Information Routing, Correspondence Finding, and Object Recognition in the Brain

DISSERTATION
zur
Erlangung des Grades
„Doktor der Naturwissenschaften“

vorgelegt beim Fachbereich Informatik und Mathematik
der Goethe-Universität Frankfurt am Main

von

Philipp Wolfrum

aus

Heilbronn

Frankfurt (2008)

vom Fachbereich Informatik und Mathematik der
Goethe-Universität Frankfurt am Main als Dissertation angenommen.

Dekan: Prof. Dr. Klaus Johannson

1. Gutachter: Prof. Dr. Rudolf Mester

2. Gutachter: Prof. Dr. Christoph von der Malsburg

Acknowledgments

This work would not have been possible without the people I have had the honor of working with over the past years. First of all I want to thank Christoph von der Malsburg, whose scientific concepts have inspired large parts of this thesis, and who granted me enough freedom to follow my own research directions while at the same time providing motivation whenever this was necessary. I also thank Rudolf Mester for good discussions that helped me retain my engineering viewpoint on the interdisciplinary problems that I worked on. Discussions with Geoff Goodhill, Bruno Olshausen, Jochen Triesch, and Alan Yuille, among others, have shaped my thinking and had an important impact on this thesis.

I thank Urs Bergmann, Jenia Jitsev, and Junmei Zhu for proof-reading parts of the thesis, and Jörg Lücke for good collaboration. The neurogroups at FIAS have provided a rich and stimulating environment, it has been fun working and thinking with you! I also thank all colleagues at FIAS for good company and the truly interdisciplinary interaction we had (e.g. at Hirschegg). Last but not least I owe this thesis to my parents, who planted curiosity and a desire for understanding in me that were stronger than the difficulties I met during this dissertation.

Contents

List of Figures vii

List of Tables ix

1 Introduction 1
1.1 Object Recognition 1
1.1.1 Strategies for Achieving Invariance 2
1.1.2 State of the Art in Object Recognition 3
1.2 Plausibility of the Two Concepts 5
1.2.1 Computational Arguments 5
1.2.2 Experimental Evidence 8
1.3 Proposal for Dynamic Routing 9

2 Neural Model for Face Recognition 13
2.1 Correspondence Finding 13
2.2 Face Recognition 14
2.3 The Basic Computational Units: Cortical Columns 16
2.3.1 Neurobiological Background 16
2.3.2 A Model of the Cortical Column 18
2.4 The Network 21
2.4.1 Input Layer 23
2.4.2 Assembly Layer 26
2.4.3 Gallery Layer 29
2.5 Results 29
2.5.1 General Network Behavior 31
2.5.2 Position Invariance 34
2.5.3 Tests on Standard Databases 36
2.5.4 Attention Experiments 39
2.6 Discussion 40

3 Switchyards—Routing Structures in the Brain 47
3.1 Multi-Stage Routing 47
3.2 Physiological Background of Dynamic Routing 48
3.3 Optimized Architectures for Routing 49
3.3.1 Routing Between Two Regions of the Same Size 50
3.3.2 Routing Circuit with Different Sizes of Input and Output Layer 56

3.4 Interpretation of Results 58
3.4.1 Difference to Sorting Networks 58
3.4.2 Physiological Interpretation 59
3.5 Discussion 63

4 Ontogenesis of Switchyards 67
4.1 Ontogenetic Plasticity Mechanisms in the Brain 68
4.2 A Model for the Growth of Routing Networks 69
4.3 Results 73
4.3.1 Noise Robustness 75
4.3.2 Growth of Three-Dimensional Networks 78
4.4 Other Potential Mechanisms 81
4.5 Conclusion 84

5 Recognition with Switchyards 85
5.1 Matching of Two Patterns 86
5.2 Recognition from a Gallery of Patterns 93
5.3 Conclusion 97

6 Discussion and Outlook 101

Appendix 105
A Self-Normalization Properties of Columnar Dynamics 105
B Gabor Transform 106

Bibliography 109

Zusammenfassung in deutscher Sprache 123
1 Einleitung 123
2 Neuronales Modell zur Gesichtserkennung 124
3 Switchyards 126
4 Ontogenese von Switchyards 127
5 Mustererkennung mit Switchyards 128
6 Diskussion 129

Lebenslauf 131

List of Figures

1.1 Challenges for a vision system 6

2.1 The correspondence problem 15
2.2 Columnar organization of cortex 17
2.3 Timecourse of unit activities 20
2.4 Principal layout of the face recognition system 22
2.5 Faces represented by a grid or a face graph 23
2.6 Architecture of the network 24
2.7 Information flow in the network 25
2.8 Average face graph 26
2.9 Interaction among control units 30
2.10 Matching process between Input and Input Assembly 32
2.11 Recognition process 33
2.12 Position invariance 35
2.13 A sample of 30 faces from the FERET database 36
2.14 Cumulative match scores for the FERET database 37
2.15 Cumulative match scores for the AR database 38
2.16 Spatial attention experiments 41
2.17 Object search experiments 42
2.18 Activity of the Gallery Assembly after priming of female faces 43
2.19 Result of a priming experiment 44

3.1 One- and two-dimensional routing networks 51
3.2 Number of required units as a function of intermediate layers 53
3.3 Prefactors c and $\tilde{c}$ 55
3.4 Possible forms of tapered networks 56
3.5 Routing network with linear decrease of layer size 57
3.6 Number of possible conflicts as a function of distance of input nodes 60
3.7 Dependence of network size on parameter α 62

4.1 Axonal growth cone 68
4.2 Switchyard architecture 70
4.3 Term G for alignment of coordinate systems 72
4.4 Role of the marker similarity term 73
4.5 Snapshots of the growth process 74
4.6 Results for layer size $n = 27$ 76

4.7 Results for layer size 125 and noisy initial conditions 77
4.8 Noise robustness of the ontogenetic mechanism 79
4.9 Growth of three-dimensional networks 80
4.10 Results with and without wraparound boundary conditions 82

5.1 Challenge of matching patterns via a switchyard 86
5.2 Information flow in the dynamic switchyard 87
5.3 Typical pattern used for matching experiments 88
5.4 Principle of matching in a switchyard 89
5.5 Snapshots of the matching process 90
5.6 Information flow in the full recognition system 93
5.7 Patterns stored in the gallery 94
5.8 Matching and recognition process in the full system 95
5.9 Signals propagated down the Gallery Assembly stream 96
5.10 Random dot stereogram 98

6.1 Extension to recognizing several categories 103

List of Tables

2.1 Recognition rates 39

4.1 Chemical mechanisms for axon guidance. 69

1 Introduction

A central task of the brain is finding homomorphies or correspondences between patterns. When we look at a scene, for example, our visual system has to find correspondence between the pattern that falls onto the retina and memories stored in the brain, enabling us to make sense of our environment. This ought to work even when the instance we get to see of an object in a certain situation differs significantly from its representation in memory that was created under very different conditions. Thus, the process of visual correspondence finding must not compare the two patterns in a direct way, but the comparison should be invariant to differences that do not change the *meaning* of the patterns.

Finding correspondences invariantly of transformations is not only important in vision. It is also required for sensory tasks like perceiving speech and music—you want to recognize words independently of the pitch of a voice, and a melody regardless of the musical instrument it is played on—up to high level tasks like understanding metaphors (extremely challenging invariances) and abstract reasoning.

When *acting* instead of perceiving, the brain has to invert this process. It may start out, e.g., with the plan of grasping an object in front of us, and then has to translate this high-level plan into the corresponding, very specific motor commands to the different arm and hand muscles. These motor patterns will turn out to be quite different when the object we want to grasp has a different weight or surface structure, and they will look different again if we want to use a tool instead of the hand alone.

This thesis tries to address some of the questions arising in the context of correspondence finding in the brain. While doing so, we will mostly focus on visual information processing as an application of correspondence finding processes. We therefore devote the following section to a survey of the state of the art in object recognition.

1.1 Object Recognition

When we look at the object in front of us, a specific pattern of activity is created in the ganglion cells of the retina. This pattern is relayed and transformed on the way via the thalamus and primary visual areas to higher cortical stages, where it may interact with and activate certain memories stored there. If this happens, we feel that we have *recognized* the object. If we see the same object under slightly changed circumstances, e.g. at a different position, our brain will recognize it as the same object as before without any problems, so smoothly in fact that in the early days of computer vision this was not even noticed as a serious problem. Yet the retinal pattern created by this new situation is entirely different. Its (mathematical) similarity with the original pattern may even be smaller than that of two patterns caused by different objects, but in the same position (cf. Duda et al. 2001, p. 189).

So how does our brain (and how can any computational system) solve this problem of recognizing the two images as being caused by the same object? The sheer amount of different situations in which we can recognize the same object makes it prohibitive to store all possible views in memory. If this is not possible, however, then our brain must have active mechanisms that recognize different patterns as coming from the same object. This is the problem of *invariant recognition*. By now, it has a history of more than 50 years of active research in such diverse disciplines as computer science and engineering, physics, neuroscience, and psychology, and it remains far from being solved. Over this period, a multitude of neural models has been proposed to explain invariant recognition. Although they all have their unique characteristics, they can roughly be cast into two different categories according to the underlying principles they follow.

1.1.1 Strategies for Achieving Invariance

Pooling or feature-based approaches

A traditional approach in computational neuroscience, which we refer to here as the *feature-based* approach, aims to achieve invariance by converging the signals from feature detectors at different positions (and scale and orientation) in an earlier layer into a single "complex" unit at a higher layer. This idea was first proposed by Frank Rosenblatt in his four-layer perceptron (Rosenblatt 1961), and a multitude of similar models has followed since (Fukushima et al. 1983, LeCun et al. 1989, Mel 1997, Riesenhuber and Poggio 1999, Deco and Rolls 2004). This convergence (also called *pooling*) of features at different positions, scales, etc., makes the response of the following complex unit invariant to those transformations. For example, a unit that pools over feature detectors at different positions will have a position invariant response. Feature hierarchies usually consist of several stages that on the one hand combine simple features into more and more complicated ones while at the same time pooling over increasingly large regions in transformational space to become more and more invariant. Both operations have to go hand in hand because in this approach there is an inherent trade-off between complexity of features and resolution at which they can be represented (cf. Serre et al. 2007).

Correspondence-based approaches

The *correspondence-based* approach does not recognize objects by the activity of a single or a few cardinal cells, but through a pattern matching process that establishes explicit correspondences between points in the input image and parts of the object model stored in memory. The idea that this might happen via synchronization of patterns in the brain was proposed by von der Malsburg (1981) and developed successively into a neural model of correspondence finding (Wiskott and von der Malsburg 1996). The idea of synchrony as a binding mechanism in the brain continues to be studied both experimentally (e.g. Gray and Singer 1989, Singer 2003) and theoretically (e.g. Wang 2005). Olshausen et al. (1993) introduced the notion of dedicated *control units* to control the flow of information between patterns.

The principle of correspondence finding between patterns requires direct links or routing networks providing connections between the two patterns. Instead of pooling over (i.e. basically responding to any activity within) lower stages, the correspondence-based approach actively selects the information that is allowed to activate the next layer. This is done by dynamically switching connections between successive layers, thus routing input information to different parts of the next layer depending on the situation. Such a routing process can in principle compensate the effects of variances, rendering the information represented at the output stage invariant of the extrinsic properties of the input image like position, scale, pose, etc., without discarding information. Dynamic information routing was proposed as a principle of invariant perception already by Pitts and McCulloch (1947), and the term *dynamic link* was introduced by Feldman (1982). Several specific routing architectures have been proposed since (Anderson and Van Essen 1987, Postma et al. 1997, Arathorn 2002). Although the ideas of active information routing and correspondence finding have been studied mostly independently in the past, we will argue in this thesis that they mutually require each other in a full vision system.

1.1.2 State of the Art in Object Recognition

After this definition of the principal approaches in computational neuroscience, let us now review the performance of current systems for object recognition (both neural models and computer vision systems) in the light of these distinctions. Feature-based and correspondence-based models have been successful in different application domains, as discussed in (Biederman and Kalocsai 1997). Feature-based approaches are very successful in classification tasks. The "standard model" from the Poggio lab (Riesenhuber and Poggio 1999, Serre et al. 2007) is a refinement of the Neocognitron (Fukushima et al. 1983). It uses two successive stages of pooling and feature extraction and a support vector machine as a final classifier. Pooling operations (over different positions and scales) are hard-coded, while features are learned via weightsharing either simply as patches collected from training data or with a radial basis function approach. The model is competitive with state-of-the-art computer vision approaches (see below) in classifying objects like cars or airplanes, and shows some success in labeling parts of scenes as 'sky', 'building', etc. Other models in this direction include "LeNet" (LeCun et al. 1989, 2004) and the model of Deco and Rolls (2004). Wersing and Körner (2003) learn sparse features, and their highest nodes do not pool over the whole image, which reduces computational costs and allows distinguishing between more objects than the few object classes the above approaches can handle. Feature-based systems are suited for classification tasks because here their relative insensitivity to small metric variations of object parts is advantageous. Also, the static connectivity in feature-based systems allows them to be tuned to specific image databases. However, see (Pinto et al. 2008) for a discussion why such standard databases may not be a very good benchmark.

