$$\begin{aligned} \mathbf{q}'^T \left(\mathbf{S}(\mathbf{h}(\mathbf{x}) - \langle \mathbf{h}(\mathbf{x}) \rangle_t)\right) &= \underbrace{(\mathbf{S}^T\mathbf{q}')}_{=:\mathbf{q}}{}^T \left(\mathbf{h}(\mathbf{x}) - \langle \mathbf{h}(\mathbf{x}) \rangle_t\right) && (4.47) \\ &= \mathbf{q}^T (\mathbf{h}(\mathbf{x}) - \langle \mathbf{h}(\mathbf{x}) \rangle_t) && (4.48) \\ &\underset{(4.46)}{=} \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} - \mathbf{q}^T \langle \mathbf{h}(\mathbf{x}) \rangle_t && (4.49) \\ &= \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} + c\,, && (4.50) \end{aligned}$$

with appropriately defined $\mathbf{H}$ and $\mathbf{f}$ according to (4.46).

We can next compute the optimal stimuli and the second derivative of the invariances of the obtained random quadratic form. To make sure that we get independent measurements we only keep one second derivative chosen at random for each random function. This operation, repeated over many quadratic forms, allows us to determine a distribution of the second derivatives of the invariances and a corresponding confidence interval.

Figure 4.7a shows the distribution of 50,000 independent second derivatives of the invariances of random quadratic forms and the distribution of the second derivatives of all invariances of the first 50 quadratic forms learned in the model system. The dashed line indicates the 95% confidence interval derived from the former distribution. The latter is more skewed towards small second derivatives and has a clear peak near zero. 28% of all invariances were classified to be significant. Figure 4.7b shows the number of significant invariances for each individual quadratic form in the model system. Each function has 49 invariances since the rank of the quadratic term is 50 (see Sect. 4.2). The plot shows that the number of significant invariances decreases with increasing ordinal number (the functions are ordered by slowness, the first ones being the slowest). 46 units out of 50 have 3 or more significant invariances. The first invariance, which corresponds to a phase shift invariance, was always classified as significant, which confirms that the units behave like complex cells.

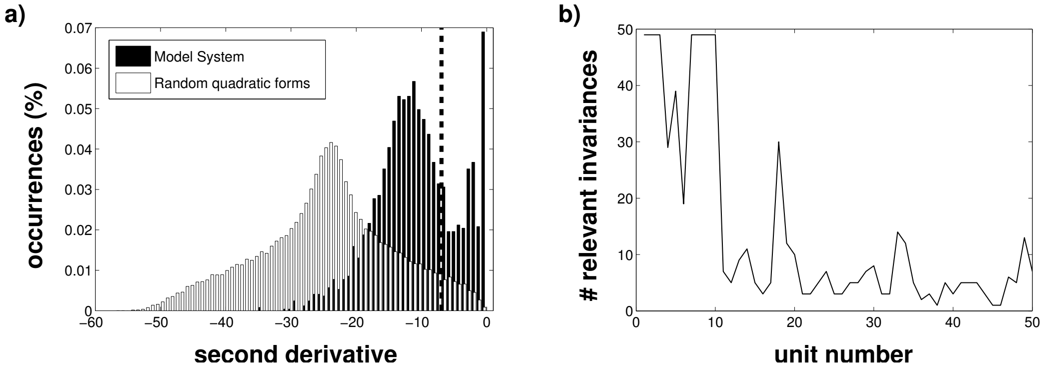


Figure 4.7 Significant invariances **(a)** Distribution of 50,000 independently drawn second derivatives of the invariances of random quadratic forms and distribution of the second derivatives of all invariances of the first 50 quadratic forms learned in the model system. The dashed line indicates the 95% confidence interval as derived from the random quadratic forms. The distribution in the model system is more skewed towards small second derivatives and has a clear peak near zero. 28% of all invariances were classified as significant. **(b)** Number of significant invariances for each of the first 50 quadratic forms learned in the model system (the functions were sorted by decreasing slowness, see Sect. 4.2). The number of significant invariances tends to decrease with decreasing slowness.

4.7 Relative contribution of the quadratic, linear, and constant term

The inhomogeneous quadratic form has a quadratic, a linear, and a constant term. It is sometimes of interest to know what their relative contribution to the output is. The answer to this question depends on the considered input. For example, the quadratic term dominates for large input vectors while the linear or even the constant term dominates for input vectors with a small norm.

A first possibility is to look at the contribution of the individual terms at a particular point. A privileged point is, for example, the optimal excitatory stimulus, especially if the quadratic form can be interpreted as a feature detector (cf. Sect. 4.4). Figure 4.8a shows for each function in the model system the absolute value of the output of all terms with $\mathbf{x}^+$ as an input. In all functions except the first two, the activity of the quadratic term is greater than that of the linear and of the constant term. The first function basically computes the mean pixel intensity, which explains the dominance of the linear term. The second function is dominated by a constant term from which a quadratic expression very similar to the squared mean pixel intensity is subtracted.

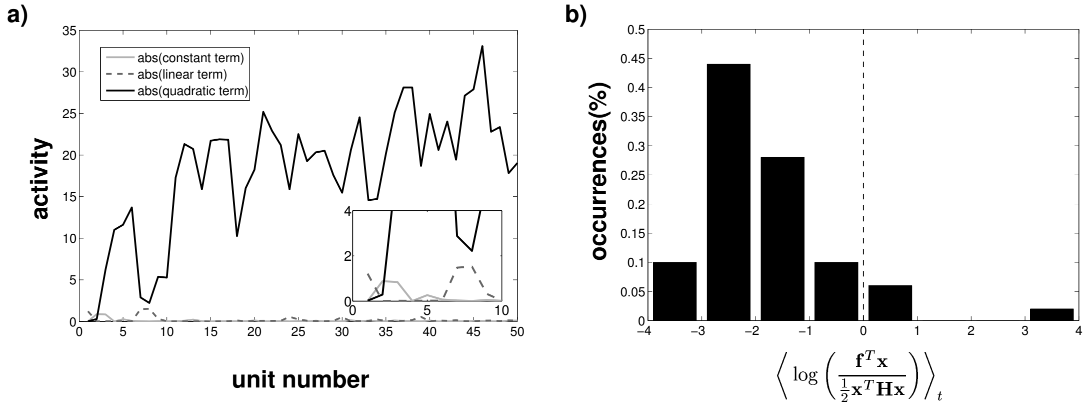


Figure 4.8 Relative contribution of the quadratic, linear, and constant term **(a)** This figure shows the absolute value of the output of the quadratic, linear, and constant term in $\mathbf{x}^+$ for each of the first 50 functions in the model system. In all but the first 2 units the quadratic term has a larger output. The subplot shows a magnified version of the contribution of the terms for the first 10 units. **(b)** Histogram of the mean of the logarithm of the ratio between the activity of the linear and the quadratic term in the model system, when applied to 90,000 test input vectors. A negative value means that the quadratic term dominates while for a positive value the linear term dominates. In all but 4 units (Units 1, 7, 8, and 24) the quadratic term is greater on average.

As an alternative we can consider the ratio between linear and quadratic term, averaged over all input stimuli:

$$\left\langle \log\left|\frac{\mathbf{f}^T\mathbf{x}}{\frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x}}\right|\right\rangle_t = \left\langle \log\left|\mathbf{f}^T\mathbf{x}\right| - \log\left|\frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x}\right|\right\rangle_t . \tag{4.51}$$

The logarithm ensures that a given ratio (e.g. linear/quadratic = 3) has the same weight as the inverse ratio (e.g. linear/quadratic = 1/3) in the mean. A negative result means that the quadratic term dominates while for a positive value the linear term dominates. Figure 4.8b shows the histogram of this measure for the functions in the model system. In all but 4 units (Units 1, 7, 8, and 24) the quadratic term is on average greater than the linear one.

4.8 Quadratic forms without linear term

In the case of a quadratic form without the linear term, i.e.

$$g(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + c \tag{4.52}$$

the mathematics of Sections 4.4 and 4.5 becomes much simpler. The quadratic form is now centered at $\mathbf{x} = \mathbf{0}$, and the direction of maximal increase corresponds to the eigenvector $\mathbf{v}_1$ with the largest positive eigenvalue. The optimal excitatory stimulus $\mathbf{x}^+$ with norm r is thus

$$\mathbf{x}^+ = r\mathbf{v}_1 \, . \tag{4.53}$$

Similarly, the eigenvector corresponding to the largest negative eigenvalue $\mathbf{v}_N$ points in the direction of $\mathbf{x}^-$.

The second derivative (Eq. 4.40) in $\mathbf{x}^+$ in this case becomes

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}(\tilde{g} \circ \boldsymbol{\varphi})(0) = \mathbf{w}^T\mathbf{H}\mathbf{w} - \frac{1}{r^2}\mathbf{x}^{+T}\mathbf{H}\mathbf{x}^+ \tag{4.54}$$

$$\underset{(4.53)}{=} \mathbf{w}^T\mathbf{H}\mathbf{w} - \mathbf{v}_1^T\mathbf{H}\mathbf{v}_1 \tag{4.55}$$

$$= \mathbf{w}^T\mathbf{H}\mathbf{w} - \mu_1 \, . \tag{4.56}$$

The vector $\mathbf{w}$ is by construction orthogonal to $\mathbf{x}^+$ and lies therefore in the space spanned by the remaining eigenvectors $\mathbf{v}_2, \dots, \mathbf{v}_N$. Since μ_1 is the maximum value that $\mathbf{w}^T\mathbf{H}\mathbf{w}$ can assume for vectors of length 1 it is clear that (4.56) is always negative (as it should since $\mathbf{x}^+$ is a maximum) and that its absolute value is successively minimized by the eigenvectors $\mathbf{v}_2, \dots, \mathbf{v}_N$ in this order. The value of the second derivative for $\mathbf{v}_i$ is given by

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}(\tilde{g} \circ \boldsymbol{\varphi})(0) = \mathbf{v}_i^T\mathbf{H}\mathbf{v}_i - \mu_1 \tag{4.57}$$

$$= \mu_i - \mu_1 \, . \tag{4.58}$$

In the same way, the invariances of $\mathbf{x}^-$ are given by $\mathbf{v}_{N-1}, \dots, \mathbf{v}_1$ with second derivative values $(\mu_i - \mu_N)$.

Quadratic forms without the linear term were analyzed in some recent theoretical [Hashimoto, 2003, Bartsch and Obermayer, 2003] and experimental [Touryan et al., 2002] studies. There, the eigenvectors of $\mathbf{H}$ were visualized and interpreted as “relevant features”. Some of them were discarded because they were “unstructured”. According to our analysis, this interpretation only holds for the two eigenvectors with largest positive and negative eigenvalues. We think that the remaining eigenvectors should not be visualized directly but applied as transformations to the optimal stimuli. Therefore it is possible for them to look unstructured but still represent a structured invariance, as illustrated in Figure 4.9. For example, Hashimoto [2003, Fig. 5a] shows in her paper the eigenvectors of a quadratic form learned by a variant of SFA performed by gradient descent. The two largest eigenvectors look like two Gabor wavelets and have the same orientation and frequency. According to the interpretation above and to the cited paper, this shows that the network responds best to an oriented stimulus and is invariant to a phase shift. The third eigenvector looks like a Gabor wavelet with the same frequency as the first two but a slightly different orientation. Hashimoto suggests that the eigenvector makes the interpretation of that particular quadratic form difficult [Hashimoto, 2003, page 777]. According to our analysis, that vector might code for a rotation invariance, which would be compatible with a complex-cell behavior.

Based on spike-triggered average (STA) Rust et al. [2004] computed the best linear approximation of the input-output function computed by some neurons. On the space orthogonal to that function they then determined the best quadratic approximation given by the spike-triggered covariance matrix. They interpreted the eigenvectors corresponding to the eigenvalues with largest absolute value as the basis that spans the subspace of stimuli that governs the response of a cell. Our analysis is consistent with this interpretation, since every stimulus that is generated by a linear combination of the optimal stimulus and the most relevant invariances is going to produce a strong output in the quadratic form.

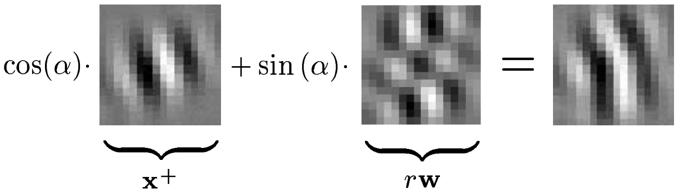


Figure 4.9 Interpretation of the invariances This figure illustrates the fact that although the vector corresponding to an invariance (center) might be difficult to interpret or even look unstructured, when applied to the optimal excitatory stimulus (left) it can code for a meaningful invariance (right). The invariance shown here is the curvature invariance of Fig. 4.6f.

4.9 Decomposition of a quadratic form in a neural network

As also noticed by Hashimoto [2003], for each quadratic form there exists an equivalent two layer neural network, which can be derived by rewriting the quadratic form using its eigenvector decomposition:

$$
\begin{aligned}
g(\mathbf{x}) &= \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x} + c && (4.59)\\
&= \frac{1}{2}\mathbf{x}^T\mathbf{V}\mathbf{D}\mathbf{V}^T\mathbf{x} + \mathbf{f}^T\mathbf{x} + c && (4.60)\\
&= \frac{1}{2}(\mathbf{V}^T\mathbf{x})^T\mathbf{D}(\mathbf{V}^T\mathbf{x}) + \mathbf{f}^T\mathbf{x} + c && (4.61)\\
&= \sum_{i=1}^{N}\frac{\mu_i}{2}(\mathbf{v}_i^T\mathbf{x})^2 + \mathbf{f}^T\mathbf{x} + c\,. && (4.62)
\end{aligned}
$$

One can thus define a neural network with a first layer formed by a set of N linear subunits $s_k(\mathbf{x}) = \mathbf{v}_k^T\mathbf{x}$ followed by a quadratic nonlinearity weighted by the coefficients $\mu_k/2$. The output neuron sums the contribution of all subunits plus the output of a direct linear connection from the input layer (Fig. 4.10a). Since the eigenvalues can be negative, some of the subunits give an inhibitory contribution to the output. It is interesting to note that in an algorithm that learns quadratic forms the number of inhibitory subunits in the equivalent neural network is not fixed but is a learned feature. As an alternative one can scale the weights $\mathbf{v}_i$ by $\sqrt{|\mu_i|/2}$ and specify which subunits are excitatory and which are inhibitory according to the sign of μ_i, since

$$
\begin{aligned}
g(\mathbf{x}) &\underset{(4.62)}{=} \sum_{i=1}^{N}\frac{\mu_i}{2}(\mathbf{v}_i^T\mathbf{x})^2 + \mathbf{f}^T\mathbf{x} + c && (4.63)\\
&= \sum_{\substack{i=1\\ \mu_i>0}}^{N}\left(\left(\sqrt{\frac{|\mu_i|}{2}}\mathbf{v}_i\right)^T\mathbf{x}\right)^2 - \sum_{\substack{i=1\\ \mu_i<0}}^{N}\left(\left(\sqrt{\frac{|\mu_i|}{2}}\mathbf{v}_i\right)^T\mathbf{x}\right)^2 + \mathbf{f}^T\mathbf{x} + c\,. && (4.64)
\end{aligned}
$$

This equation also shows that the subunits are unique only up to an orthogonal transformation (i.e. a rotation or reflection) of the excitatory subunits and another one of the inhibitory subunits, which can be seen as follows. Let $\mathbf{A}^+$ and $\mathbf{A}^-$ be the matrices having as rows the vectors $\sqrt{|\mu_i|/2}\ \mathbf{v}_i$ for positive and negative μ_i, respectively. Equation (4.64) can then be rewritten as

$$
g(\mathbf{x}) = \|\mathbf{A}^+\mathbf{x}\|^2 - \|\mathbf{A}^-\mathbf{x}\|^2 + \mathbf{f}^T\mathbf{x} + c\,. \qquad (4.65)
$$

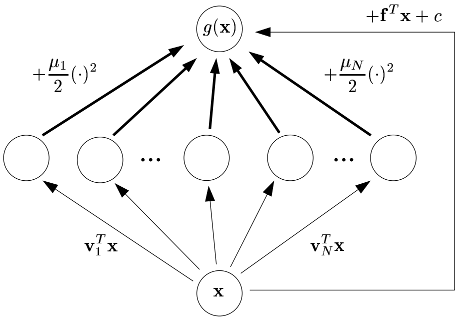

$g(\mathbf{x})$
$+\mathbf{f}^T\mathbf{x} + c$
$+\frac{\mu_1}{2}(\cdot)^2$
$+\frac{\mu_N}{2}(\cdot)^2$
...
...
$\mathbf{v}_1^T\mathbf{x}$
$\mathbf{v}_N^T\mathbf{x}$
$\mathbf{x}$

Figure 4.10 Neural network related to inhomogeneous quadratic forms This figure shows a neural network architecture equivalent to an inhomogeneous quadratic form. The first layer consists of linear subunits, followed by a quadratic nonlinearity weighted by the coefficients $\mu_i/2$. The output neuron sums the contribution of each subunit plus the output of a direct linear connection from the input layer. In this plot, the norm of the subunits is normalized to 1, i.e. $\|\mathbf{v}_i\| = 1$, and the ellipse in the input layer represents a multidimensional input.

Since the length of a vector doesn't change under rotation or reflection, the output of the function remains unchanged if we introduce two orthogonal transformations $\mathbf{R}^+$ and $\mathbf{R}^-$:

$$g(\mathbf{x}) = \|\mathbf{R}^+\mathbf{A}^+\mathbf{x}\|^2 - \|\mathbf{R}^-\mathbf{A}^-\mathbf{x}\|^2 + \mathbf{f}^T\mathbf{x} + c\,. \tag{4.66}$$

Figure 4.11 shows the weights of the subunits of the neural network equivalent to one of the units learned in the model system as defined by the eigenvectors of $\mathbf{H}$ (Eq. 4.62) and some examples of the weights after a random rotation of the excitatory and inhibitory subunits. The subunits are not as structured as in the case of the eigenvectors, although the orientation and frequency can still be identified.

The neural model suggests alternative ways to learn quadratic forms, for example by adapting the weights by backpropagation. The high number of parameters involved, however, could make it difficult for an incremental optimization method to avoid local extrema. On the other hand, each network of this form can be transformed into a quadratic form and analyzed with the techniques described in this chapter, which might be useful for example to compute the optimal stimuli and the invariances.

4.10 Conclusion

In this chapter we have presented a collection of tools to analyze nonlinear functions and in particular quadratic forms. These tools allow us to visualize the coefficients of the individual terms of an inhomogeneous quadratic form, to compute its optimal stimuli (i.e. the stimuli that maximize or minimize the quadratic form under a fixed energy constraint) and their invariances (i.e. the transformations of the optimal stimuli to which the quadratic form is most insensitive), and to determine which of these invariances are statistically significant. We have also proposed a way to measure the relative contribution of the linear and quadratic term. Moreover, we have discussed a neural network architecture equivalent to a given quadratic form. The methods presented here can be used in a variety of fields, in particular in physiological experiments to study the nonlinear receptive fields of neurons and in theoretical studies.

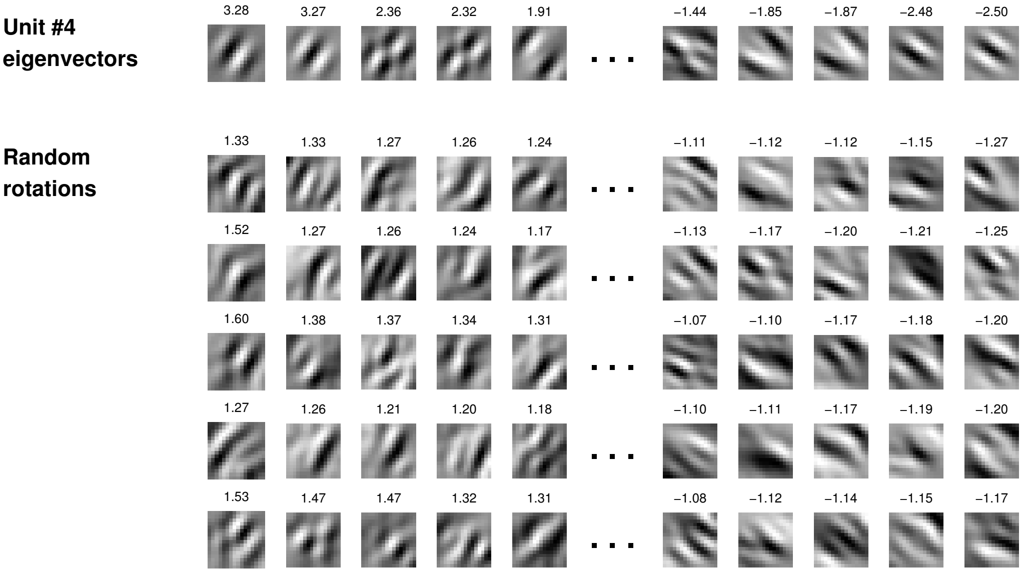


Figure 4.11 Random rotations of the positive and negative subunits The weights of the subunits of Unit 4 as defined by the eigenvectors of $\mathbf{H}$ (Eq. 4.62) and five examples of the weights after a random rotation of the excitatory and inhibitory subunits. The numbers above the patches are the weighting coefficients on the second layer when the weight vectors of the first layer are normalized to 1.