Correspondence-based systems, on the other hand, prevail in recognition tasks in which small differences in features and their arrangements are important. A typical example for this is face recognition. Since the very successful Elastic Graph Matching (EGM) model (Wiskott et al. 1997), the best performing systems for face recognition have mostly been correspondence-based (Phillips et al. 2003, see also www.frvt.org).

Let us now review how pure computer vision systems, whose only goal is to achieve the highest performance possible for certain tasks without following a specific "philosophy", can be classified according to the two above distinctions. The most obvious variance that a visual system has to deal with is translation of an object. While the most important biological mechanism for dealing with translations is eye movements (saccades), the origin of these saccades requires explanation in the first place. And even without eye movements we are capable—with some limitations under unnatural conditions (Cox et al. 2005)—to recognize images that have been shifted on the retina (Bar and Biederman 1999, Fiser and Biederman 2001). When the (two-dimensional) Fourier transform is applied to an image, the resulting amplitude spectrum represents the global frequency content of the image and is therefore invariant to translations. This means that two images of the same object at different positions have the same Fourier spectrum. Pollen et al. (1971) suggested that this mechanism might be used in the visual system to achieve translation invariance, and it has been used to some extent in artificial vision systems. Unfortunately, the amplitude spectrum of the Fourier transform is not unique: since phase information is not retained, for any given image there are totally different, nonsensical images that have the same amplitude spectrum.

A generalization of the Fourier amplitude spectrum is the use of histograms that represent the number of certain features in an image without paying attention to spatial relations. Histogram approaches have a long history in computer vision (Schiele and Crowley 2000, Swain and Ballard 1991). The currently most popular approach in this direction are *bag-of-features* models, a name derived from similar *bag-of-words* approaches in document analysis (Joachims 1998). Bag-of-features models (e.g., Leung and Malik 2001, Lazebnik et al. 2003, Csurka et al. 2004) represent images as an unstructured set of image patches or other features. Since they do not model any spatial relations, only statistics of an image, they have been especially successful in scene classification (i.e. catching the gist of an image, like whether it shows an office environment, a street scene, or mountains). Examples of work in this direction include (Oliva and Torralba 2006, Torralba et al. 2003, Lazebnik et al. 2006). Pure bag-of-features models are related to the feature-based approach in its crudest form. Pyramid approaches with bags of features (Lazebnik et al. 2006) introduce a bit of spatial ordering to the features and correspond to multilayer feature hierarchies like (Fukushima et al. 1983) and subsequent models.

Although the simplicity of use and robustness to occlusions of bag-of-features models make them very popular in computer vision, tasks like object localization or accurate recognition usually require modeling of the geometric relations between object parts. One step in this direction is for example adding pairwise relations between neighboring features to the bag-of-features representation (Sivic et al. 2005). This approach is similar to the way neighborhood relations are encouraged in Elastic Graph Matching (Wiskott et al. 1997), although a full object model is still missing. Approaches sporting a full geometric object model include the generative models of Perona's group (Fei-Fei et al. 2003, Fergus et al. 2003, Song et al. 2003) or the geometric correspondence search of (Berg et al. 2005). In (Song et al. 2003), for example, human body shapes are represented by a mixture of decomposable triangulated graphs. Such a body model can be learned from unlabeled or labeled data and can then be used to detect moving humans in image sequences. Representing objects by flexible graph structures is exactly the approach taken by correspondence-based systems for face recognition as in (Wiskott et al. 1997).

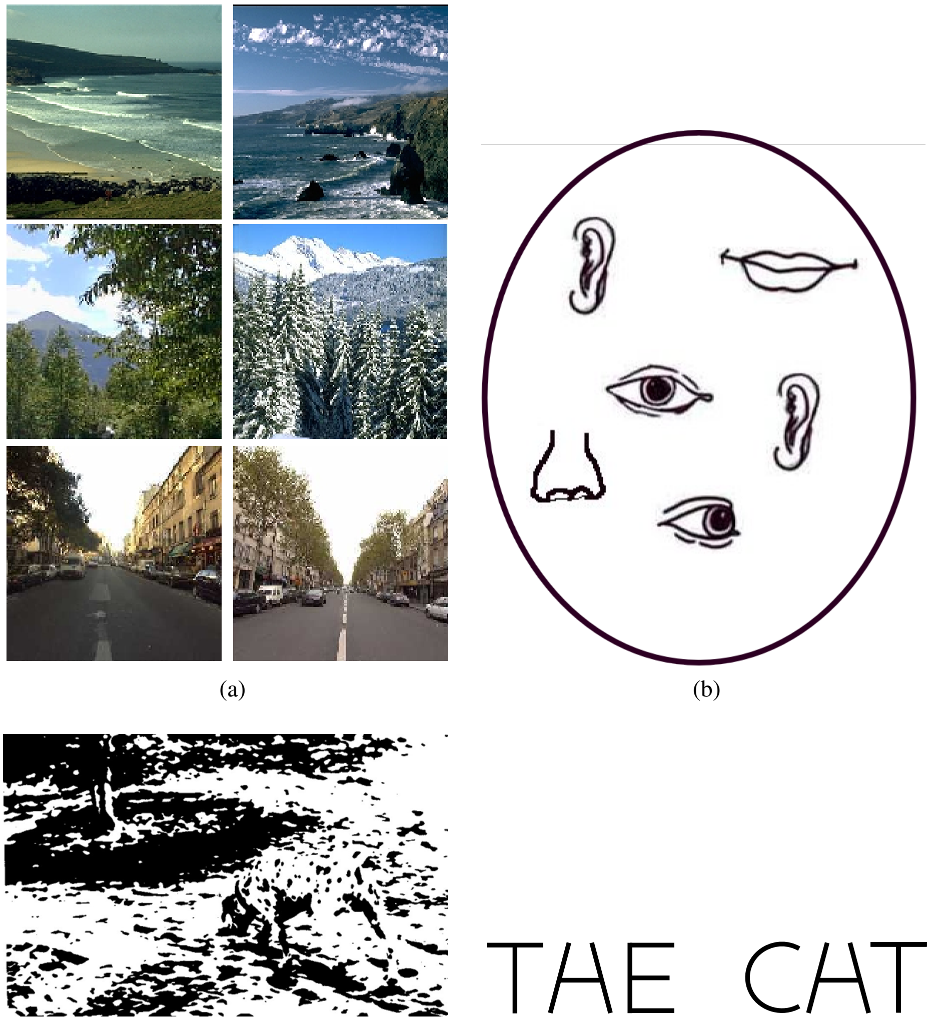
Another question besides spatial representation of objects is how the choice of features helps achieve invariance. In this regard, feature-based and correspondence-based approaches in neural modeling differ. In the former approach, a feature hierarchy produces invariant features that can be used for classification. Since information about the original *variance* is discarded on the way, the recognition process cannot ensure any more that the features extracted from different parts of the image are actually consistent (see Section 1.2 for a further discussion). In the correspondence-based approach, it is the matching process that makes recognition invariant and simultaneously transforms non-invariant into normalized features. Since this matching process is global, it automatically ensures that invariances are globally consistent across the whole object. While it would go beyond the scope of this introduction to review the many kinds of feature types that are used in computer vision, let us focus on one specific type that is interesting with respect to the above distinction. The SIFT (*scale invariant feature transform*) extracts features that are scale and orientation invariant *without* discarding information about these variances. Keypoints are chosen by detecting extrema in scale-space (i.e. over position and different scales), and at those keypoints a local orientation is calculated on the basis of the local image gradient direction. Thus, every keypoint gets assigned a specific scale and orientation which is used subsequently to normalize local image information, yielding invariant local features. At the same time, the information about keypoint position, scale, and orientation can be used to ensure that the object recognition process uses only features which are mutually consistent in terms of their variances.

With rising computing power, probabilistic approaches to computer vision have received increasing attention in recent years. Factor graphs (Kschischang et al. 2001) can be used to make very fast inferences about visual scenes, while generative models (e.g., Murray and Kreutz-Delgado 2007) represent objects in explicit models including possible variances, enabling them to *generate* images of specific instances of an object. This is very similar to the way objects are represented by correspondence-based models, at least newer ones like the system developed in Chapter 2 of this thesis. Much effort in probabilistic modeling goes into how inference on them is carried out. Since many classic inference techniques are prohibitive owing to the sheer size of vision problems, correspondence-based neural models might actually provide inspiration here. For overviews of probabilistic approaches see (Yuille and Kersten 2006, Chater et al. 2006).

1.2 Computational and Biological Plausibility of the Two Concepts

1.2.1 Computational Arguments

So what are the computational differences between feature-based and correspondence-based approaches to vision, and what consequences do they have in terms of performance of the resulting models? As we have seen above, a main difference between the concepts is whether they explicitly represent the spatial layout of objects. Feature-based approaches, which more or less neglect this information, consider an object as recognized when all its constituing features are present somewhere in the scene. This approach is perfectly fine for problems where the spatial



(a) (b)

(c) (d)

Figure 1.1: These images illustrate situations that a vision system may encounter. **(a)** Landscape scenes. Without using spatial information, state-of-the-art feature-based approaches classify these coast and forest images as "mountain", and the street images as "highway". Images reprinted with kind permission of Anna Bosch. **(b)** An image consisting of the scrambled parts of a face (inspired by a similar image in Olshausen 1994). **(c)** A Dalmatian. If you have ever seen this image, you will recognize it immediately. If you have not seen it before: the dog is in the right half of the image, walking towards the left, its muzzle to the ground. **(d)** What is written here?

arrangement of parts is more or less irrelevant, like catching the gist of simple scenes (those in Figure 1.1a are already too challenging, see below). In these cases, feature-based approaches are actually superior since their simplicity and their static connectivity make them very easy to train and optimize, while finding useful geometric scene models as correspondence-based systems would use them might turn out difficult. Nevertheless, even for recognizing scenes, some spatial information may be helpful. Bosch et al. (2008) show that scenes which are misclassified by pure feature-based approaches (see Figure 1.1a) can be classified correctly with a system that combines discriminative (i.e. feature-based) approaches with explicit spatial models.

For real object recognition, however, this lack of spatial ordering is a serious disadvantage, since it makes a system susceptible to falsely recognizing as an object an image that contains the parts of this object, but in a completely scrambled setup (e.g. the scrambled face in Figure 1.1b). This problem is especially likely to occur in scenes with complex background, where the system might pick features present in the background to "hallucinate" an object. Newer models have solved this problem partially in two ways. Interleaving many of the feature extraction and pooling stages and limiting the range of pooling at any single stage can reduce the insensitivity to spatial arrangement of features to some extent (Serre et al. 2007). The other approach to alleviating the problem is using overcomplete dictionaries of features that are dedicated to specific object classes (Mel and Fiser 2000). The hope is that this will provide additional features that are sensitive to the spatial constellation of parts and can thus deal with scrambled images or background effects.

Nevertheless, there are visual tasks that require *exact* spatial information. An impressive one for example is our ability to recognize three-dimensional shapes in random dot stereograms. This requires that exact geometric correspondences be found, this time not between an image and internal memory, but between different regions of the input image. The single local features, random dots, are of no great value in finding these correspondences, rather a matching process between large constellations of points is required. Consequently, models addressing this task (e.g. Marr and Poggio 1976) are correspondence-based. Most probably, any kind of stereopsis skill will require correspondence finding mechanisms.

A related question is the role that local and global decisions play in recognition. When looking at the famous Dalmatian in Figure 1.1c, for example, local features are totally useless, and only a global, model-driven recognition process can make sense of the image. In Figure 1.1d, on the other hand, local features are useful, but ambiguous. This ambiguity of the central letters in the two words can only be resolved by contextual feedback from the global decision. The classical feature-based paradigm does not support the notion of local feature detectors incorporating cues from global decisions or from the decisions of their neighbors, while feedback and local interactions are fundamental principles of correspondence-based approaches.

Leaving the question of spatial representation and global interaction aside, one problem of feature hierarchies remains: by pooling over variances, they do not only become invariant to them, but they effectively discard information about these variances! In consequence, there is no way of ensuring that features in the image assumed to represent a certain object are actually mutually consistent in terms of their variances. Again, overcomplete coding (Mel and Fiser 2000) may solve some of these problems by introducing overlapping features. But even if a system of this kind is able to detect and recognize objects, it has no way of telling where the object is, what size it has, whether the person just recognized has a happy or a sad expression on her face, etc.

Moreover, because the pooling operation is not invertible, feature-based systems cannot *generate* specific instances from high-level representations. Their object models are mere *detectors* instead of explicit models representing objects in all their possible variances. It has been argued (cf. the "predictive coding" of Rao and Ballard 1999) that such capability to regenerate the current percept and compare it with the actual stimulus may be advantageous for a vision system, because it increases the signal-noise-ratio and allows a global consistency check of the features (compare the above discussion). And as mentioned above, the recent success of Bayesian models lies in their having explicit models of object appearances (the *likelihood* in Bayes' rule). It is explicit, generative models that enable advanced visual functions like reasoning about a percept, mental filling in of occluded regions, or testing hypotheses about it, in short that give us the feeling of being in direct contact with our visual environment. Correspondence-based systems do not automatically have explicit object models, but we will argue in this thesis how they can be implemented.

1.2.2 Experimental Evidence

Let us now review physiological, anatomical, and psychophysical evidence that argues for and against the two approaches. For this, it is interesting to look at what is known about feature processing and receptive fields (RFs) in the visual system. It is often argued that the primate ventral stream constitutes a kind of hierarchy of more and more complex features (Tanaka 1996, Oram and Perret 1994), from Gabor-like RFs in V1 to neurons in inferotemporal cortex (IT) that react invariantly to large parts of objects. This is exactly what feature-based systems like (Riesenhuber and Poggio 1999) try to model. Correspondence-based systems, on the other hand, have relied so far on representing objects by groups of rather basic features, which appears less realistic. Note, however, that while V1 cells respond similarly in awake and anesthetized animals, it is nearly impossible to drive IT cells under anesthesia (for recent results on the large differences between general neuron responses in awake and anesthetized animals, see e.g. Greenberg et al. 2008). So the very complex effective RFs of IT cells in awake animals cannot directly correspond to anatomical RFs, since then they should respond similarly also under anesthesia. Instead, they might arise from interaction of many cells with simpler RFs. This is the way complex percepts like whole faces are represented in the model of Chapter 2. Nevertheless, correspondence-based models should try to address feature extraction more explicitly than in the past, especially since there are no fundamental obstacles for doing so.

The notion of effective RFs leads to the general question of how static or flexible RFs are. There is abundant physiological evidence that they are not static at all. Shifting receptive fields have been found in lateral intraparietal cortex (Duhamel et al. 1992, Kusunoki and Goldberg 2003), in MT (Womelsdorf et al. 2006), and even in V2 and V4 (Luck et al. 1997). Therefore it would be possible that effective receptive fields in the visual system change from one instance to the next to route and match the current stimulus of interest to representations in memory.

A main argument for feature-based feedforward recognition has been the processing speed of the human visual cortex. Thorpe and coworkers have shown (Thorpe 1988, Thorpe et al. 1996) that humans can decide whether an image contains an animal or not in less than 150ms. In the area of face recognition, Debruille et al. (1998) found that event-related potentials (ERPs) in response to novel vs. known faces start to differ as early as 76 to 130ms. Since such times are not much longer than the time required for a first wave of spikes to travel through the ventral stream after presentation of an image, it has been argued that visual recognition must be feedforward. However, such an interpretation seems to capture only part of the story. For instance, population codes can increase the speed of information transmission. The average spike rate of large excitatorily coupled neuron populations can be read out on a timescale that is much faster than the average spike latency of their single constituing neurons (van Vreeswijk and Sompolinsky 1998). Thus, networks that have such “high gain” connectivity can respond very sensitively to subtle and fast input changes (similar to the principle of criticality; Bak, 1996). Furthermore, independently of population coding, correspondence-based systems can react very fast if they are primed, i.e. if their dynamic connections have already been prepared for a specific stimulus. This might be the case in simple classification tasks like in the experiments of Thorpe (1988). In such cases, even correspondence-based systems react in a feedforward way.

Another point that begs discussion is the important role of priming or congruency effects in general object perception (for a review, see Graf 2006). For example, when we look at the moon rising over a distant skyline, it looks much larger than when we see it high in the sky. This is because the size of the buildings around the moon primes our visual system for a certain scaling which is then unconsciously applied to the whole scene. Psychophysical experiments have shown that priming certain scales (Bundesen and Larsen 1975) or orientations (Jolicoeur 1985, Lawson and Jolicoeur 1999) changes our performance and reaction times in object recognition. From these findings we can conclude that it does take effort and time to align the external world with internal representations, suggesting active dynamic processes for correspondence finding rather than passive pooling operations. In Section 2.5.4 we show how seamlessly priming effects can be incorporated into correspondence-based models of object recognition.

Combining the evidence for feedforward processes on the one hand and correspondence-based ones on the other, it appears likely that the brain employs both strategies. This could be in the form that first there is a fast and unconscious feedforward sweep that is followed by more in-depth recurrent processing, only the latter leading to conscious perception (Lamme 2003). This is consistent with findings by Johnson and Olshausen (2003), who report two ERP signals related to object recognition, an early presentation-locked one, and a later signal that correlates in timing with the response times for recognition.

1.3 Proposal for Dynamic Routing as Principle of Brain Function

From the preceding discussion of computational, biological, and behavioral aspects we can conclude that the human visual system and most probably any other powerful object recognition system has to combine both feature-based and correspondence-based mechanisms. Wherever possible, the brain will employ feedforward mechanisms, since they are fast and undemanding. This may suffice in highly trained situations where immediate and stereotypical responses are required. Correspondence-based mechanisms, on the other hand, become necessary in ambiguous or novel situations, as well as in visual tasks beyond pure classification, like filling in of missing information or reasoning about a percept. As argued at the beginning of this introduction, we are convinced that dynamic information routing and the finding of correspondences also play a crucial role in other areas of brain function like auditory perception, understanding and producing syntactically correct speech, action planning, and producing appropriate motor outputs.

The necessity of correspondence-based mechanisms has been neglected in large parts of the neural modeling literature. In computer vision, on the other hand, it is well accepted, sometimes only showing up as an underlying principle, sometimes explicitly like in generative models and Bayesian approaches. Here, in turn, there have been few efforts to explain how the correspondence finding process could be implemented in a dynamic distributed system like the brain. This neurally plausible realization of correspondence finding processes will be at the focus of this thesis.

Even beyond vision, the general task of mapping corresponding patterns to each other (and, on the other hand, recognizing when two patterns do not match) is central to our survival and our intelligence. Although we seem to solve it without difficulties, it poses certain requirements to the brain as a physical and computational system.

1. Signal pathways must exist in the brain that allow routing of information between different parts of the brain, such that the patterns residing there can be compared. These pathways have to be manifold enough to allow the many types of routing and transformations we hinted at above; on the other hand they should be as parsimonious as possible for evolutionary reasons.

2. The brain must have computational mechanisms that implement the actual correspondence finding using these pathways. Realistically, this process has to be structured into several stages, to achieve high flexibility without drowning in a combinatorial explosion. These stages have to cooperate since the task of one alone can usually not be carried out without solving the full problem.

These requirements give rise to a multitude of questions: How can correspondence finding be implemented in the brain? What kinds of information routing pathways are advantageous? How can such structures self-organize during ontogenesis? And how can all this be integrated into a functional object recognition system?

Outline of Thesis

This dissertation tries to contribute to some of those open questions. In Chapter 2, we develop a neurally plausible model for object recognition. In that chapter, we make very simple assumptions on the underlying routing structures and rather concentrate on the correspondence finding mechanisms. Chapter 3 argues for the need of multistage routing structures and introduces an architecture that is optimal in terms of required neural resources. In Chapter 4 we derive a mechanism that could explain the ontogenesis of such structures. Finally, Chapter 5 outlines how these ideas could be combined into an integrated system performing routing over several stages *and* object recognition.

2 A Correspondence-Based Neural Model for Face Recognition

In this chapter we develop a correspondence-based model for object recognition. We will focus here on the question how correspondence finding can be realized neurally, using very simple assumptions for the underlying routing structures (a more realistic treatment of these will be given in Chapter 3).

The general underlying principle of correspondence finding is similar to that developed in (Bienenstock and von der Malsburg 1987). We introduce this principle in Section 2.1. The function of the system as a whole is similar to Elastic Graph Matching (EGM) systems (Lades et al. 1993, Wiskott et al. 1997). However, while EGM is an algorithmic system that explicitly minimizes energy functions to arrive at the final solution, the model proposed here is a biologically plausible network of cortical columns. And while EGM uses two separate, subsequent steps for object localization and identification, the present system integrates these steps into one coherent dynamic network, so that the outcome of both the localization and the recognition process is the final state of one large dynamic system.

The system was developed mostly with the application to face recognition in mind, a topic that we introduce and motivate in Section 2.2. The basic computational unit of the network is a model of the cortical column. This model was inspired by Jörg Lücke's work on modeling cortical columns (Lücke 2005, Lücke and von der Malsburg 2004), but it is functionally different to enable multi-layered networks of columns with continuous internal information transmission that are capable of object recognition. The column model is introduced in Section 2.3. Having introduced the background, we proceed to outline the full network in Section 2.4. We report the behavior of the network and test it for different tasks in Section 2.5, before concluding the chapter in Section 2.6. The contents of this chapter were partially published in (Wolfrum, Lücke and von der Malsburg 2008, Wolfrum, Wolff, Lücke and von der Malsburg 2008), the material presented in Section 2.5.4 in (Wolfrum and von der Malsburg 2008).

2.1 Correspondence Finding

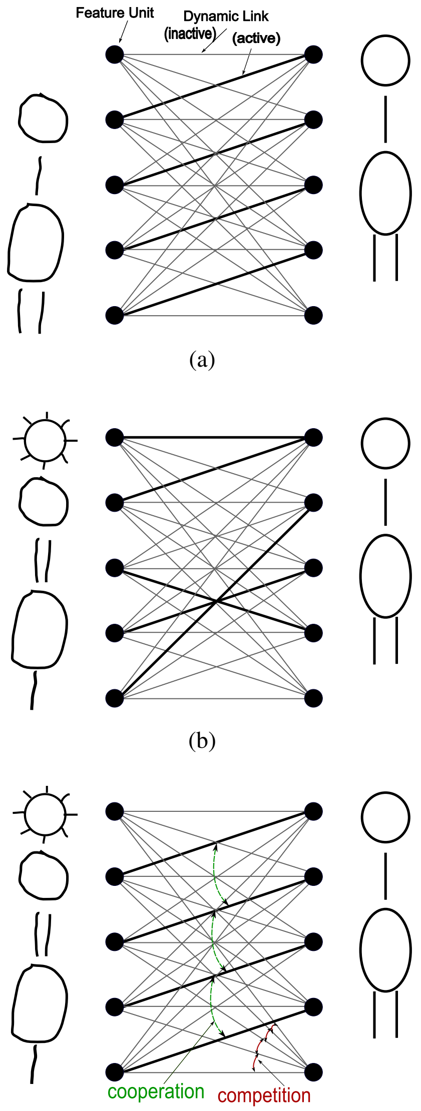
How do correspondence-based systems find correspondences between images in a distributed, not centrally organized way? The basic problem is illustrated in Figure 2.1a, which shows two stick-figures as input and model. Both of these objects are represented by a layer of *feature units* (black circles). The general correspondence problem is to identify points in the input image and their corresponding counterparts in the model image, making it possible to map one image onto the other. When those images are represented neurally, it means that corresponding units have to be identified and their relationship has to be marked somehow. We do this by activating *links* between the units. In Figure 2.1a, black lines represent active links (i.e. correct correspondences) as a subset of potential connections (gray lines).

As a prerequisite for correspondence finding, feature similarities must be computed. In the case of Figure 2.1a, simply activating links between those units with the highest similarity solves the correspondence problem. Unfortunately, in realistic scenarios high feature similarity is not sufficient to find correct correspondences. Different images of the same object may vary greatly, leading to high similarity between non-corresponding points (see, e.g., Wiskott 1999). Unrelated features in the background of an image may add to this confusion. Figure 2.1b shows this in cartoon form, heavy lines connecting the features with highest similarity. The interference of the background structure above the head of the stick figure and the changed appearance of neck and legs results in wrong or ambiguous correspondences in this case. For realistic inputs such situations are very frequent and the ambiguities increase the more kinds of feature detectors are used. For a human observer, in distinction, it is easy to find correct correspondences, also in Figure 2.1b. The reason for this is that an object is defined by its features *and* their spatial arrangement. Correspondence-based systems therefore also have to take both of these cues into account. We do this here by allowing *topologic interaction* between links (see Figure 2.1c). Links representing mutually consistent correspondences (parallel links in this simple case) strengthen each other, while mutually exclusive links (links emanating from the same node) inhibit each other. With the right balance between topologic interaction and feature similarity influence, this method will find the right global correspondences in spite of local feature discrepancies. This approach was first taken in dynamic link matching systems (Lades et al. 1993, Wiskott and von der Malsburg 1996, Würtz 1995). Here, we take the same principal approach, but use explicit units that control the connectivity between layers similarly to control units in shifter circuits (Olshausen et al. 1993). In (Lücke et al. 2008) a system is described that finds correspondences between two patterns using this approach.

2.2 Face Recognition

The object recognition system presented in this chapter was developed with a focus on and mainly applied to face recognition. Faces have a well-defined spatial layout, which allows them to be treated holistically (Biederman and Kalocsai 1997), obviating the need to address hierarchical composition out of sub-objects (Biederman 1987). On the other hand, the similarity of different faces in general appearance requires very fine discrimination concerning both the exact location of landmark points and textural differences. These two constraints (compact shape and sensitivity to details) make face recognition specifically suited for correspondence-based approaches.

Face recognition is interesting because it is an important capability of the human mind, the ability to perceive and interpret faces being central to human social interactions. Consequently, there exist dedicated neural resources for face recognition in the brain. While the fusiform face area (FFA) is specialized in face *recognition* (Kanwisher and Yovel 2006, Tsao et al. 2006), neurons in medial frontal cortex seem to be involved in face *detection* (Summerfield et al. 2006), and the amygdala is responsible for evaluating *emotional* cues, mainly fear.



(a)

(b)

(c)

Figure 2.1: The visual correspondence problem is the task of linking corresponding points between two images. **(a)** Input and model images are represented by arrays of feature nodes (black circles). All potential correspondences are symbolized by lines between the feature nodes. High feature similarities are indicated as heavy lines. In this case they represent the correct correspondence. In **(b)**, evaluation of feature similarity alone leads to wrong correspondences. **(c)** This problem is solved by additional interaction between dynamic links, which help finding the correct global correspondence. Competition suppresses multiple matches to a single node, while cooperation encourages globally consistent mappings.