Chapter 5

Pattern recognition with slow feature analysis

5.1 Introduction

As discussed in Chapter 1 a functional model of the sensory cortex is of potential interest for technological applications, since it captures the abstract principles with which our brain performs difficult tasks like object recognition. We have seen in Section 2.1 that temporal slowness can be understood as a principle to learn salient features of time series in a way invariant to frequent transformations [see also Wiskott, 1998]. Such a representation would of course be convenient to perform classification in pattern recognition problems. Most such problems, however, do not have a temporal structure, and it is thus necessary to reformulate the algorithm. The basic idea is to introduce an artificial time structure by constructing a large set of small time series with only two elements chosen from patterns that belong to the same class (Fig. 5.1a). In order to be slowly varying the functions learned by SFA will need to respond similarly to both elements of the time series (Fig. 5.1b) and therefore to ignore the transformation between the individual patterns. As a consequence, patterns corresponding to the same class will cluster in the feature space formed by the output signals of the slowest functions, making it suitable to perform classification with simple techniques such as Gaussian classifiers. It is possible to show that in the ideal case the output of the functions is constant for all patterns of a given class and that the number of relevant functions is small (Sect. 5.2). Notice that this approach does not use any a priori knowledge of the problem. SFA simply extracts the information about relevant features and common transformations by comparing pairs of patterns.

In the next section we adapt the SFA algorithm to the pattern recognition problem and consider the optimal output signal. In Section 5.3 we then apply the proposed method to a handwritten digit recognition problem and discuss the results.

5.2 SFA for pattern recognition

The pattern recognition problem can be summarized as follow. Given C distinct classes $c_1, \ldots, c_C$ and for each class c_m a set of P_m patterns $\mathbf{p}_1^{(m)}, \ldots, \mathbf{p}_{P_m}^{(m)}$ we are requested to learn the mapping $c(\cdot)$ between a pattern $\mathbf{p}_j^{(m)}$ and its class $c(\mathbf{p}_j^{(m)}) = c_m$. We define $P := \sum_{m=1}^{C} P_m$ to be the total number of patterns.

In general in a pattern recognition problem the input data does not have a temporal structure and it is thus necessary to reformulate the definition of the SFA algorithm. Intuitively, we want to obtain a set of

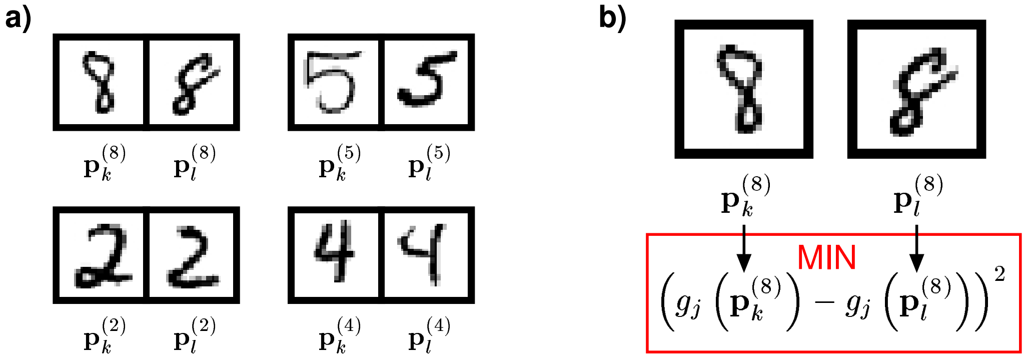

a)
$\mathbf{p}_k^{(8)}$ $\mathbf{p}_l^{(8)}$
$\mathbf{p}_k^{(5)}$ $\mathbf{p}_l^{(5)}$
$\mathbf{p}_k^{(2)}$ $\mathbf{p}_l^{(2)}$
$\mathbf{p}_k^{(4)}$ $\mathbf{p}_l^{(4)}$
b)
$\mathbf{p}_k^{(8)}$ $\mathbf{p}_l^{(8)}$
MIN
$\left(g_j\left(\mathbf{p}_k^{(8)}\right) - g_j\left(\mathbf{p}_l^{(8)}\right)\right)^2$

Figure 5.1 SFA for pattern recognition The input to SFA for pattern recognition is formed by small time series consisting of pairs of patterns that belong to the same class. **(a)** Sample time series in a digit recognition problem. **(b)** In order to be slowly varying the functions learned by SFA must respond similarly to both elements of the time series, as defined in Eq. (5.1).

functions that respond similarly to patterns belonging to the same class. The basic idea is to consider time series of just two patterns $(\mathbf{p}_k^{(m)}, \mathbf{p}_l^{(m)})$, where k and l are two distinct indices in a class c_m (Fig. 5.1).

We can rewrite the slowness objective (2.13) using the mean over all possible pairs obtaining

$$\Delta(y_j) = a \cdot \sum_{m=1}^{C} \sum_{\substack{k,l=1 \\ k<l}}^{P_m} \left(g_j(\mathbf{p}_k^{(m)}) - g_j(\mathbf{p}_l^{(m)}) \right)^2 , \tag{5.1}$$

where the normalization constant a divides by the number of all possible pairs:

$$a = \frac{1}{\sum_{m=1}^{C} \binom{P_m}{2}} . \tag{5.2}$$

We can then reformulate Constraints (2.14)–(2.16) by substituting the average over time with the average over all patterns, so that the learned functions are going to have zero mean, unit variance, and be decorrelated when applied to the whole training data.

$$\frac{1}{P} \sum_{m=1}^{C} \sum_{k=1}^{P_m} g_j(\mathbf{p}_k^{(m)}) = 0 \quad \text{(zero mean)}, \tag{5.3}$$

$$\frac{1}{P} \sum_{m=1}^{C} \sum_{k=1}^{P_m} g_j(\mathbf{p}_k^{(m)})^2 = 1 \quad \text{(unit variance)}, \tag{5.4}$$

$$\forall i < j, \quad \frac{1}{P} \sum_{m=1}^{C} \sum_{k=1}^{P_m} g_i(\mathbf{p}_k^{(m)}) g_j(\mathbf{p}_k^{(m)}) = 0 \quad \text{(decorrelation and order)}. \tag{5.5}$$

Comparing the mathematical formulation of the SFA problem (Eqs. (2.13–2.16) with Equations (5.1, 5.3–5.5) it is possible to see that to respect the new formulation, matrix $\mathbf{A}$ (Eq. 2.39) must be computed using the derivatives of all possible pairs of patterns while matrix $\mathbf{B}$ (Eq. 2.40) using all training patterns just once. Since the total number of pairs increases very fast with the number of patterns it is sometimes necessary to approximate $\mathbf{A}$ with a random subset thereof.

It is clear by (5.1) that the optimal output signal for the slowest functions consists of a signal that is constant for all patterns belonging to the same class, in which case the objective function is zero, as illustrated in Figure 5.2a (see Sect. 5.5.1 for additional remarks). Signals of this kind can be fully represented by a C-dimensional vector, where each component contains the output value for one of the classes. The zero-mean constraint (Eq. 5.3) eliminates one degree of freedom, such that in the end all possible optimal signals span a $(C-1)$-dimensional space. Constraints (5.4) and (5.5) force the output signals to build an orthogonal basis of this space. In a simulation it is therefore possible to extract at most $(C-1)$ such signals. The output signals in the limit case (i.e. when the function space $\mathcal{F}$ is large enough) are thus known in advance. The feature space is very small ($C-1$ dimensions) and consists of C sets of superimposing points (one for each class). In actual applications, of course, the response to the patterns belonging to one class is not exactly constant, but tends to be narrowly distributed around a constant value (Fig. 5.2b). The representation in the $(C-1)$-dimensional feature space will thus consist of C clusters (Fig. 5.2c). As a consequence, classification can be performed with simple methods such as Gaussian classifiers.

For a given input and a fixed function space, the approach proposed above has no parameters (with the exception of the number of derivatives used to estimate $\mathbf{A}$ if the number of patterns per class is too high), which is an important advantage with respect to other algorithms that need to be fine-tuned to the considered problem. Moreover, since SFA is based on an eigenvector problem, it finds the global solution in a single iteration and has no convergence problems, in contrast for example to algorithms based on gradient descent that might get trapped in local minima. On the other hand, SFA suffers from the curse of dimensionality, since the size of the covariance matrices $\mathbf{A}$ and $\mathbf{B}$ that have to be stored during training grows rapidly with increasing dimensionality of the considered function space, which limits the number of input components (see also Sect. 5.3.3).

5.3 Example application

5.3.1 Methods

We illustrate our method by its application to a digit recognition problem. We consider the MNIST digit database, which consists of a standardized and freely available set of 70,000 handwritten digits, divided into a training set (60,000 digits) and a test set (10,000 digits). Each pattern consists of a handwritten digit of size 28×28 pixels (some examples are shown in Fig. 5.1). Several established pattern recognition methods have been applied to this database by LeCun et al. [1998]. Their paper provides a standard reference work to benchmark new algorithms.

We perform SFA on spaces of polynomials of a given degree d, whose corresponding expansion functions $\mathbf{h}$ include all monomials up to order d and have

$$D = \binom{N+d}{d} - 1 \tag{5.6}$$

output dimensions, where N is the number of variables, i.e. the input dimension (see Sect. 5.5.2 for a proof of Eq. 5.6). In this very large space we have to compute the two covariance matrices $\mathbf{A}$ and $\mathbf{B}$, each of which has D^2 component. It is clear that the problem quickly becomes intractable because of the high memory requirements. For this reason, the input dimensionality N is first reduced by principal component analysis (PCA) from $28 \times 28 = 784$ to a smaller number of dimensions (from 10 to 140).