Face recognition has a well-established history in computer vision. Differently to many other object recognition areas, stiff competitive tests are carried out on widely available image galleries (e.g., Phillips et al. 2000, Messer et al. 2004, Phillips et al. 2005). The existence of such tests and databases allows objective judgment of the performance of single systems. When photos are taken under controlled conditions, the performance of technical systems can be as good as that of humans or it even exceed it (Adler and Schuckers 2007). In difficult situations, however, humans still outperform machine vision systems. Thus, face recognition is a very relevant and mature field with much experimental background available. This nourishes the hope that developing a model of face recognition that on the one hand is guided by many of the relevant neurobiological facts and on the other hand comes close to the functional performance of state-of-the-art technical systems can help gaining genuine insight into the operating principles of the brain. This is what we set out to do in the following sections.

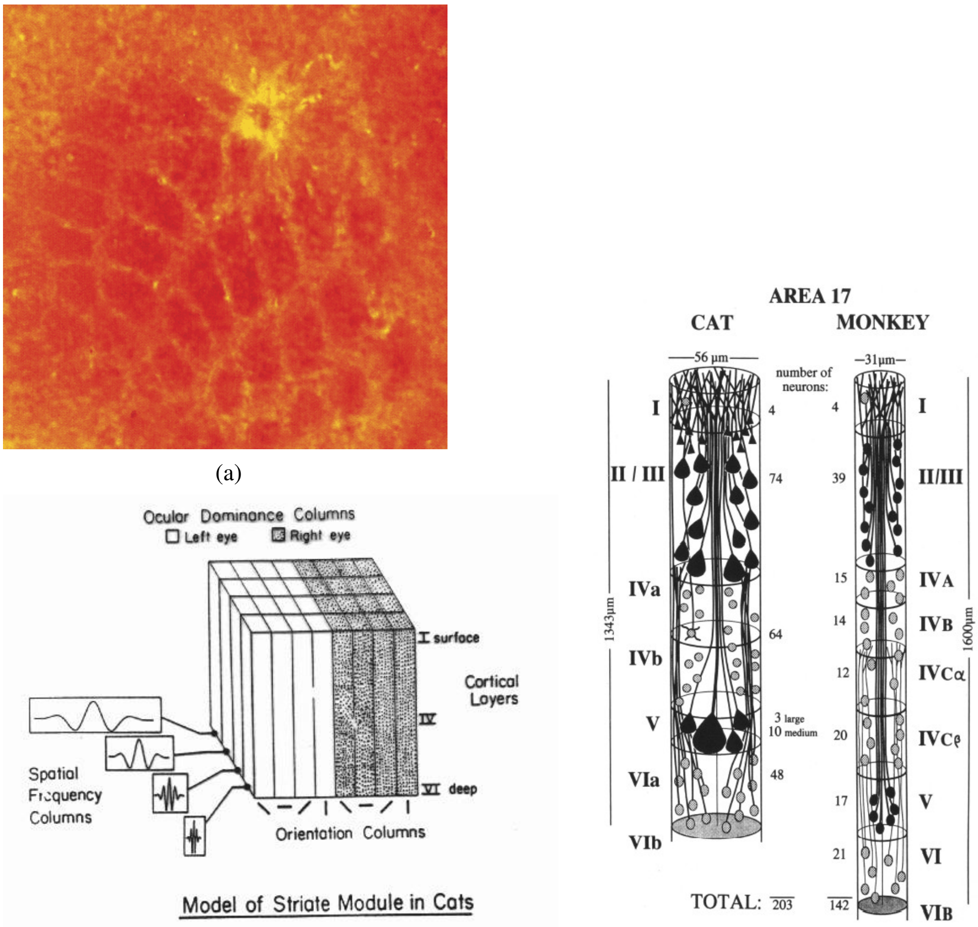
Apart from faces, there is evidence suggesting that FFA can also serve as an area of expertise (Tarr and Gauthier 2000, Gauthier et al. 2000) for other object classes. In the same sense, our model is not confined to face recognition, but could be used for recognition of any kind of object type that has a prototypical shape and requires high sensitivity to small differences among objects.

2.3 The Basic Computational Units: Cortical Columns

2.3.1 Neurobiological Background

Our system for face recognition is implemented as a large network of cortical columns. The model we use to simulate the dynamics of a column is motivated by anatomical and physiological properties of the cortex on the scale of a few hundred microns. In particular, it reflects the columnar organization of the cortex (see, e.g., Mountcastle 1997) and the concept of canonical cortical microcircuits as suggested, e.g., by Douglas et al. (1989). Columns are physiologically defined groups of neurons that extend through all cortical layers and have a diameter of roughly one mm. In some cases, they can be made visible through staining (Figure 2.2a). Depending on the perspective or the cortical area, a cortical column is commonly referred to as macrocolumn (Mountcastle 1997), segregate (Favorov and Diamond 1990), hypercolumn (Hubel and Wiesel 1977) or simply column (e.g. Yoshimura et al. 2005). In primary visual cortex, a column comprises all neurons that receive input from one point in visual space.

The analysis of the fine-structure within a column suggests disjunct populations of excitatory neurons as functional elements. Anatomically, axons and dendrites of pyramidal cells have been found to bundle together and to extend orthogonally to the pial surface through the cortical layers. All neurons that directly contribute to one such bundle form a thin columnar module of just a few tens of microns in diameter (Peters and Yilmaz 1993, Buxhoeveden and Casanova 2002), as shown in Figure 2.2c. Together with associated inhibitory neurons (see, e.g., DeFelipe et al. 1989, Peters et al. 1997) such a module was termed *minicolumn* (Favorov and Kelly 1994, Buxhoeveden and Casanova 2002, Mountcastle 2003) and was suggested as the basic computational unit of cortical processing (but see (Jones 2000) or (Rockland and Ichinohe 2004) for critical discussions). More recent evidence for disjunct functional units within a cortical column



(a) (b) (c)

Figure 2.2: Columnar organization of cortex. **(a)** Columns ("barrels") in rat barrel cortex, made visible through cytochrome oxidase staining. From (Troncoso et al. 2004) with permission of Oxford University Press. **(b)** Functional sketch of a hypercolumn in striate cortex (≙primary visual cortex of cat). Reprinted from (Valois and Valois 1990) with permission of Oxford University Press. **(c)** Drawing of pyramidal cell modules in cat and monkey primary visual cortex. Taken from (Peters and Yilmaz 1993) with permission of Oxford University Press.

comes from experiments using focal uncaging of glutamate combined with intracellular recordings (Yoshimura et al. 2005). It was found that a column has a fine-structure of functionally relatively disjunct populations of layer 2/3 pyramidal cells. The relation of these populations to the cortical minicolumn has yet to be clarified, however. The main potential difference is that the concept of a minicolumn requires neurons in a population to be spatially adjacent whereas for neurons in the functional populations described in Yoshimura et al. (2005) this is not necessarily the case.

Independent of the spatial arrangement of a column's functional sub-populations, there is little dispute about the existence of lateral coupling of such populations via a system of inhibitory neurons (Peters et al. 1997, Yoshimura et al. 2005). Yoshimura et al. (2005) for example have found the excitatory populations of layer 2/3 to receive common and population-unspecific input from inhibitory neurons of the same layer as well as from inhibitory neurons of layer 4 (see also Dantzker and Callaway 2000).

2.3.2 A Model of the Cortical Column

We will define our dynamic model of a cortical column in accordance with these experimental findings. To be somewhat independent of different terminologies used in different communities, we will refer to the cortical column simply as *column* (instead of, e.g., macrocolumn or hypercolumn) and we will refer to its functional subpopulations as the column's *units*.

Generally speaking, a column represents all relevant features that are present at one location of either external, retinotopic space (cf. Figure 2.2b), or in some internal coordinate frame. Each unit of a column represents one such feature or quality. If necessary, competition among its units allows a column to represent only the strongest qualities at its location in a soft winner-take-all manner (see below). According with anatomical findings, each unit stands for an assembly of approximately 100 neurons. Since these neurons all represent the same feature, their mean firing rate (also called *population activity*) can be used to encode that feature. Contrary to using the average firing rate of a single neuron, however, this code is much faster and more reliable (for mean-field arguments see, e.g., (Wilson and Cowan 1973, van Vreeswijk and Sompolinsky 1998, Gerstner 2000), and (Lücke and von der Malsburg 2004) for a columnar model). This fast and robust information processing is a reason why our model can achieve recognition times comparable (in neural time scales) to human performance in spite of its inherently recurrent processing.

We describe the unit's neural activity by a differential equation called *modified evolution equation*. This equation represents our model of inhibition amongst the column's units and is a generalization of the well-known deterministic evolution equation (see, e.g., Eigen 1971).

The activity x_i of the ith unit in a column of K units is given by

$$\tau \frac{d}{dt} x_i = x_i^{\nu} I_i - x_i \sum_{j=1}^{K} I_j x_j , \tag{2.1}$$

where τ is a time constant and I_i represents the input to unit x_i. The exponent ν parameterizes the competition strength among the units. This competition signal is global and changes during the recognition process (see below). However, it may be shifted in time for the different layers of the network, and it does not have an effect on all columns. In (Körner et al. 1999) the source of a fast and global modulatory signal to the cortex is discussed as the intralaminar nuclei of thalamus.

For $\nu = 0$, there is no competition, and (2.1) simplifies to

$$\tau \frac{d}{dt} x_i = I_i - x_i \sum_{j=1}^{K} I_j x_j . \tag{2.2}$$

In this case, all units represent their input proportionally, while the interaction term $\sum_j I_j x_j$ leads to activity normalization in the column (see Appendix A for a proof). For $\nu = 1$, on the other hand, we get the dynamics

$$\tau \frac{d}{dt} x_i = x_i (I_i - \sum_{j=1}^{K} I_j x_j). \tag{2.3}$$

Now we have strong competition among the units, leading to winner-take-all (WTA) behavior (again, see Appendix A for a proof and further analysis).

In our model of object recognition we assume that there are two types of columns with different functions. Dynamically, they only differ in the use of the competition parameter ν:

- *Feature columns* represent their input in a linear fashion (see Figure 2.3a). Consequently, the units in a feature column have no need to compete among each other, i.e. for them the parameter $\nu = 0$.

- *Decision columns* show a WTA behavior leading towards a state where only the unit getting the strongest input remains active. These units receive a ν-signal that linearly rises from 0 to 1[1]. So they start out with linear dynamics like feature columns. With rising ν, competition increases, finally leading to a WTA behavior that leaves only the unit with the strongest input active. The typical dynamics of a decision unit is shown in Figure 2.3b.

The crucial computations in our system are performed by decision columns, whereas feature columns serve for information representation. Both kinds of columns may actually have the same neural substrate with the only difference that feature columns do not receive (or just do not respond to) the ν signal.

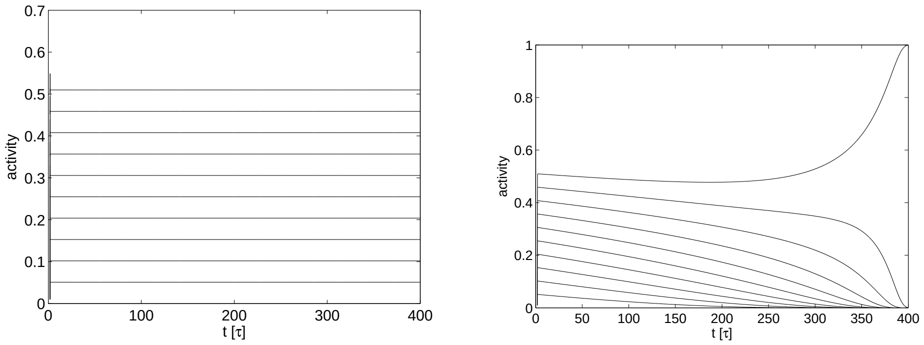
In the networks that we will introduce in the following section, units communicate with units of other columns. For this communication, a column scales the output activities of its K units such that its output energy (i.e. the Euclidean norm of the column activity vector) stays constant[2]:

$$\bar{x_i} := \frac{x_i}{\sqrt{\sum_{j=1}^{k} x_j^2}} . \tag{2.4}$$

---

[1] In principle, the competition parameter ν could be set to a constant value of $\nu = 1$. However, slowly increasing competition within the columns of a network has in earlier systems proven to efficiently avoid local optima (Lücke et al. 2008). This is related to the slow change of the temperature parameter in simulated annealing like systems (Kirkpatrick et al. 1983), which serves the same purpose.

[2] For brevity of notation, we will sometimes just use the name of a certain unit type (like $\mathcal{C}$ for control units) to denote the output of that unit. We will always point this out when we do so.



(a) Feature column ($\nu \equiv 0$).

(b) Decision column (ν rises from 0 to 1 over the shown time period).

Figure 2.3: Typical time course of the unit activities in an isolated feature column (a) and decision column (a). The inputs to the $K = 10$ units are spread equidistantly between 0 and 0.5. **(a)** After a sharp initial rise, a feature column represents its inputs in a linear fashion. **(b)** For a decision column, the competition parameter ν rises from 0 to 1 during the cycle time of $T = 400\tau$. The column starts out with activities proportional to the inputs like a feature column. Rising values if ν induce rising competition among the units, finally leaving only the unit with strongest input active. Note that the WTA behavior seen here results directly from the growth of the competition parameter ν. The internal dynamics of a column is much faster, so that with respect to the slow growth of ν, a column is always in quasi-steady-state. This can be seen also in the fast rise of the unit activities from very small initial values to the significantly higher steady states at the very start of the plot.

This kind of output normalization is advantageous for maintaining homeostasis in networks of columns and may be carried out by neurons in layer 5 of the cortex as suggested by Douglas and Martin (2004). Note that for feature columns this Euclidean normalization happens automatically in steady state (cf. Appendix A). For decision columns, explicit normalization is only necessary during the central phase of the cycle. At the beginning, it follows feature column dynamics anyway, while activity in the final state has both a 1-norm and a 2-norm of 1.

2.4 The Network

The principal architecture of the system is roughly visualized in Figure 2.4a. It consists of three main parts, an Input Layer for image representation, an Assembly Layer, and a Gallery Layer as memory. The Assembly Layer establishes correspondences between input and memory. It recurrently integrates information about feature similarity, feature arrangement, and face identity. Given an input, the integration of these information components results in the system to converge to a state that represents a percept. Figure 2.4a sketches the system after such a convergence when it has correctly established correspondences between a person's face stored in memory (i.e. in the Gallery Layer) and a given input image of this person. The principle of information integration from both the Input and the Gallery Layer in the Assembly Layer is sketched in Figure 2.4b. Note the inherent symmetry of bottom-up and top-down information flow (however, as we will see below, this information flow is realized in different ways). In the following, we will discuss the architecture of the system in detail.

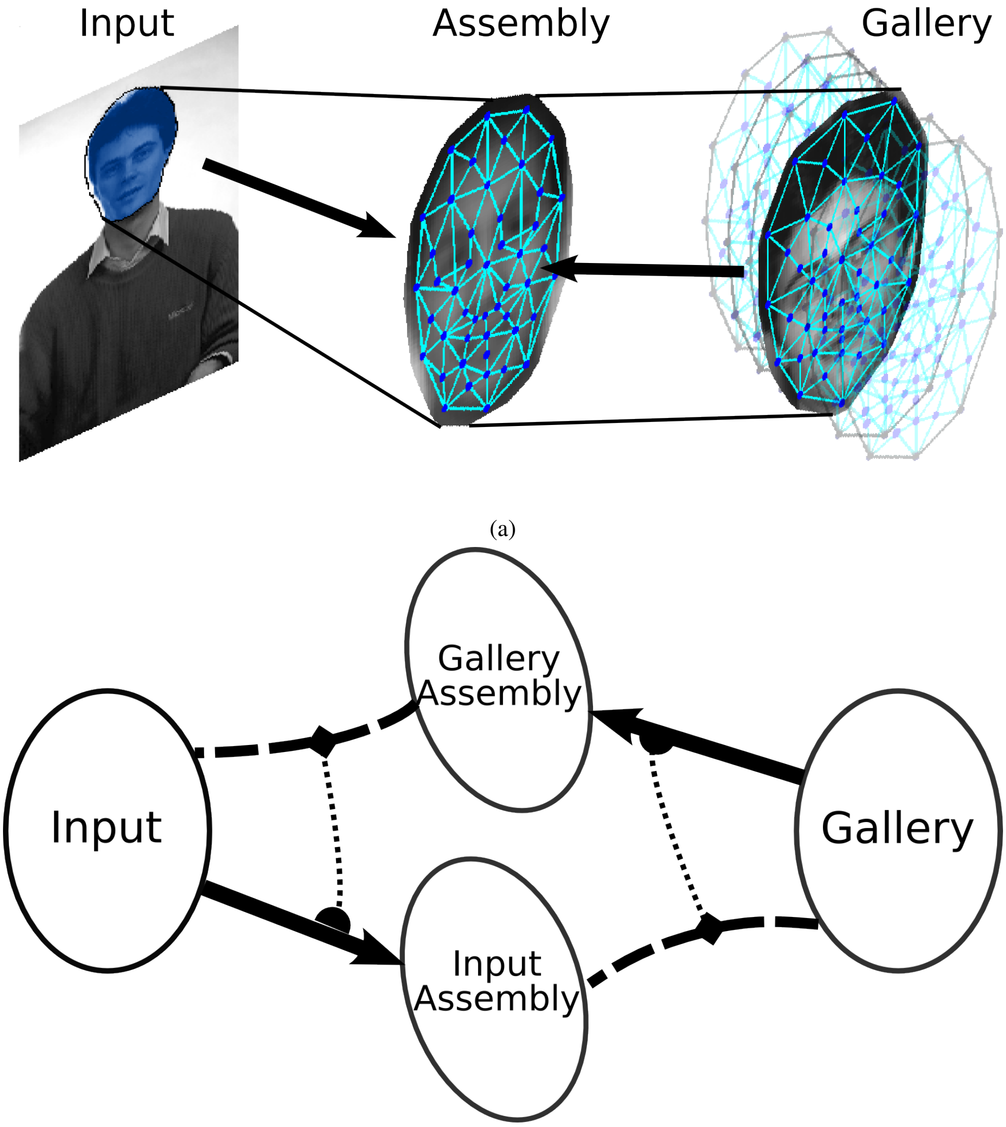
As we could see before, the largest subunits of the network are *layers*. These loosely correspond to the different cortical areas that make up the visual system (we are not speaking here of the layers of anatomically different neurons that can be distinguished within one area of cortex). Layers are organized topologically, with a topology that may be stimulus space, like in V1 and somatosensory cortex, or a more abstract space. The layers of our network interact recurrently and activity collectively converges towards a final state that represents the "percept" of the network, in our case the possible recognition of a face.

Layers may contain both feature columns and decision columns. If we assume every feature column to represent all relevant features at one position of a retinal image, then layers of feature columns can represent whole images. The network introduced below uses layers of two different spatial arrangements:

- Rectangular grid: Straightforward representation suitable for any image. Every column represents one specific geometric location (see Figure 2.5a).

- Face graph structure: An arrangement specifically suited for faces, where each column represents an important landmark position on a face (Figure 2.5b). Note that in this case, a column does not necessarily represent a fixed spatial location in the image, but rather a fixed semantic location (nose, mouth, eye, chin, etc.). Spatial locations of landmarks can change according to the face they represent.

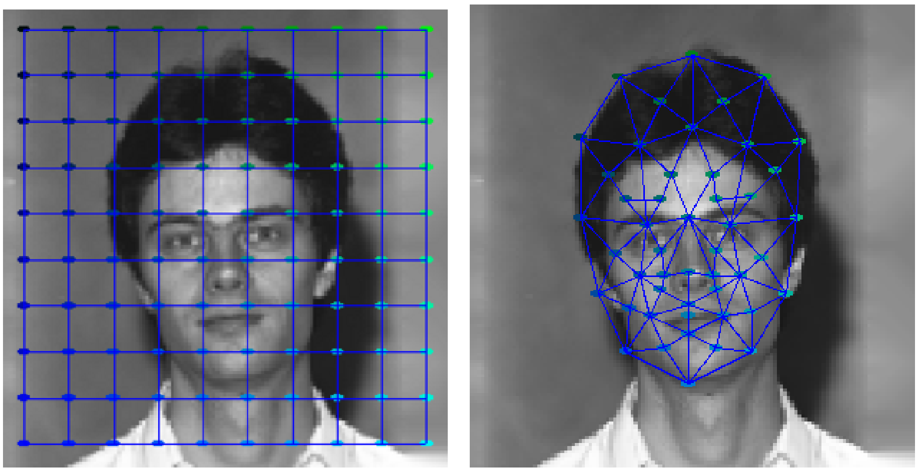
The network consists of the following three layers (see Figure 2.6):



(a)

(b)

Figure 2.4: Principal layout of the system. **(a)** The system has to simultaneously represent information about position *and* identity of the input face and its parts. Positional information is represented by *dynamic links* establishing correspondences between points in the input image and an in the internal reference frame ("Assembly Layer"). Identity information is represented by the activity of Gallery units, different graphs storing memories of different faces. **(b)** Both modalities contribute to the activity of the internal Assembly Layer, which represents visual information in its two sublayers Input Assembly and Gallery Assembly. Information flow to the Input Assembly is controlled by correspondences between Input and Gallery Assembly, while information flow from the Gallery to the Gallery Assembly depends on the similarity of the Input Assembly and models stored in the Gallery.



(a) Rectangular grid (b) Face graph

Figure 2.5: Different representations of facial images. A rectangular grid graph (a) is used for input image representation, a face graph (b) consisting of characteristic points (landmarks) is a dedicated data structure used for internal face representation.

- Input Layer $\mathcal{I}$: Represents the input image in a rectangular grid.
- Assembly Layer: Integrates intermediate information from both the input image (represented in the *Input Assembly* units $\mathcal{IA}$, see Figure 2.7) and the gallery (represented by the *Gallery Assembly* units $\mathcal{GA}$).
- Gallery Layer $\mathcal{G}$: Represents all gallery faces in terms of the weights of its afferent and efferent connections to the Assembly Layer.

The following three subsections describe these layers in detail.

2.4.1 Input Layer

The Input Layer represents the input image using 400 feature columns arranged in a rectangular grid of $P = 20 \times 20$ points. Each feature column represents by its units' activities K features extracted from the image at that position.

If we neglect color and binocularity, the response properties of neurons in primary visual cortex are commonly described by the well-known Gabor wavelets (Ringach 2002, Jones and Palmer 1987, Daugman 1980). In our model we use a predefined set of Gabor wavelets that appropriately sample orientation (over 8 orientations) and spatial frequency (over 5 scales) space, resulting in a number of $K = 40$ features at each point. That is, we use Gabor filter responses to model the RFs of the feature units in the Input Layer. For extracting the filter responses, we use the standard Gabor transform, as described in Appendix B. As feature values we use the magnitude $\mathcal{J}$ of the responses, thus ignoring Gabor phase, to model complex cell responses

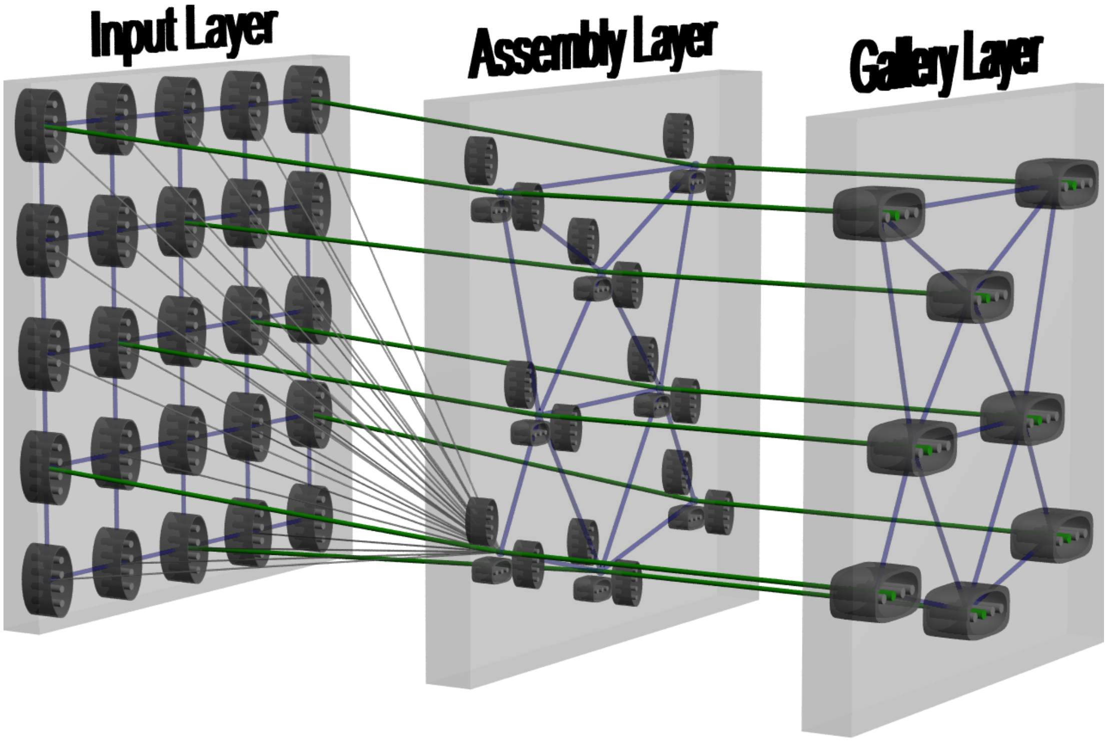


Figure 2.6: Architecture of the network. The gray oval structures represent columns (the vertical ones feature columns, the horizontal ones decision columns), with units as lighter cylinders inside. The numbers of units and columns shown here are chosen exemplarily for visualization purposes only and are not identical to the real numbers of units used in this work. The Input Layer is organized in a rectangular grid (represented by the light lines connecting columns), while both the Assembly Layer and the Gallery Layer have face graph topology. At each landmark in the Assembly Layer there are three columns, two feature columns of the Input Layer and Gallery Assembly, and one control column. Input and Assembly are connected all-to-all (shown exemplarily for the left-lowermost point in the Assembly Layer), while Assembly landmarks are connected only to the same landmarks in Gallery, but to all identity units there (see also Figure 2.7). The dark lines connecting the three layers and the subset of dark ($\widehat{=}$ activated) Gallery units represent a possible final state of the network.