On the preprocessed data we then apply SFA as described in Section 5.2. We compute the covariance matrix $\mathbf{B}$ using all training patterns, and we approximate $\mathbf{A}$ with 1 million derivatives (100,000 derivatives for each digit class), chosen at random without repetition from the set of all derivatives. As explained in Section 5.2, since we have 10 classes only the 9 slowest signals should be relevant. This is confirmed by the Δ-values of the learned functions (Eq. 5.1), which increase abruptly from function g_9 to function g_{10}

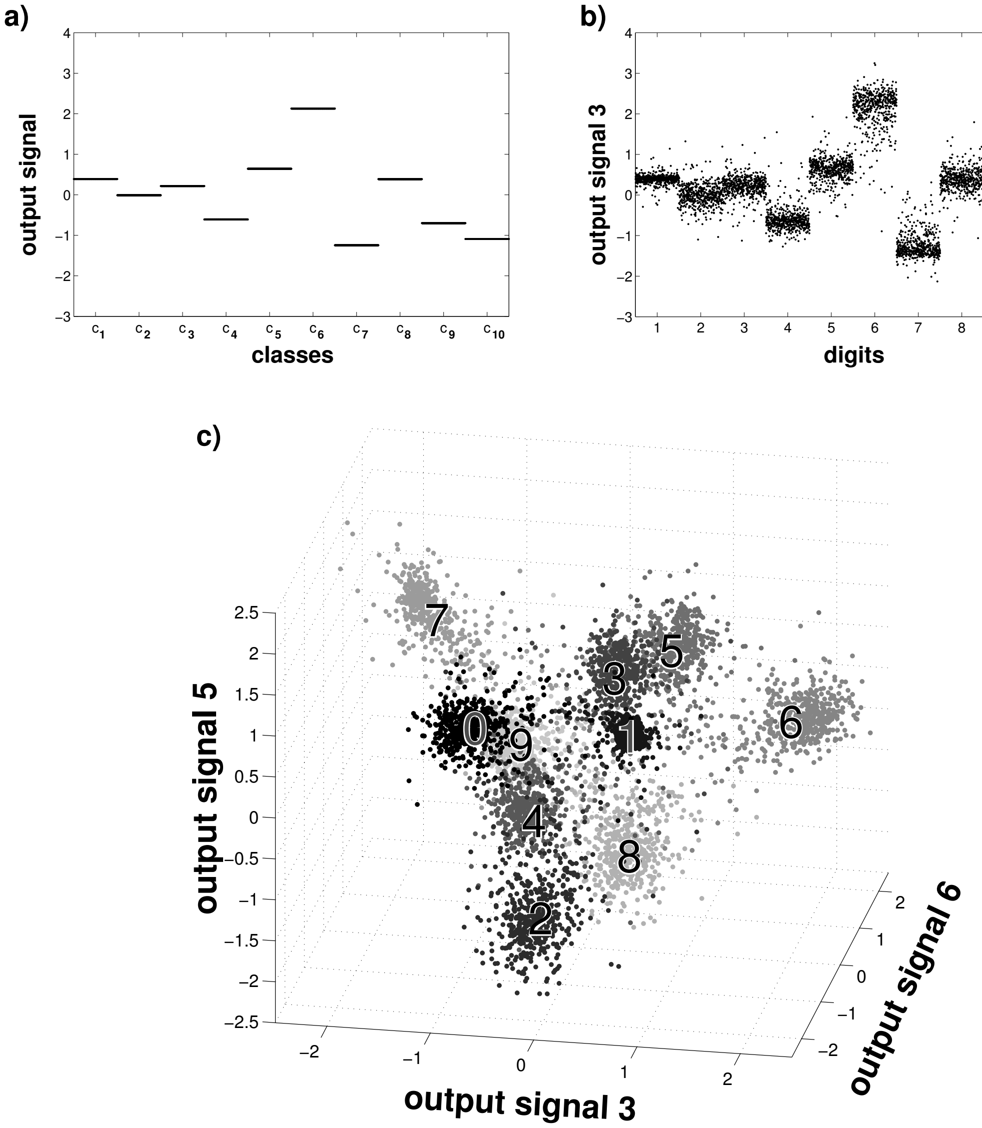


Figure 5.2 Output signals **(a)** In the limit case (i.e. when the considered function space is large enough) the optimal output signal is constant for all patterns of a given class. **(b)** Output signal of one of the functions learned in the best among the simulations of Sect. 5.3, performed on the MNIST database using polynomials of degree 3 and 35 input dimensions. In this plot the function was applied to 500 test digits for each class. Its output for a specific class is narrowly distributed around a constant value. **(c)** Feature space representation given by 3 output signals in the same simulation as in (b). Each class forms a distinct cluster of points.

(Fig. 5.3). The increase factor $\Delta(y_{10})/\Delta(y_9)$ in our simulations was on average 3.7 . For this reason, we only need to learn the 9 slowest functions.

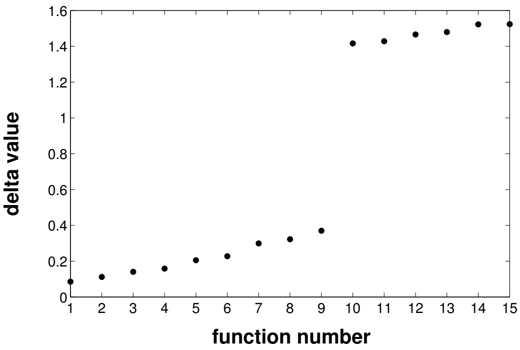


Figure 5.3 Δ-values Δ-values (Eq. 5.1) of the first 15 functions in the simulation with polynomials of degree 3 and 35 input dimensions. As explained in Sect. 5.2, since there are 10 classes only the 9 slowest signals should be relevant. This is confirmed by the abrupt increase in temporal variability at function number 10.

For classification, we apply the 9 functions to the training data of each class separately and fit a Gaussian distribution to their output by computing mean and covariance matrix. In this way we obtain 10 Gaussian distributions, each of which represents the probability $P(\mathbf{y}|c_m)$ of an output vector $\mathbf{y}$ given the class c_m. We then consider the test digits and compute for each of them the probability to belong to each of the classes

$$P(c_m|\mathbf{y}) = \frac{P(\mathbf{y}|c_m)P(c_m)}{\sum_{j=1}^{C} P(\mathbf{y}|c_j)}\,, \tag{5.7}$$

where $P(c_m) = P_m/P$ is the probability of occurring of class c_m. Finally, we assign the test digit to the class with the highest probability.

5.3.2 Results

Figure 5.4 shows the training and test errors for experiments performed with polynomials of degree 2 to 5 and number of input dimensions from 10 to 140. With polynomials of second degree the explosion in the dimensionality of the expanded space with increasing number of input dimensions is relatively restricted, such that it is possible to perform simulations with up to 140 dimensions, which explain 94% of the total input variance. The test error settles down quickly around 2%, with a minimum at 100 dimensions (1.95%). Simulations performed with higher order polynomials have to rely on a smaller number of input dimensions, but since the function space gets larger and includes new nonlinearities, one obtains a remarkable improvement in performance. Given a fixed dimensionality, the error rate decreases monotonically with increasing degree of the polynomials (Fig. 5.4).

The best performance in our simulations is achieved with polynomials of degree 3 and 35 input dimensions, with an error rate of 1.5% on test data. Figure 5.2b shows the output of one of the functions learned in this simulation when applied to 500 test digits for each class. For each individual class the signal is narrowly distributed around a constant value, and approximates the optimal solution shown in Figure 5.2a. Some of the digit classes (e.g. 1, 2, and 3) have similar output values and cannot be distinguished by looking at this signal alone. However, other functions represent those classes in different ways and considering all

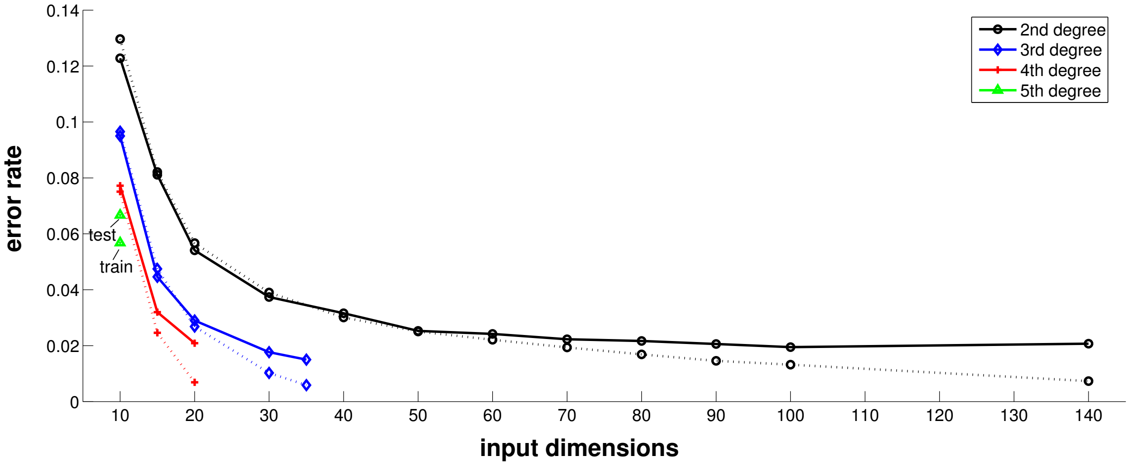


Figure 5.4 Test and training error Error rates for simulations performed with polynomials of degree 2 to 5 and number of input dimensions from 10 to 140. Dotted and solid lines represent training and test error rate, respectively. For polynomials of degree 5 the error rate for test and training data are indicated by a label.

signals together it is possible to separate them. For example, Figure 5.2c shows the feature space representation given by 3 output signals. It is possible to see that each class forms a distinct cluster. Considering the whole 9-dimensional feature space improves the separation further. Figure 5.5 shows all 146 digits that were misclassified out of the 10,000 test patterns. Some of them seem genuinely ambiguous, while others would have been classified correctly by a human. The reduction of the input dimensionality by PCA is partly responsible for these error, since it erases some of the details and makes some patterns difficult to recognize, as illustrated in Figure 5.6.

Table 5.1 compares the error rates on test data of different algorithms presented in [LeCun et al., 1998] with that of SFA. The error rate of SFA is comparable to but does not outperform that of the most elaborate algorithms. Its performance is however remarkable considering the simplicity of the method and the fact that it has no a priori knowledge on the problem, in contrast for example to the LeNet-5 algorithm which has been designed specifically for handwritten character recognition. In addition, for recognition, our method has to store and compute only 9 functions and has thus small memory requirements and a high recognition speed.