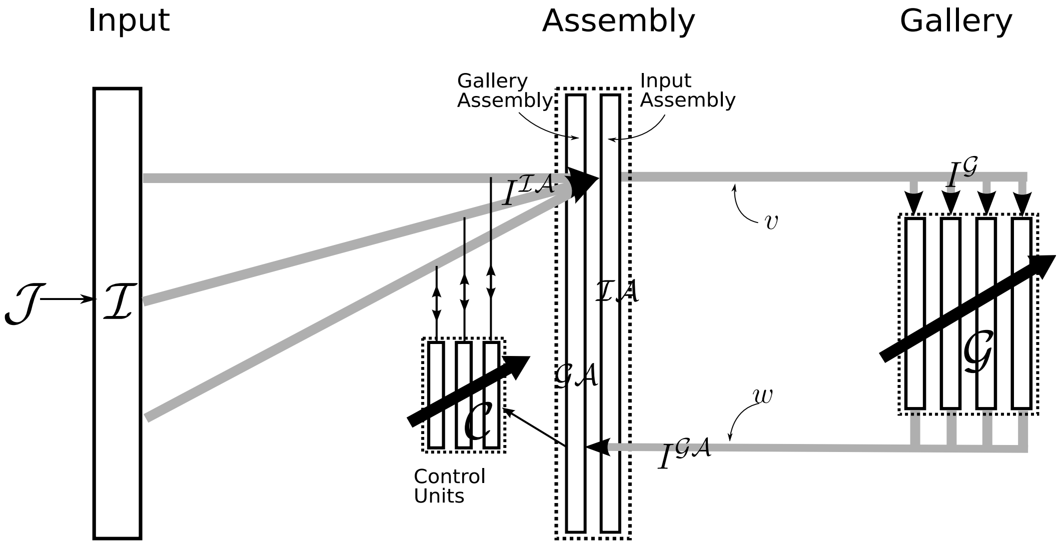


Figure 2.7: Information flow in the network. Visual information in form of Gabor jets $\mathcal{J}$ extracted from an input image activates the Input Layer $\mathcal{I}$. It flows to the Assembly Layer (Input Assembly, $\mathcal{IA}$) and from there to the Gallery $\mathcal{G}$, where it activates via receptive fields v some memories more strongly than others. Information representing the active memories (stored in projection fields w analogous to v) flows back to the Gallery Assembly $\mathcal{GA}$. Information flow $I^{\mathcal{IA}}$ from the Input Layer to the Input Assembly is modulated by the control units $\mathcal{C}$, which in turn are driven by the similarity of those image patches in the Input Layer and the Gallery Assembly that they connect. By activating those control units that connect positions of the Input Layer containing similar information as the Gallery Assembly, the system effectively focusses on those parts of the input image that contain visual information most similar to the current reconstruction in the Gallery Assembly, formed by superposition of active units in the Gallery Layer. The thick black arrows represent the competition among the decision columns of which the Gallery and the control columns consist. The symbols correspond to those used in the text.

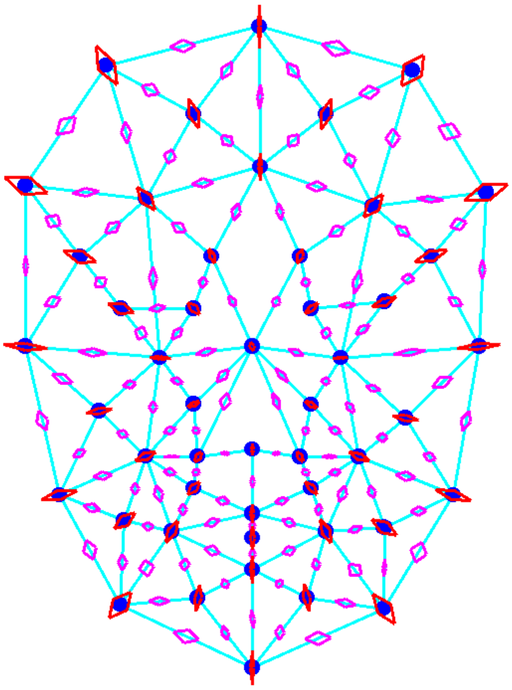


Figure 2.8: Average face graph. The diamonds around the nodes denote the first and second moments of the standard deviation of landmark positions. The diamonds on the edges denote standard deviation of landmark distance.

(Hubel and Wiesel 1977). Implicitly, Gabor phase is still represented by the positions of the feature columns in the input image. In applications using Gabor features it has turned out that with $K = 40$, as above, good results can be achieved (Wundrich et al. 2004). Performance increases for more wavelets, but 40 represents a good compromise between performance and computational cost.

Each Input Layer unit being responsive to a certain Gabor feature $\mathcal{J}_i^p$ at its position p on the input grid, the unit activities follow the dynamics (cf. (2.2))

$$\tau \frac{d}{dt} x_i^{\mathcal{I}_p} = \mathcal{J}_i^p - x_i^{\mathcal{I}_p} \sum_{j=1}^{K} \mathcal{J}_j^p x_j^{\mathcal{I}_p}. \tag{2.5}$$

2.4.2 Assembly Layer

The Assembly layer integrates intermediate information from both the input image (represented in the *Input Assembly* units) and the gallery (represented by the *Gallery Assembly* units, see Figure 2.7). The role of the Input Assembly is to represent a normalized version of the input image, while the Gallery Assembly accommodates a weighted average of all Gallery faces. This information is organized in a face graph arrangement with $Q = 48$ landmarks (see Figure 2.5b). Since the face graph in the Assembly Layer has to be able to represent many different faces, we determine its geometry by averaging over several hundred face graphs of individual faces:

Assume M face graphs given, each with Q landmarks at positions. All coordinates are normalized relative to the full image width and height, going from 0 to 1. To discount possible different relative positions of the phase graphs in the coordinate system, the average graph is calculated in a translation-invariant fashion and centered around the image center. More specifically, the coordinates used from every single face graph to calculate the average graph are discounted by the amount that the center of mass of this face graph deviates from the image center $(0.5, 0.5)$. Let

$$\mathbf{c}_m = \frac{1}{Q}\sum_{q=1}^{Q} \mathbf{x}_{q,m}$$

be the center of mass of a face graph m. Then we calculate the average position of a landmark q as

$$\mu_q = \frac{1}{M}\sum_{m=1}^{M} \mathbf{x}_{q,m} - \left(\mathbf{c}_m - \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}\right),$$

which will produce an average graph with its center of mass at the image center. Additionally, it is possible to calculate the standard deviation of each average landmark position for a certain population of face images. This may be useful to assign different reliability to the different landmarks, or to define directions in which the average graph should be flexible. While we consequently calculated this information (see Figure 2.8), we did not use it in the scope of this work.

The columns of the Input Assembly and Gallery Assembly are feature columns, i.e. they integrate their inputs (defined below) according to (2.2). The input $I^{\mathcal{IA}}$ to the ith Input Assembly unit at position q of the face graph is a weighted sum of the ith Gabor feature at all grid positions p of the Input Layer, modulated by the respective control units:

$$I^{\mathcal{IA}}_{q,i} = \frac{1}{\sqrt{P}}\sum_{p=1}^{P} \mathcal{C}_{p,q}\mathcal{I}_{p,i}, \tag{2.6}$$

with $\mathcal{C}_{p,q}$ the output strength of the dynamic link (see below) controlling the flow of the output of Input column $\mathcal{I}_p$ to Input Assembly column $\mathcal{IA}_q$.

The input $I^{\mathcal{GA}}$ to a Gallery Assembly column at position q is the superposition of all Gallery projective fields $w_{q,m}$, weighted by the activity of the respective Gallery unit $\mathcal{G}_{q,m}$:

$$I^{\mathcal{GA}}_{q,i} = \frac{1}{\sqrt{M}}\sum_{m=1}^{M} w_{q,m,i}\mathcal{G}_{q,m}, \tag{2.7}$$

with the “efferent weight” $w_{q,m,i}$ representing the strength of Gabor feature i in landmark q of Gallery image m (of M in total).

Control units

The Assembly Layer also contains the *control units* mentioned above, which mediate the signal coming in from the Input Layer. These control units provide potential connections (*dynamic links*) between every Input Layer point to every point in the Input Assembly. The activity of the control units is driven by the feature similarity of the corresponding points in the Input Layer and the *Gallery* Assembly. That is, the similarity between the non-normalized input face in the Input Layer and the weighted average face in the Gallery Assembly controls via the control units how input information flows to the Input Assembly. In that sense the control units define a geometric mapping between Input and Assembly Layer. Additionally to the feature similarity input, control units get support from neighboring control units that represent similar mappings (see Figure 2.9 and paragraph below for details.).

The dynamic links are decision units, meaning that their dynamics follow (2.1). The input $I^{\mathcal{C}}$ to a dynamic link $\mathcal{C}_{p,q}$ connecting input position p and assembly position q is given by the scalar product between both column outputs plus a topological interaction term:

$$I^{\mathcal{C}}_{p,q} = \sum_{i=1}^{K} \mathcal{I}_{p,i}\, \mathcal{GA}_{q,i} + \frac{c_{\text{top},\mathcal{C}}}{|\text{neighbors}|} \sum_{\tilde{p},\tilde{q}} f_{\text{top}}(p,q,\tilde{p},\tilde{q})\,\mathcal{C}_{\tilde{p},\tilde{q}}, \tag{2.8}$$

where $c_{\text{top},\mathcal{C}}$ defines the maximal strength of topological interaction between control units (see below), and $|\text{neighbors}|$ is the number of topological neighbors the control column has in the face graph.

Topological cooperation among control units

As mentioned before, there is topological cooperation among the control units of the Assembly Layer. The purpose of this cooperation is to establish a continuous mapping between the different geometries of the Input Layer and the Input Assembly. A given dynamic link connects a specific column A of the Input Layer with a column B of the Input Assembly. Due to the geometry of both layers, both columns represent distinct positions $\vec{z}_A$ and $\vec{z}_B$ in retinal coordinates and internal image representation space, respectively. Consequently, the dynamic link between them represents a certain geometric distance $\vec{d_i} = \vec{z}_B - \vec{z}_A$.

The idea is now to have topological connections in order to support parallel or near-parallel dynamic links. Therefore we define the strength of a topological connection between any two dynamic links i and j whose columns are neighbors in the face graph through a monotonically decreasing function of their non-parallelity/disparity:

$$f_{\text{top}}(i,j) = f(\|\mathbf{d_j} - \mathbf{d_i}\|_\mathbf{2}). \tag{2.9}$$

Here we use a linearly decreasing thresholded function of the form

$$f(y) = \max\left(0, 1 - \frac{y}{\beta}\right). \tag{2.10}$$

Thus topological interaction is always positive and acts only between more or less (depending on β) parallel neighboring links. This principle is depicted in Figure 2.9. To obtain the topological interaction f_{top} in (2.8) between two control units $\mathcal{C}_{p,q}$ and $\mathcal{C}_{\tilde{p},\tilde{q}}$, we first calculate from the coordinates of the columns they control in the Input and the Assembly Layer the geometric distances $\vec{d}_{p,q}$ and $\vec{d}_{\tilde{p},\tilde{q}}$ represented by them. From these we calculate the disparity of the two control columns according to (2.9) and the topological interaction via (2.10).

2.4.3 Gallery Layer

The Gallery Layer represents all M gallery face images in a face graph of Q decision columns. Each column corresponds to one landmark, with the units representing specific feature vectors for the individual faces at the respective landmarks by their afferent and efferent connections (see Figure 2.7). The units in the Input Assembly activate the Gallery units through receptive fields v representing the stored facial landmark features, activating more strongly units of faces that are similar to the normalized input image in the Input Assembly:

$$I^{\mathcal{G}}_{q,m} = \sum_{i=1}^{K} v_{q,m,i}\, \mathcal{IA}_{q,i} + \frac{c_{\text{top},\mathcal{G}}}{Q} \sum_{\tilde{q}=1}^{Q} \mathcal{G}_{\tilde{q},m}. \tag{2.11}$$

Additionally, there is interaction among the Gallery units, with $c_{\text{top},\mathcal{G}}$ defining how strongly Gallery units representing the same face cooperate. That is, all landmarks that belong to the same face cooperate, and at each landmark the corresponding features of all different faces compete.

The Gallery projects a weighted superposition of its stored faces to the Gallery Assembly through efferent weights w that are identical to its afferent weights v (cf. (2.7)). See Section 2.6 for some remarks on why we think this dual representation is crucial for any full visual system. Point-to-point comparison with the Input Assembly and competition among stored models leaves only the correctly recognized identity active in the end.

2.5 Results

We now simulate the dynamics defined in the above sections using natural images of faces as input and as memories in the gallery. The size of the network is determined by the following parameters. The input grid consists of $P = 400$ columns, all face graphs (Assembly and Gallery Layers) contain $Q = 48$ columns. Consequently, there is a total of $400 \times 48 = 19\,200$ control units. For representation of visual information in the Input and Assembly Layers, we use $K = 40$ Gabor wavelets. The number M of Gallery faces depends on the size of the database on which the system is tested. If not stated differently, we used the following settings and parameters. The radius for topological interaction among control units was $\beta = 0.05 \times \text{imagesize}$. Maximum strength of this topological interaction was $c_{\text{top},\mathcal{C}} = 3.5$. Cooperation strength between neighboring gallery units was $c_{\text{top},\mathcal{G}} = 0.1$.

Note that we can numerically simulate dynamics (2.1) without specifying a value of the time constant τ. As long as the simulation time T remains constant relative to τ, simulation results will be independent of τ. The question of how the time-course of the dynamics translates to recognition times in biological terms does, however, crucially depend on the actual choice of τ.

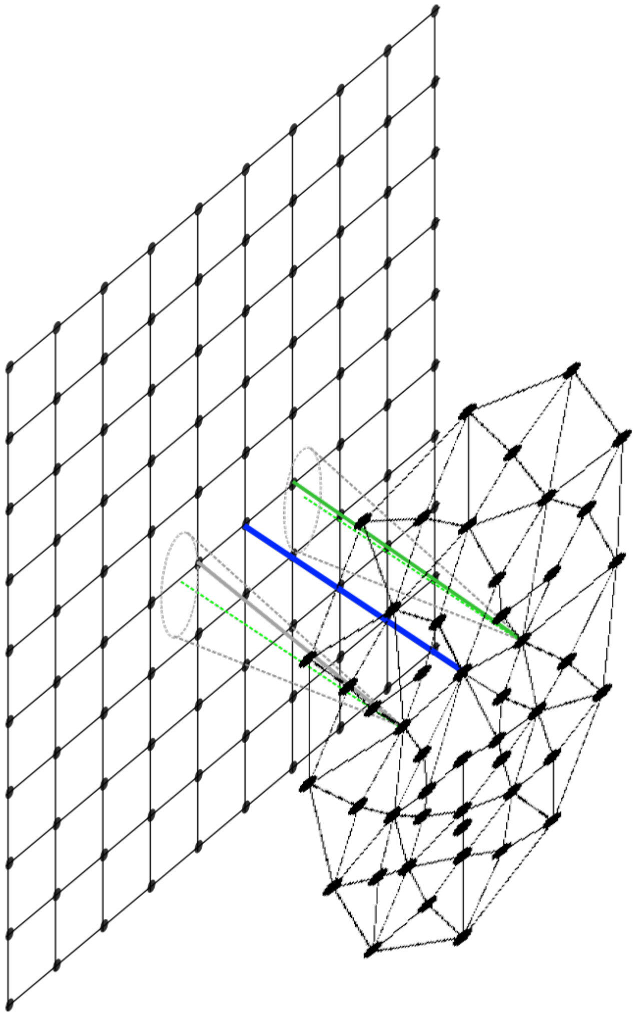


Figure 2.9: Interaction among control units to achieve a topologically consistent (i.e., continuous) mapping. The unit controlling the dark link strengthens control units in neighboring columns that represent links of similar orientation. Maximal cooperation would occur with perfectly parallel links (the dashed axes of the grey cones). Since in reality links in the network only exist to the nodes of the input grid (full lines), the strength of cooperation depends on the degree of parallelity with the dark link, equivalent to the distance of a link's end point from the cone center.

We chose a time-constant of $\tau = 0.2\mathrm{ms}$ and the length of the ν cycle as $T = 400\,\tau$. This results in a system that selects the winning sub-populations of its decision columns in about $80\mathrm{ms}$ (with $T = 400\,\tau$, compare Figure 2.3). The whole network could consequently converge to a face position and identity within about the same time. A biophysical source for this oscillating ν signal might be oscillations of brain activity in the gamma or theta range (for the role of oscillations in the brain for perception, see, e.g., Singer 2003). Numerical simulations of a single column with explicitly modeled spiking neurons suggest an even smaller time-constant (see Lücke and von der Malsburg (2004) or compare Muresan and Savin (2007) for population activation times on the order of $10\mathrm{ms}$ which suggest similarly fast deactivation times). All units have a small, but non-zero initial activity $x(0) = 0.01$. To integrate the Equations (2.1), we simply use the Euler method but adapt its time step dynamically to the average change of activity in the network in order to keep the system stable. For this, we regulate global network activity by controlling signals like the average change of columnar activity. Within a predefined maximal range, integration speed is decreased when these signals are too high, and it is increased when they fall below a certain threshold. We believe that such homeostatic processes are key ingredients to building large artificial networks that are adaptive and robust at the same time.

2.5.1 General Network Behavior

The units in the Input Layer, which receive input directly from the incoming image (cf. (2.5)), quickly converge to a state where they represent the input image via the different Gabor feature values at all grid positions. This information flows to the Input Assembly modulated by the activities of the control units (2.6) which connect every point in the Input Layer with every point in the Input Assembly. Since initially all control units have equal activity, this leads to a superposition of image information from all Input Layer points at each Input Assembly location, resulting in a feature-less, more or less homogeneous image in the Input Assembly (first image in Figure 2.10). In the Gallery Layer, all faces are equally active initially. The Gallery Assembly, which receives input from all Gallery units (2.7), will therefore initially receive a superposition of all Gallery faces, resembling an "average face" (like the first image in Figure 2.11).

To each control unit in the Assembly Layer a unique pair of feature columns is assigned, one in the Input Layer and the other one in the Gallery Assembly. The control units are driven by the similarity (expressed in terms of the scalar product) of the information stored in their dedicated feature columns, see (2.8). Therefore control units that connect points of the average face with similar input points will become stronger, while control units representing irrelevant matches will be weakened. Over the process of recognition, the activity distribution of the control units becomes more and more sparse, until it finally represents a unique mapping between the Input Layer and the Assembly Layer (see left column in Figure 2.10). Since purely local similarity of images can be quite ambiguous, the additional topological interaction among the control units is necessary in this process to achieve a globally consistent match. As the information flow from the Input Layer to the Input Assembly is modulated by the control units, the image in the Input Assembly will start to develop from a gray nondescript superposition to a more and more clear version of the input image (right column in Figure 2.10). It may be shifted and possibly distorted such that it conforms to the topology of the face graph of the Gallery Assembly.

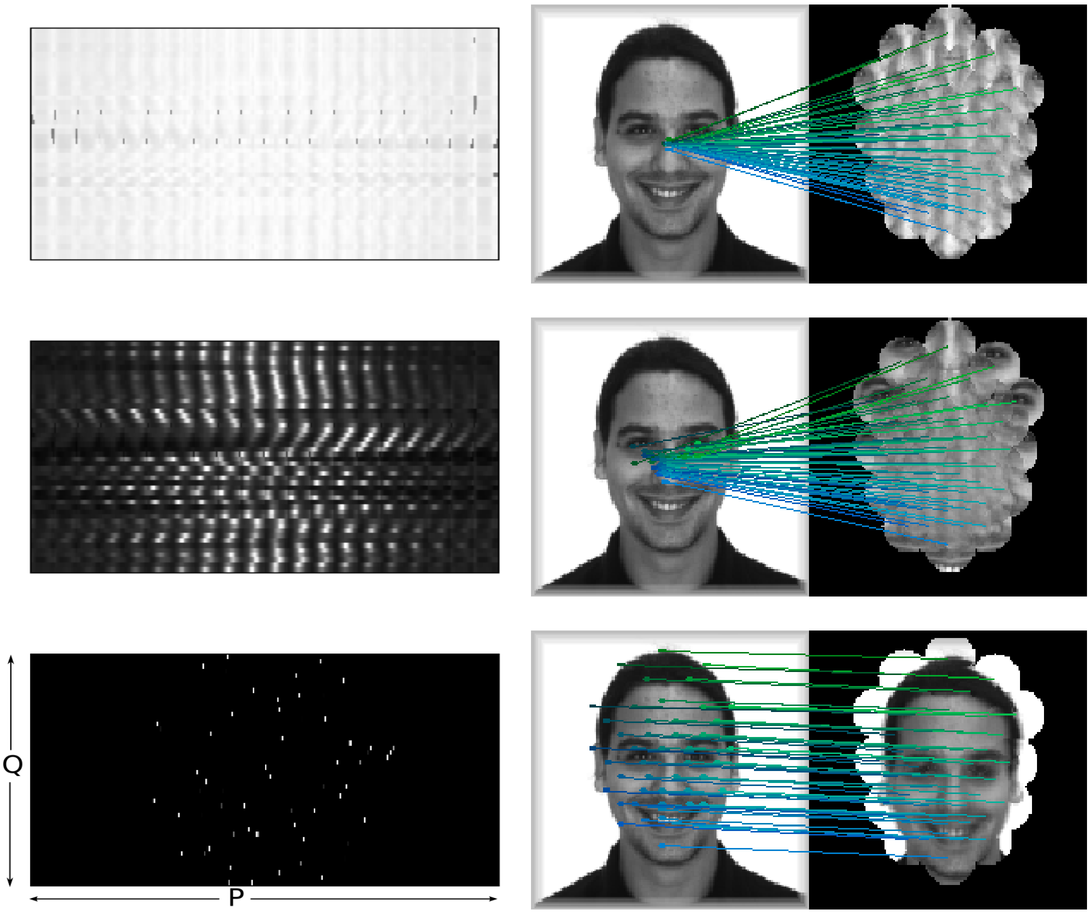


Figure 2.10: The process (from top to bottom) of finding the correct mapping between the Input Layer and the Input Assembly. Each row shows the control unit activities on the left side (white means high relative activity; values are scaled individually for each image to exploit the full dynamic range.), and on the right first the constant input image, and then an image reconstructed from the activities of the 48 landmarks of the Input Assembly (each little patch is a superposition of the Gabor wavelets of all units of one column, weighted by the activity of the respective unit.). Initially, the control units all have nearly identical, but small activity, and therefore the Input Assembly receives a superposition of all input information, resulting in the same uniform image information at all landmarks (row one). With the control units developing a topologically consistent match between Input and Input Assembly (rows two and three), this image starts to differentiate towards a normalized (i.e. shifted and deformed if necessary) version of the input image. The mapping via the control units is also visualized by the colored lines connecting the input image with the Input Assembly. Each line represents the "center of mass" of a control column, i.e. the location in the input image where its units are pointing to as a group, weighted by their activity.

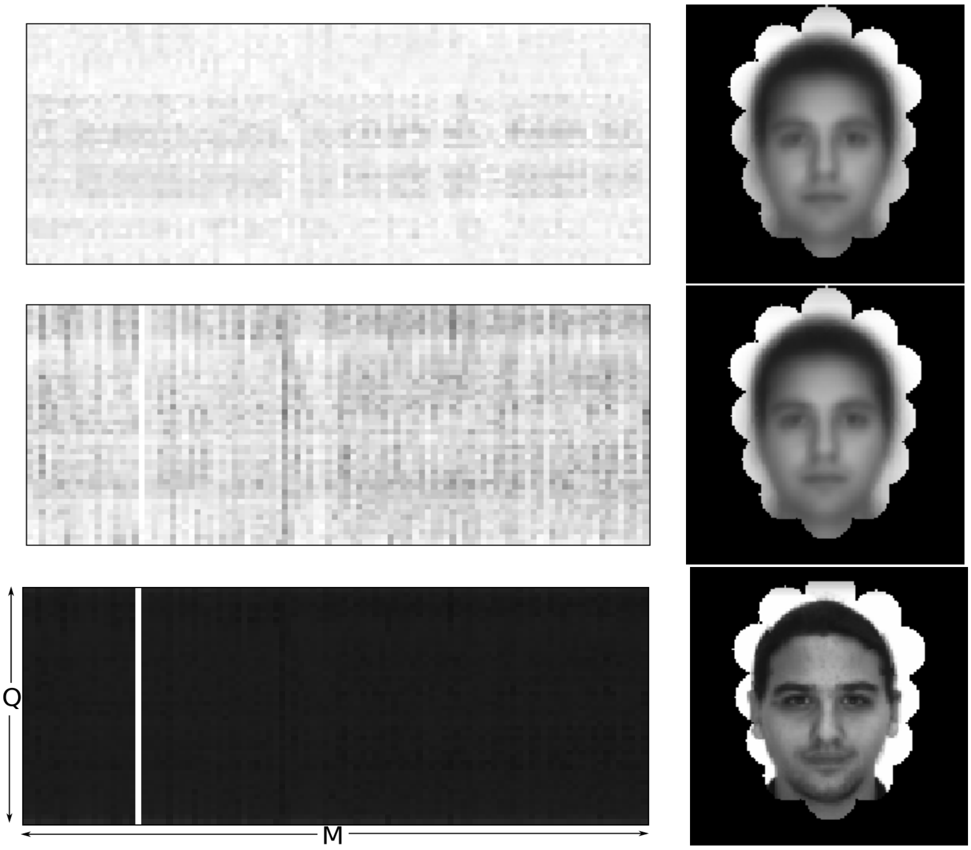


Figure 2.11: Time course (from top to bottom) of the Gallery unit activities (left, values are scaled by the maximum activity for each image separately) and of the resulting image representation in the Gallery Assembly (right). The Gallery Assembly gets input from all Gallery units and thus contains an activity weighted average of all faces in the gallery. Initially, when all Gallery units are nearly equally active, this weighted average is a real average of all gallery faces, i.e. a mean face (uppermost row). With ongoing dynamics and rising competition, the Gallery units fitting the input image better get stronger, and the Gallery Assembly activity develops towards the respective gallery faces. Finally, only one unit of all Gallery columns is active, and the Gallery Assembly contains a representation of the image the system has recognized (which is not identical to the input image in most applications, cf. input image in Figure 2.10).