The comparison with the Tangent Distance (TD) algorithm [Simard et al., 1993, LeCun et al., 1998, Simard et al., 2000] is particularly interesting. TD is a nearest-neighbor classifier where the distance between two patterns is not computed with the Euclidean norm but with a metric that is made insensitive to local transformations of the patterns which have to be specified a priori. In the case of digit recognition, they might include for example local translation, rotation, scaling, stretching, and thickening or thinning of the image. This method, as most memory-based recognition systems, has very high memory and computational requirements. One can interpret SFA as an algorithm that learns a nonlinear transformation of the input such that the Euclidean distance between patterns belonging to the same class gets as small as possible. A natural objective function for this would be, defining $\mathbf{g}$ as the vector of all learned functions

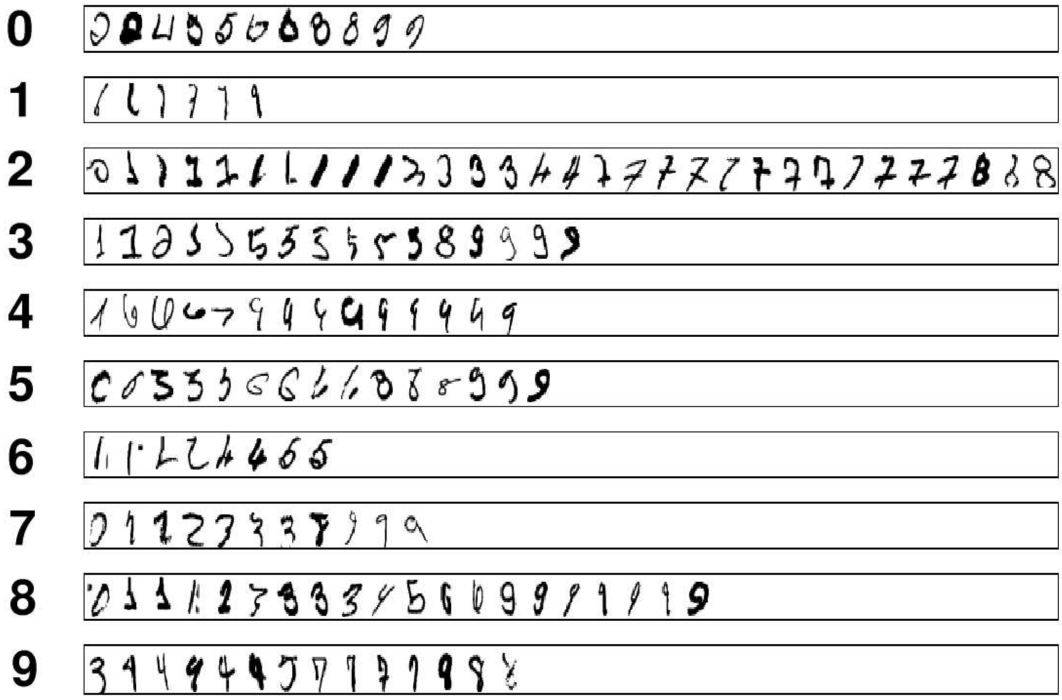


Figure 5.5 Classification errors This figure shows all the 146 digits that were misclassified out of the 10,000 test patterns in the best among our simulations performed with polynomials of degree 3 and 35 input dimensions. The patterns in the first line were classified as 0, the ones in the second line as 1, etc.

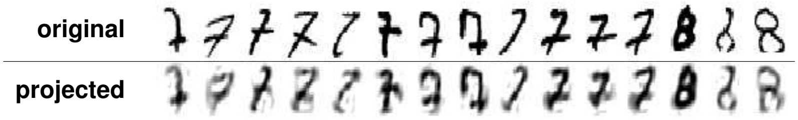


Figure 5.6 Dimensionality reduction The reduction by PCA of the number of input dimensions might be responsible for some of the classification errors. This figure shows some of the test patterns that have been misclassified as 2. In the top row we plot the original patterns from the MNIST database. In the bottom row we plot the same patterns after a projection onto the first 35 principal components. Due to dimensionality reduction some of the patterns become more ambiguous.

METHOD	% ERRORS
Linear classifier	12.0
K-Nearest-Neighbors	5.0
1000 Radial Basis Functions, linear classifier	3.6
Best Back-Propagation NN (3 layers with 500 and 150 hidden units)	2.95
Reduced Set SVM (5 deg. polynomials)	1.0
LeNet-1 (16 × 16 input)	1.7
LeNet-5	0.95
Tangent Distance (16 × 16 input)	1.1
Slow Feature Analysis (3 deg. polynomials, 35 input dim)	**1.5**

Table 5.1 Performance comparison Error rates on test data of various algorithms. All error rates are taken from [LeCun et al., 1998].

$\mathbf{g} = (g_1, \dots g_K)$:

$$a \cdot \sum_{m=1}^{C} \sum_{\substack{k,l=1 \\ k<l}}^{P_m} \left\| \mathbf{g}(\mathbf{p}_k^{(m)}) - \mathbf{g}(\mathbf{p}_l^{(m)}) \right\|^2 \tag{5.8}$$

$$= a \cdot \sum_{m=1}^{C} \sum_{\substack{k,l=1 \\ k<l}}^{P_m} \left[\sum_{j=1}^{K} \left(g_j(\mathbf{p}_k^{(m)}) - g_j(\mathbf{p}_l^{(m)}) \right)^2 \right] \tag{5.9}$$

$$= \sum_{j=1}^{K} \left[a \cdot \sum_{m=1}^{C} \sum_{\substack{k,l=1 \\ k<l}}^{P_m} \left(g_j(\mathbf{p}_k^{(m)}) - g_j(\mathbf{p}_l^{(m)}) \right)^2 \right] \tag{5.10}$$

$$\underset{(5.1)}{=} \sum_{j=1}^{K} \Delta(y_j)\,, \tag{5.11}$$

which has to be minimized under Constraints (2.14–2.16). For a given K the objective function (5.11) and that of SFA for pattern recognition (5.1) are identical, except that in the former case the average of the temporal variation over all K learned functions is minimized, while in the latter the functions are optimized one after the other inducing an order. The set of functions that minimizes the SFA objective function (Eq. 5.1) also minimizes Equation (5.11). Furthermore, all other solutions to (5.11) are orthogonal transformations of this special one (a sketch of the proof is given in Sect. 5.5.3). The solution found by SFA is particular in that it is independent from the choice of K, in the sense that the components are ordered such that it is possible to compute the global solution first and then reduce the number of dimensions afterward as needed.

In conclusion, while the TD algorithm keeps the input representation fixed and applies a special metric, SFA transforms the input space such that the TD-metric gets similar to the Euclidean one. As already mentioned, the new representation has only few dimensions (since K in Eq. 5.11 can be set to $K = C - 1$, see Sect. 5.2), which decreases memory and computational requirements dramatically, and can be easily learned from the input data without a priori knowledge on the transformations.

5.3.3 Extensions

In this example application we tried to keep the pattern recognition system as basic as possible in order to show the simplicity and effectiveness of SFA. Of course, a number of standard techniques could be applied to improve its performance. A trivial way to improve the performance would be to use a computer with more memory and perform SFA with a higher number of input dimensions and/or polynomials of higher degree. This approach would however rapidly reach its limits, since the dimensionality of the matrices $\mathbf{A}$ and $\mathbf{B}$ grows exponentially, as discussed above. An important improvement to our method would be to find an algorithm that performs a preliminary reduction of the dimensionality of the expanded space by discarding directions of high temporal variation. Note that it is theoretically not possible to reduce the dimensionality of the expanded space by PCA or by any other method that does not take into account the derivatives of the expanded signal, as suggested in [Bray and Martinez, 2002] and [Hashimoto, 2003], since there is in general no simple relation between the spatial and the temporal statistics of the expanded signal (in the case of PCA the solution would even depend on the scaling of the basis functions). If we consider for example the simulation with polynomials of degree 3 and 35 input dimensions and reduce the dimensionality of the expanded space by PCA down to one half (which is still an optimistic scenario since in general the reduction has to be more drastic), the error rate shows a substantial increase from 1.5% to 2.5%. A viable solution if the number of patterns in the data set is small (which is not the case in our example application nor usually in real-life problems) is to use the standard *kernel trick* and compute the covariance matrices on the temporal dimensions instead of in space [see Müller et al., 2001, Bray and Martinez, 2002]. In this case it is even possible to use an infinite-dimensional function space if it permits a kernel function (i.e. if it satisfies Mercer's Condition, see [Burges, 1998]).

Other standard machine learning methods that have been applied to improve the performance of pattern recognition algorithms might be applied in our case as well, like for example a more problem-specific preprocessing of the patterns, boosting techniques, or mixture of experts with other algorithms [Bishop, 1995, LeCun et al., 1998]. Some of the simulations in [LeCun et al., 1998] were performed on a training set artificially expanded by applying some distortions to the original data, which improved the error rate. Such a strategy might be particularly effective with SFA since it might help to better estimate the covariance matrix $\mathbf{A}$ of the transformations between digits.

Finally, the Gaussian classifier might be substituted by some more enhanced supervised classifier, although we would not expect a particularly dramatic improvement in performance, due to the simple form taken by the representation in the feature space both in the limit case and in simulations (Fig. 5.2).

5.4 Conclusions

In this chapter we described the application of SFA to pattern recognition. The presented method is problem-independent and for a given input signal and a fixed function space it has no parameters. In an example application it yields an error rate comparable to that of other established algorithms. The learned feature space has a very small dimensionality, such that classification can be performed efficiently in terms of speed and memory requirements.

Since most pattern recognition problems do not have a temporal structure it is necessary to present the training set in the way described in Section 5.2. If instead the input does have a temporal structure (e.g. when classifying objects that naturally enter and leave a visual scene), it is possible to apply SFA directly on the input data [cf. Wallis and Rolls, 1997]. By applying more elaborated unsupervised clustering techniques for classification it should be thus possible to achieve totally unsupervised pattern recognition (i.e. without supervision from perception to classification).

5.5 Technical remarks to Chapter 5

5.5.1 Input sequences in the limit case

In the limit case (i.e. when the function space is large enough) one would obtain the optimal signals also by just showing C sequences, each consisting of all patterns $\mathbf{p}_1^{(m)}, \dots, \mathbf{p}_{P_m}^{(m)}$ belonging to one class. In general, however, if SFA cannot produce a perfectly constant solution it could find and exploit some structure in the particular patterns succession used during training, which would in turn create some artificial structures in the output signals. For this reason one obtains better results by presenting the patterns in all same-class pairs as mentioned above.

5.5.2 Dimensionality of a space of polynomials

The space of polynomial of degree d has

$$D = \binom{N+d}{d} - 1 \tag{5.12}$$

dimensions. This can be seen as follow: The number of monomials of degree i is equal to the number of combinations of length i on N elements with repetition, i.e.

$$\binom{N+i-1}{i} . \tag{5.13}$$

The number of dimensions is thus

$$\sum_{i=0}^{d} \binom{N+i-1}{i} - 1 , \tag{5.14}$$

where we subtracted the constant term $i = 0$, which is fixed by the zero-mean constraint (5.3). Equation (5.12) is then easily proved by induction.

5.5.3 The solutions of Equation 5.11

In this section we prove that the functions $g_1, \dots, g_K$ that minimize Equation (5.11) under the zero mean, unit variance, and decorrelation constraints (Eqs. 2.14–2.16) are identical up to an orthogonal transformation with the SFA solutions that minimize the slowness objective (Eq. 2.13) under the same constraints.

Without loss of generality we consider only the linear case

$$g_j(\mathbf{x}) = \mathbf{w}_j^T \mathbf{x} \tag{5.15}$$

and we assume that the input vectors $\mathbf{x}$ are whitened, such that $\mathbf{B} = \langle \mathbf{x}\mathbf{x}^T \rangle_t = \mathbf{I}$, which can be obtained with an orthogonal transformation. Under this conditions Constraints (2.14–2.16) simply mean that the vectors $\mathbf{w}_j$ must be orthogonal and have norm 1 (cf. Eqs. 2.29–2.30). Moreover, the generalized eigenvalue problem (2.28) becomes a standard eigenvalue problem

$$\mathbf{A}\mathbf{W} = \mathbf{W}\boldsymbol{\Lambda} \tag{5.16}$$

and the SFA solutions correspond to the eigenvectors with the smallest eigenvalues. Moreover, Equation (5.11) can be written as

$$\sum_{j=1}^{K} \Delta(y_j) \underset{(2.23)}{=} \sum_{j=1}^{K} \mathbf{w}_j^T \mathbf{A} \mathbf{w}_j . \tag{5.17}$$

Minimizing this equation involves computing the Lagrange multipliers of (5.17) under the conditions

$$\mathbf{w}_i^T \mathbf{w}_j = \delta_i^j \,, \quad \forall i, j < K \,, \tag{5.18}$$

where δ_i^j is the Kronecker delta. The complete calculation is reported in [Bishop, 1995, Appendix E] for an equivalent PCA theorem. The solution is found to be an arbitrary orthogonal transformation of the eigenvectors of $\mathbf{A}$ corresponding to the smallest eigenvalues, i.e. an arbitrary orthogonal transformation of the special SFA solution.

Chapter 6

Conclusion and future directions

A first objective of this thesis was to provide evidence of the relevance of temporal slowness as a principle for self-organization of the sensory cortex. The results of Chapter 3 show that slow feature analysis applied to natural image sequences reproduces many properties of complex cells in V1, not only the two basic ones, namely a Gabor-like optimal stimulus and phase-shift invariance, but also secondary ones like direction selectivity, non-orthogonal inhibition, end-inhibition, and side-inhibition. Earlier studies have shown that the *what* and *where* information (i.e. identity and position) can be learned for whole objects in a hierarchical network of SFA modules [Wiskott and Sejnowski, 2002, Wiskott, 2003b]. It is remarkable that the slowness principle is able to extract relevant information at such different levels of representation. These results make temporal slowness a good candidate as a computational principle for the sensory cortex.

The slowness hypothesis will need to be verified in greater detail in future research. First of all, a necessary condition for a computational principle to be a realistic model for the self-organization of the sensory cortex is that it must be realizable by a plausible physiological mechanism. Preliminary theoretical considerations suggest that SFA can be implemented with spiking neurons using a spike timing dependent plasticity rule (Michaelis and Wiskott, personal communication). Further, the hypothesis should be falsifiable by predictions at different levels [Hurri, 2003]: i) qualitative, ii) quantitative with respect to earlier physiological measurements, and iii) quantitative with respect to new physiological measurements. We addressed the first two points in Chapter 3 and found a good match between the set of slowly varying functions and the population of complex cells in V1. Regarding the third point, our model predicts a relation between the behavior of a neuron and the slowness of its output (Sect. 3.3), although this relation is not particularly strong.

Some experiments can be suggested to collect further evidence:

- One could record the response of the units learned by SFA and of complex cells in V1 to given image sequences. The output of the SFA units could then be used as a predictor for the response of complex cells [cf. the use of the output of Fourier filters as predictors in Theunissen et al., 2001]. A good fit would indicate that complex cells lie in the subspace spanned by the slowest functions and are thus slowly varying themselves.

- The model presented in Chapter 3 could be extended to one or more successive layers of SFA modules, corresponding to higher visual areas. Neurons in those areas respond to features of increasing complexity [for an overview see Oram and Perrett, 1994], but their function is still poorly understood, mostly because a systematic exploration of the input space with physiological experiments is difficult. The results of a hierarchical SFA model might provide some predictions about this function and suggest further physiological experiments.

- Finally, SFA could be applied to other sensory modalities, like for example on the auditory signals coming from cochlear filters. Should the properties of the learned function be similar to those of neurons in the corresponding cortical area, this would prove the generality of the temporal slowness principle for the whole sensory cortex.

In Chapter 4 we introduced some methods to analyze and interpret quadratic forms as receptive fields, including an algorithm to compute the optimal excitatory and inhibitory stimuli (i.e. the stimuli that maximize and minimize a quadratic form, respectively, given a fixed energy constraint), an algorithm to compute and visualize the invariances of the optimal stimuli, which are the transformations to which the quadratic form is most insensitive, and a test to determine which of these are statistically significant. Moreover, we defined two measures to determine the relative contribution of the different terms of a quadratic form to its output and presented a two-layer neural network architecture equivalent to a given quadratic form. As mentioned in Section 4.1 it should be straightforward to extend most of these methods to other nonlinear functions while preserving the same interpretations, so that they should be applicable in general for the analysis of physiological receptive fields. It will be interesting to see if they will provide new insights on the behavior of cortical neurons.

Another purpose of this thesis was to show the potentialities of temporal slowness for engineering applications. In Chapter 5 we presented a way to apply SFA to pattern recognition problems by defining an artificial temporal structure on the patterns, and showed that the optimal solution in this context consists of a signal that is constant for all patterns belonging to the same class. In the simulations of Section 5.3 this method achieved a good performance on a digit recognition problem, although its error rate did not outperform that of more elaborate algorithms. However, an advantage with respect to those is that our system uses no a priori knowledge on the problem and has to store and compute only a small number of functions, so that it has small memory requirements and a high recognition speed. Some extensions to this basic system were proposed in Section 5.3.3. One particularly important improvement to SFA would be to find an efficient algorithm to perform a preliminary dimensionality reduction in the expanded space, which would allow to perform simulations with higher order polynomials and a larger number of input components.

Appendix A

Additional material

Scientific results should be readily available and easily reproducible. In the case of studies based on computer models and simulations this is particularly feasible and at the same time fundamental. The publication of the source code serves three important purposes: i) to make the results reproducible; ii) to allow to perform additional experiments, for example in order to further explore the parameters space; iii) to provide an explicit and exact description of the applied algorithm.

To complete this thesis we published following software and data online:

- Chapter 2: The SFA algorithm has been made available in two software packages: the `sfa-tk` Matlab toolbox [Berkes, 2004] and the MDP Python library [Berkes and Zito, 2004].
- Chapter 3: The site `http://itb.biologie.hu-berlin.de/~berkes/slowness` contains detailed single-unit analysis, additional population statistics, larger versions of Figures 3.3 and 3.13, and Matlab functions to perform simulations like those described in that chapter.
- Chapter 4: The animations corresponding to Figure 4.6 and Matlab source code for the algorithms of Table 4.2 and 4.3 are available at
`http://itb.biologie.hu-berlin.de/~berkes/qforms` .
- Chapter 5: Python source code and data to reproduce the simulations are available at
`http://itb.biologie.hu-berlin.de/~berkes/pattern_recognition` .

Anhang B

Zusammenfassung in deutscher Sprache

B.1 Einleitung

Ein möglicher Ansatz zum Verständnis des sensorischen Kortex besteht darin, die seiner Funktion und Selbstorganisation zu Grunde liegenden Prinzipien zu untersuchen. Dieser Ansatz hat in den letzten Jahrzehnten in Folge des Bedarfs an theoretischen Grundlagen für die Motivation und Interpretation von Experimenten zunehmend an Wichtigkeit gewonnen. Als grundlegende Prinzipien wurden *Kompaktheit*, *Unabhängigkeit*, *Spärlichkeit* und *zeitliche Langsamkeit* vorgeschlagen. Ein attraktiver Aspekt vieler theoretischer Modelle sensorischer Verarbeitung ist ihre Relevanz für technische Anwendungen. Ziel dieser Doktorarbeit ist die Untersuchung von zeitlicher Langsamkeit als Prinzip für die Selbstorganisation des sensorischen Kortex sowie für die Mustererkennung.

B.2 Das Prinzip der zeitlichen Langsamkeit

Das Prinzip der zeitlichen Langsamkeit basiert auf der Beobachtung, dass sensorische Signale aus direkten Messungen unserer physikalischen Umwelt entstehen und deshalb im allgemeinen sehr empfindlich gegen kleine Veränderungen der Umgebung oder des Beobachters sind. Im Gegensatz dazu sind die für uns wesentlichen Merkmale der Umgebung, etwa Identität oder Position von Objekten im visuellen Umfeld, stabil und variieren auf einer sehr viel langsameren Zeitskala. Wenn es also gelingt, langsam veränderliche Merkmale aus den schnell variierenden sensorischen Signalen zu extrahieren, so ist es wahrscheinlich, dass diese wesentliche Eigenschaften unserer Umwelt repräsentieren. Zudem sind sie invariant oder zumindest robust gegenüber typischen Transformationen der sensorischen Eingangssignale, z.B. gegenüber Translationen, Rotationen oder Zoom von visuellen Eindrücken. Kapitel 2 gibt eine Einführung und Diskussion des Prinzips der zeitlichen Langsamkeit sowie einen Überblick über vorhergehende Arbeiten zu diesem Ansatz. Im Abschnitt 2.3 wird die von uns verwendete Formulierung von Wiskott und Sejnowski (2002) dargestellt und der entsprechende Algorithmus *Slow Feature Analysis* (SFA) eingeführt.

B.3 Zeitliche Langsamkeit als Modell für die Selbstorganisation rezeptiver Felder im primären visuellen Kortex

In Kapitel 3 untersuchen wir zeitliche Langsamkeit als Lernprinzip für rezeptive Felder im visuellen Kortex. Unter Verwendung von SFA werden Transformationsfunktionen gelernt, die, angewendet auf natürliche Bildsequenzen, möglichst langsam variierende Merkmale extrahieren. Die Funktionen können als nichtlineare raum-zeitliche rezeptive Felder interpretiert und mit Neuronen im primären visuellen Kortex (V1) verglichen werden. Wir zeigen, dass sie viele Eigenschaften von komplexen Zellen in V1 besitzen, nicht nur die primären, d.h. Gabor-ähnliche optimale Stimuli und Phaseninvarianz, sondern auch sekundäre, wie Richtungsselektivität, nicht-orthogonale Inhibition sowie End- und Seiteninhibition. Dies wird qualitativ und anhand eines quantitativen Vergleichs mit der Populationsstatistik von komplexen Zellen in V1 nachgewiesen. Anhand einer Reihe von Kontrollexperimenten untersuchen wir zudem, welche Rolle die statistischen Eigenschaften natürlicher Bilder und die unterschiedlichen Transformationen, die zur Erstellung der Bildsequenzen verwendet werden, für die Ergebnisse spielen.

B.4 Analyse und Interpretation von quadratischen Formen als rezeptive Felder

Für die Analyse der mit SFA gelernten nichtlinearen Funktionen haben wir eine Reihe mathematischer und numerischer Werkzeuge entwickelt, mit denen man die quadratischen Formen als rezeptive Felder charakterisieren kann. Diese Techniken sind auch von allgemeinerem Interesse für Approximationen zweiter Ordnung und theoretische Modelle physiologischer rezeptiver Felder. In Kapitel 4 stellen wir zwei Algorithmen vor, um die optimalen Stimuli und deren Invarianzen zu berechnen, und einen Test, um festzustellen, welche von diesen statistisch signifikant sind. Außerdem definieren wir zwei Maße, mit denen die Beiträge des linearen und quadratischen Terms verglichen werden können. Wir stellen schließlich ein neuronales Netz vor, das zu einer gegebenen quadratischen Form äquivalent ist.

B.5 Mustererkennung mit Slow Feature Analysis

Den Abschluss dieser Arbeit bildet die Anwendung des Prinzips der zeitlichen Langsamkeit auf Mustererkennungsprobleme. Die fehlende zeitliche Struktur in dieser Problemklasse erfordert eine Modifikation des SFA-Algorithmus. Wir stellen eine entsprechende alternative Formulierung vor und präsentieren die optimalen Lösungen. Anschließend wenden wir dieses System auf eine Standard-Datenbank von handgeschriebenen Ziffern an.

B.6 Schlussfolgerung

Eines der Ziele dieser Arbeit war, zu zeigen, dass zeitliche Langsamkeit ein leistungsfähiges Prinzip für die Selbstorganisation des sensorischen Kortex sein kann. Die Resultate von Kapitel 3 belegen, dass SFA angewendet auf natürliche Bildsequenzen viele Eigenschaften von komplexen Zellen in V1 reproduziert. Frühere Studien haben gezeigt, dass Identität und Position für ganze Objekte in einem hierarchischen Netz von SFA-Modulen erlernt werden können. Es ist bemerkenswert, dass das Langsamkeitsprinzip in der Lage ist, relevante Informationen auf so unterschiedlichen Darstellungsniveaus zu extrahieren. Zusätzliche Experimente können vorgeschlagen werden, um die Langsamkeithypothese weiter zu prüfen. Ein anderer Zweck dieser Doktorarbeit war, das Potenzial der zeitlichen Langsamkeit für technische Anwendungen zu untersuchen. Die Simulationen von Kapitel 5 bestätigen, dass SFA leicht und erfolgreich für Mustererkennungsprobleme verwendet werden kann.

Bibliography

E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal Optical Society of America A*, 2(2):284–299, 1985.

J. Atick and A. Redlich. What does the retina know about natural scenes? *Neural Computation*, 4:196–210, 1992.

J. Atick, L. Zhaoping, and A. Redlich. Understanding retinal color coding from first principles. *Neural Computation*, 4:559–572, 1992.

F. Attneave. Informational aspects of visual perception. *Psychological Review*, 61:183–193, 1954.

R. Baddeley, L. Abbott, M. Booth, F. Sengpiel, T. Freeman, E. Wakeman, and E. Rolls. Responses of neurons in primary and inferior temporal visual cortices to natural scenes. *Proc R Soc Lond B Biol Sci.*, 264(1389):1775–83, 1997.

H. Barlow. Possible principles underlying the transformations of sensory messages. In W. Rosenblith, editor, *Sensory Communication*, pages 217–234. MIT Press, 1961.

H. Barlow. The exploitation of regularities in the environment by the brain. *Behavioral and Brain Sciences*, 24(4):602–607, 2001.

H. Bartsch and K. Obermayer. Second-order statistics of natural images. *Neurocomputing*, 52–54:467–472, 2003.

S. Becker. Mutual information maximization: Models of cortical self-organization. *Network: Computation in Neural Systems*, 7:7–31, 1996.

S. Becker and G. E. Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355:161–163, 1992.

A. J. Bell and T. J. Sejnowski. The ‘independent components’ of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

A. Belouchrani, K. Abed Meraim, J.-F. Cardoso, and E. Moulines. A blind source separation technique based on second order statistics. *IEEE Transactions on Signal Processing*, 45(2):434–444, 1997.

P. Berkes. sfa-tk: Slow feature analysis toolkit for Matlab (v.1.0.1). http://itb.biologie.hu-berlin.de/˜berkes/software/sfa-tk/sfa-tk.shtml, 2004.

P. Berkes and L. Wiskott. Applying slow feature analysis to image sequences yields a rich repertoire of complex cell properties. In J. R. Dorronsoro, editor, *Artificial Neural Networks - ICANN 2002 Proceedings*, Lecture Notes in Computer Science, pages 81–86. Springer, 2002.

P. Berkes and L. Wiskott. Slow feature analysis yields a rich repertoire of complex-cell properties. Cognitive Sciences EPrint Archive (CogPrint) 2804, http://cogprints.ecs.soton.ac.uk/archive/00002804/, 2003.

P. Berkes and L. Wiskott. On the analysis and interpretation of inhomogeneous quadratic forms as receptive fields. Cognitive Sciences EPrint Archive (CogPrints) 4081, http://cogprints.org/4081/, 2005a.

P. Berkes and L. Wiskott. Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision*, 2005b. (accepted).

P. Berkes and T. Zito. Modular toolkit for Data Processing (MDP). http://mdp-toolkit.sourceforge.net/, 2004.

C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

T. Blaschke, L. Wiskott, and P. Berkes. What is the relation between independent component analysis and slow feature analysis? (submitted), 2004.

A. Bray and D. Martinez. Kernel-based extraction of Slow Features: Complex cells learn disparity and translation invariance from natural images. In *NIPS 2002 proceedings*, 2002.

C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

O. Creutzfeldt and H. Nothdurft. Representation of complex visual stimuli in the brain. *Naturwissenschaften*, 65(6):307–318, 1978.

P. Dayan and L. F. Abbott. *Theoretical neuroscience: computational and mathematical modeling of neural systems*. The MIT Press, 2001.

R. De Valois, D. Albrecht, and L. Thorell. Spatial frequency selectivity of cells in macaque visual cortex. *Vision Res.*, 22:545–559, 1982a.

R. De Valois, E. Yund, and N. Hepler. The orientation and direction selectivity of cells in macaque visual cortex. *Vision Res.*, 22(5):531–44, 1982b.

G. DeAngelis, R. Freeman, and I. Ohzawa. Length and width tuning of neurons in the cat's primary visual cortex. *Journal of Neurophysiology*, 71(1):347–374, 1994.

A. Dobbins, S. W. Zucker, and M. S. Cynader. Endstopped neurons in the visual cortex as a substrate for calculating curvature. *Nature*, 329:438–441, October 1987.

D. Dong and J. Atick. Temporal decorrelation: a theory of lagged and nonlagged responses in the lateral geniculate nucleus. *Network: Computation in Neural Systems*, 6:159–178, 1995a.

D. W. Dong and J. J. Atick. Statistics of natural time-varying images. *Network: Computation in Neural Systems*, 6(3):354–358, 1995b.

V. Dragoi, C. Rivadulla, and M. Sur. Foci of orientation plasticity in visual cortex. *Nature*, 411(6833): 80–86, 2001.

W. Einhäuser, C. Kayser, K. Körding, and P. König. Learning multiple feature representation from natural image sequences. In J. R. Dorronsoro, editor, *Artificial Neural Networks - ICANN 2002 Proceedings*, Lecture Notes in Computer Science, pages 21–26. Springer, 2002.

D. J. Field. What is the goal of sensory coding? *Neural Computation*, 6(4):559–601, 1994.

P. Földiák. Learning invariance from transformation sequences. *Neural Computation*, 3:194–200, 1991.

C. Fortin. A survey of the trust region subproblem within a semidefinite framework. Master's thesis, University of Waterloo, 2000.

K. Fukushima. Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–0202, 1980.

Gantmacher. *Matrix Theory*, volume 1. AMS Chelsea Publishing, 1959.

M. S. Gizzi, E. Katz, R. A. Schumer, and J. A. Movshon. Selectivity for orientation and direction of motion of single neurons in cat striate and extrastriate visual cortex. *Journal of Neurophysiology*, 63(6): 1529–1543, June 1990.

P. J. Hancock, R. J. Baddeley, and L. S. Smith. The principal components of natural images. *Network: Computation in Neural Systems*, 3(1):61–70, 1992.

W. Hashimoto. Quadratic forms in natural images. *Network: Computation in Neural Systems*, 14(4): 765–788, 2003.

G. Hinton. Connectionist learning procedures. *Artificial Intelligence*, 40:185–234, 1989.

P. Hoyer and A. Hyvärinen. Independent component analysis applied to feature extraction from colour and stereo images. *Network: Computation in Neural Systems*, 11(3):191–210, 2000.

D. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160:106–154, 1962.

J. Hurri. *Computational Models Relating Properties of Visual Neurons to Natural Stimulus Statistics*. PhD thesis, Helsinki University of Technology, 2003. http://lib.hut.fi/Diss/2003/isbn951226823X/.

J. Hurri and A. Hyvärinen. Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural Computation*, 15(3):663–691, 2003a.

J. Hurri and A. Hyvärinen. Temporal and spatiotemporal coherence in simple-cell responses: a generative model of natural image sequences. *Network: Computation in Neural Systems*, 14(3):527–551, 2003b.

A. Hyvärinen and P. Hoyer. Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation*, 12(7):1705–1720, 2000.

A. Hyvärinen and P. Hoyer. A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Research*, 41(18):2413–2423, 2001.

A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley–Interscience, 2001.

J. P. Jones and L. A. Palmer. An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58(6):1233–1257, 1987.

C. Jutten and J. Karhunen. Advances in nonlinear blind source separation. *Proc. of the 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 245–256, 2003.

Y. Karklin and M. Lewicki. Learning higher-order structures in natural images. *Network: Computation in Neural Systems*, 14:483–499, 2003.

C. Kayser, W. Einhäuser, O. Dümmer, P. König, and K. Körding. Extracting slow subspaces from natural videos leads to complex cells. In *Artificial Neural Networks - ICANN 2001 Proceedings*, pages 1075–1080. Springer, 2001.

C. Kayser, K. P. Körding, and P. König. Learning the nonlinearity of neurons from natural visual stimuli. *Neural Computation*, 15(8):1751–1759, 2003.

T. Kohonen, S. Kaski, and H. Lappalainen. Self-organized formation of various invariant-feature filters in the adaptive-subspace SOM. *Neural Computation*, 9:1321–1344, 1997.

K. Körding, C. Kayser, W. Einhäuser, and P. König. How are complex cell properties adapted to the statistics of natural scenes? *Journal of Neurophysiology*, 91(1):206–212, 2004.

B. Lau, G. B. Stanley, and Y. Dan. Computational subunits of visual cortical neurons revealed by artificial neural networks. *Proc. Natl. Acad. Sci. USA*, 99(13):8974–8979, 2002.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

T.-W. Lee and M. Lewicki. Unsupervised image classification, segmentation, and enhancement using ICA mixture models. *IEEE Trans. Image Proc.*, 11(3):270–279, 2002.

D. MacKay. The significance of "feature sensitivity". In D. Rose and V. Dobson, editors, *Models of the visual cortex*, pages 47–53. John Wiley & Sons Ltd., 1985.

B. MacLennan. Gabor representation of spatiotemporal visual images. Technical Report CS-91-144, Computer Science Departement, University of Tennessee, 1991.

G. Mitchison. Removing time variation with the anti-Hebbian differential synapse. *Neural Computation*, 3:312–320, 1991.

L. Molgedey and G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Physical Review Letters*, 72(23):3634–3637, 1994.

K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel–based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–202, 2001.

I. Ohzawa, G. Sclar, and R. Freeman. Contrast gain control in the cat visual cortex. *Nature*, 298(5871): 266–268, 1982.

B. Olshausen. Sparse codes and spikes. In R. P. N. Rao, B. A. Olshausen, and M. S. Lewicki, editors, *Probabilistic Models of the Brain: Perception and Neural Function*. MIT Press, 2002.

B. Olshausen. Principles of image representation in visual cortex. In L. Chalupa and J. Werner, editors, *The visual neurosciences*, pages 1603–1615. MIT Press, 2003.

B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, Jun 1996.

M. Oram and D. Perrett. Modelling visual recognition from neurobiological constraints. *Neural Networks*, 7:945–972, 1994.

R. C. O'Reilly and M. H. Johnson. Object recognition and sensitive periods: a computational analysis of visual imprinting. *Neural Computation*, 6:357–389, 1994.

H. C. Peng, L. F. Sha, Q. Gan, and Y. Wei. Energy function for learning invariance in multilayer perceptron. *Electronics Letters*, 34(3), 1998.

D. Pollen and S. Ronner. Phase relationship between adjacent simple cells in the visual cortex. *Science*, 212:1409–1411, 1981.

D. L. Ringach, C. E. Bredfeldt, R. M. Shapley, and M. J. Hawken. Suppression of neural responses to nonoptimal stimuli correlates with tuning selectivity in macaque V1. *Journal of Neurophysiology*, 87: 1018–1027, 2002.

D. L. Ruderman and W. Bialek. Statistics of natural images: Scaling in the woods. *Phys Rev Lett*, 73: 814–817, 1994.

N. C. Rust, O. Schwartz, J. A. Movshon, and E. Simoncelli. Spike-triggered characterization of excitatory and suppressive stimulus dimensions in monkey V1. *Neurocomputing*, 58–60:793–799, 2004.

P. Schiller, B. Finlay, and S. Volman. Quantitative studies of single-cell properties in monkey striate cortex. I. Spatiotemporal organization of receptive fields. *J. Neurophysiol.*, 39(6):1288–1319, 1976a.

P. Schiller, B. Finlay, and S. Volman. Quantitative studies of single-cell properties in monkey striate cortex. II. Orientation specificity and ocular dominance. *J. Neurophysiol.*, 39(6):1320–1333, 1976b.

P. Schiller, B. Finlay, and S. Volman. Quantitative studies of single-cell properties in monkey striate cortex. III. Spatial frequency. *J. Neurophysiol.*, 39(6):1334–1351, 1976c.

B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

I. A. Shevelev. Second-order features extraction in the cat visual cortex: selective and invariant sensitivity of neurons to the shape and orientation of crosses and corners. *BioSystems*, 48:195–204, 1998.

A. Sillito. The contribution of inhibitory mechanisms to the receptive field properties of neurons in the striate cortex of the cat. *J. Physiol.*, 250:305–329, 1975.

P. Simard, Y. LeCun, and J. Denker. Efficient pattern recognition using a new transformation distance. In S. Hanson, J. Cowan, and L. Giles, editors, *Advances in Neural Information Processing Systems*, volume 5. Morgan Kaufmann, 1993.

P. Y. Simard, Y. LeCun, J. S. Denker, and B. Victorri. Transformation invariance in pattern recognition – tangent distance and tangent propagation. *International Journal of Imaging Systems and Technology*, 11 (3), 2000.

E. Simoncelli. Vision and the statistics of the visual environment. *Current Opinion in Neurobiology*, 13, Apr 2003.

C. Skottun, R. De Valois, D. Grosof, J. Moshnov, D. Albrecht, and A. Bonds. Classifying simple and complex cells on the basis of response modulation. *Vision Res.*, 31(7/8):1079–1086, 1991.

J. Stone and A. Bray. A learning rule for extracting spatio-temporal invariances. *Network: Computation in Neural Systems*, 6(3):429–436, 1995.

J. V. Stone. Learning perceptually salient visual parameters using spatiotemporal smoothness constraints. *Neural Computation*, 8:1463–1492, 1996.

J. V. Stone. Blind source separation using temporal predictability. *Neural Computation*, 13:1559–1574, 2001.

D. Stork and J. Levinson. Receptive fields and the optimal stimulus. *Science*, 216:204–205, 1982.

B. Szatmáry and A. Lõrincz. Independent component analysis of temporal sequences subject to constraints by lateral geniculate nucleus inputs yields all three major cell types of the primary visual cortex. *Journal of Computational Neuroscience*, 11:241–248, 2001.

F. Theunissen, S. David, N. Singh, A. Hsu, W. Vinje, and J. Gallant. Estimating spatio-temporal receptive fields of auditory and visual neurons from their responses to natural stimuli. *Network: Computation in Neural Systems*, 12:289–316, 2001.

J. Touryan, B. Lau, and Y. Dan. Isolation of relevant visual features from random stimuli for cortical complex cells. *Journal of Neuroscience*, 22(24):10811–10818, 2002.

J. van Hateren and A. van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. R. Soc. Lond. B*, 265:359–366, 1998.

M. Versavel, G. A. Orban, and L. Lagae. Responses of visual cortical neurons to curved stimuli and chevrons. *Vision Res.*, 30(2):235–248, 1990.

W. Vinje and J. Gallant. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science*, 287(5456):1273–1276, Feb 2000.

G. Walker, I. Ohzawa, and R. Freeman. Asymmetric suppression outside the classical receptive field of the visual cortex. *The Journal of Neuroscience*, 19(23):10536–10553, 1999.

G. Wallis and E. Rolls. Invariant face and object recognition in the visual system. *Progress in Neurobiology*, 51(2):167–194, 1997.

W. Walter. *Analysis 2*. Springer Verlag, 1995.

B. Willmore and D. Tolhurst. Characterizing the sparseness of neural codes. *Network: Computation in Neural Systems*, 12(3):255–270, 2001.

L. Wiskott. Learning invariance manifolds. In L. Niklasson, M. Bodén, and T. Ziemke, editors, *Proc. Intl. Conf. on Artificial Neural Networks, ICANN'98, Skövde*, Perspectives in Neural Computing, pages 555–560. Springer, 1998.

L. Wiskott. Estimating driving forces of nonstationary time series with slow feature analysis. arXiv.org e-Print archive, http://arxiv.org/abs/cond-mat/0312317/, Dec. 2003a.

L. Wiskott. Slow feature analysis: A theoretical analysis of optimal free responses. *Neural Computation*, 15(9):2147–2177, Sept. 2003b.

L. Wiskott and T. Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 14(4):715–770, 2002.

C. Zetzsche and G. Krieger. Nonlinear mechanisms and higher-order statistics in biological vision and electronic image processing: review and perspectives. *Journal of Electronic Imaging*, 10(1):56–99, 2001.

C. Zetzsche and F. Röhrbein. Nonlinear and extra-classical receptive field properties and the statistics of natural scenes. *Network: Computation in Neural Systems*, 12:331–350, 2001.

A. Ziehe and K.-R. Müller. TDSEP – an efficient algorithm for blind separation using time structure. In *8th International Conference on Artificial Neural Networks (ICANN'98)*, pages 675 – 680, Berlin, 1998. Springer Verlag.

Selbständigkeitserklärung

Hiermit erkläre ich, die vorliegende Arbeit selbständig ohne fremde Hilfe verfaßt und nur die angegebene Literatur und Hilfsmittel verwendet zu haben.

Pietro Berkes
30.3.2005