The image information in the Input Assembly in turn acts as input to the Gallery units, where it gets filtered through the individual receptive fields of the units (2.11), exciting those units more that represent faces more similar to the input image. Owing to competition between the units of each Gallery column and cooperation among units of different landmarks representing the same face, the Gallery will start to favor some of the stored faces over others (cf. left column of Figure 2.11). This in turn changes the image in the Gallery Assembly from an average face to a superposition that is already biased towards one or several of the better fitting gallery faces (second and third face image of Figure 2.11). This sharpened target face now helps to position the normalized input image even more precisely, and so forth. In the final state, the Input Assembly will contain a shifted and maybe distorted version of the input image, while in the Gallery Layer the units of only one face are still active, and the Gallery Assembly contains a copy of that face of the Gallery that the system judges to be most similar to the input image.

In some cases, the system has not yet found a global optimum (i.e. a consistent match) after the first ν cycle. In these cases, letting the system run for a second or third cycle usually produces the right result. In that sense, our system will always produce *some* result after a limited time, but it will continue to improve this result if given more time.

2.5.2 Position Invariance

Recognizing objects invariantly of their exact position is a crucial ability of visual animals. Position invariant recognition in humans is limited by the log-polar mapping from retina to cortex, which represents peripheral areas poorly, and is not perfect at the fovea either (see, e.g., Cox et al. 2005). As a general principle, however, position invariance is well accepted and forms the basis of psychophysical paradigms like “popout” (Humphreys and Heinke 1998, Thornton and Gilden 2007). Even processing of totally novel objects seems to be position invariant (Bar and Biederman 1999, Fiser and Biederman 2001), suggesting generic, object independent mechanisms for position invariant recognition.

The system presented in this work possesses such inherent position invariance. As was outlined in Section 2.4.2, control columns are not biased towards a certain *absolute* position in the input image, but only try to represent a similar *relative* shift as their neighbors. In that sense, the result of the matching process can be a match to any position of the input image as long as it is globally consistent.

To assess the position invariant matching capabilities of our system, we store a single face in the Gallery as a model image. For this experiment, we organize all layers as rectangular grids and use an Input Layer four times as large as the Assembly Layer. Thus the task of the system is now to detect somewhere in the large Input Layer that sub-image that resembles most closely the model face in the Gallery. Figure 2.12 illustrates how the model handles this. Of course, the system is also position invariant when doing recognition from galleries containing many models. For this we refer to the experiments in Section 2.5.4 (where the topology of the Assembly and the Gallery is again a face graph like in the standard system architecture).

Note that if the target face in an input image sits so far at the periphery that parts of it are not represented by the Input Layer anymore, the graph in the Assembly Layer cannot find complete correspondence in the Input Layer. in this case it is necessary to extend the topologic interactions

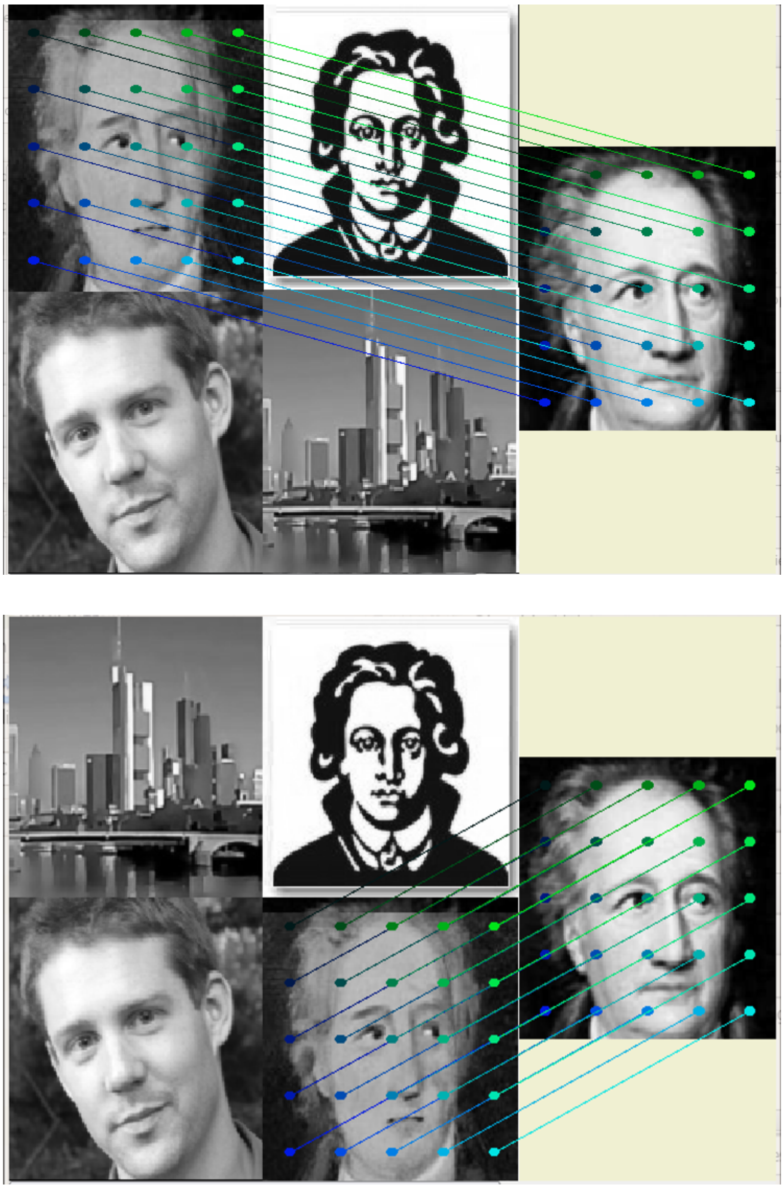


Figure 2.12: Position invariance of our matching process. The Input Layer, containing a collage of several images, is four times as large as the Assembly Layer. Above, the final states of two matching processes are depicted. The input image is shown on the left and the model image to the right of it. Active dynamic links are shown as colored lines connecting input and model image. In both cases, the system successfully selects that sub-image in the Input Layer that resembles the model image most closely. Note that this process may take two ν cycles until the correct match is achieved, with the system sometimes first only reaching an inconsistent match due to interfering lateral influence from the other parts of the input image.

Figure 2.13: A sample of 30 faces from the FERET database.

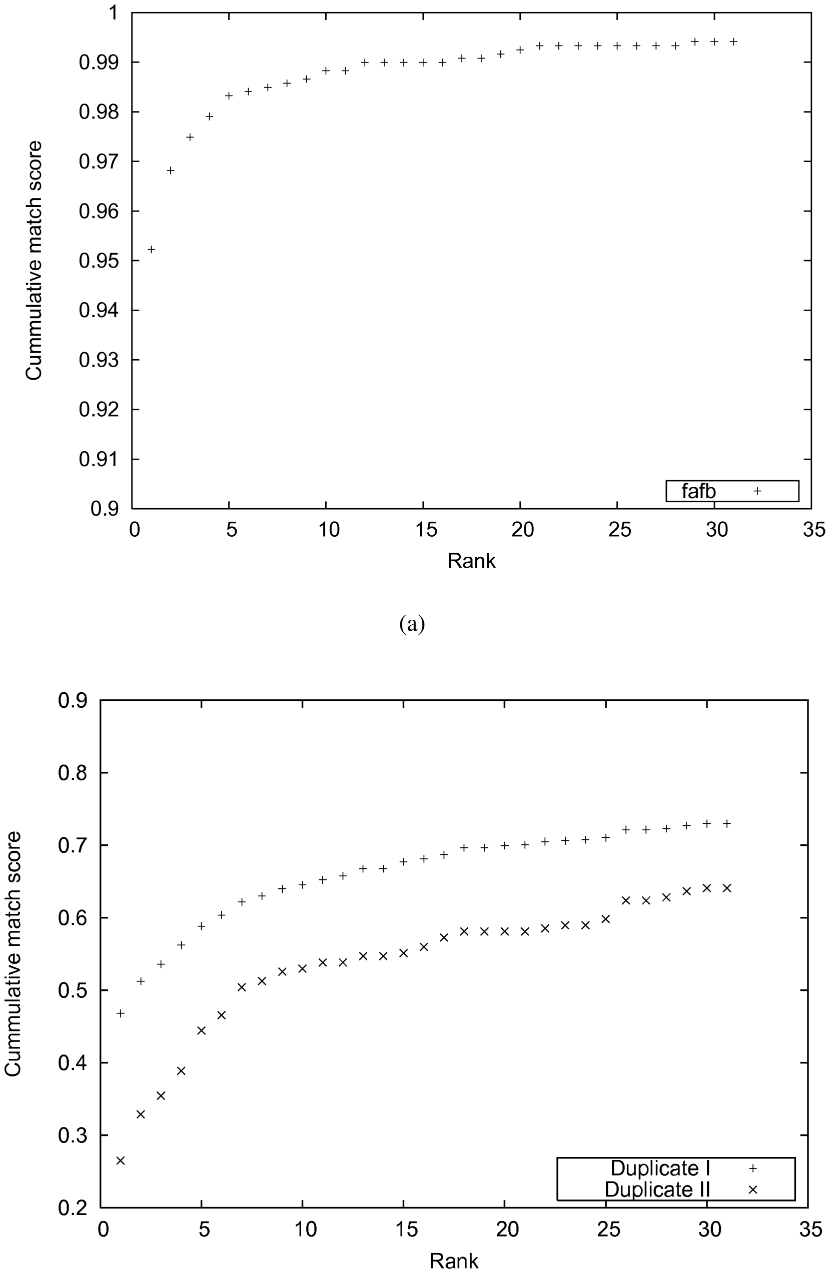
of the control units to wrap-around cooperation, enabling the nodes of the Assembly graph that would otherwise not find a partner to match to points on the opposite side of the input image so that the consistency of the graph matching is preserved.

2.5.3 Tests on Standard Databases

To quantitatively compare our system to other approaches, we tested it on the FERET (Phillips et al. 1998) and the AR (Martinez and Benavente 1998) benchmark databases. We followed the testing protocols of (Phillips et al. 2000, the official FERET evaluation) and (Tan et al. 2005). The FERET database contains images of 1196 individuals (a sample of 30 of them is shown in Figure 2.13), while the subsets of the AR database used in (Tan et al. 2005) and by us contain 100 faces. Accordingly, the size of the Gallery when testing these databases is $M_{\text{FERET}} = 1196$ faces and $M_{\text{AR}} = 100$ faces, respectively.

In order to test the performance of a face recognition system, it is confronted with a gallery of images of all faces in the database, and is then asked to identify a different set of images containing pictures of (possibly a subset of) the faces in the gallery photographed under different conditions. Often not only the best match chosen by the system is recorded, but also the follow-up matches. This allows to construct *cumulative match scores*, the match score of rank n representing the fraction of test images whose correct match appears among the n best matches found by the system.

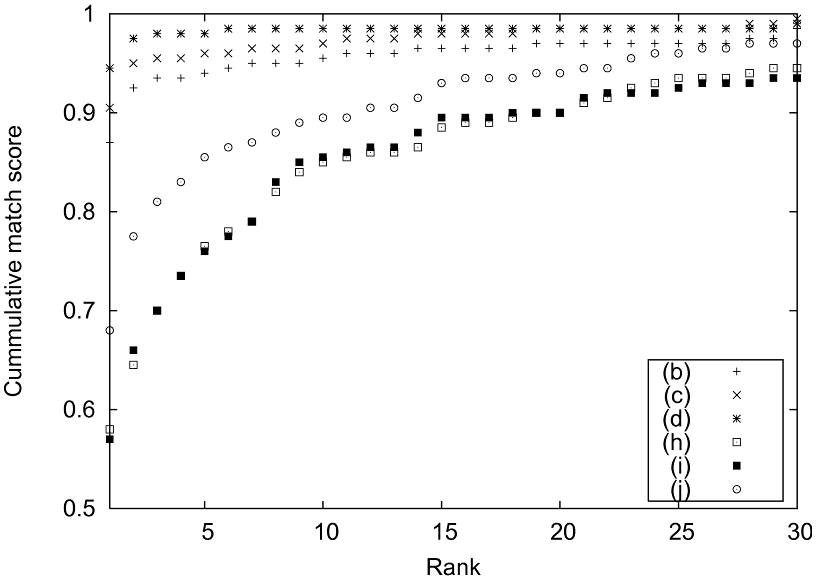
From the FERET database we used the following testing subsets: The set `fafb` contains photographs of 1195 individuals taken on the same day as the gallery images, but with the subjects showing a different facial expression. The set `Duplicate I` contains 722 of images that were taken at least one day but less than 18 months after the gallery images. Finally, the set `Duplicate II` contains 234 images taken more than 18 months after the gallery images. The cumulative match scores of our system for this database are shown in Figure 2.14. The AR Face Database contains several testing subsets with images of the same 100 subjects that make up the gallery. The subsets b, c, and d contain images of the subjects smiling, expression anger, or screaming, respectively. While those subsets were taken on the same day as the gallery images, the subsets h, i, and j show the subjects at a later session expression those same three emotions. Subsets e and f show subjects wearing sunglasses and scarfs, and subsets k and l show the same situation at a later date. Cumulative match scores for this database are shown in Figure 2.15.



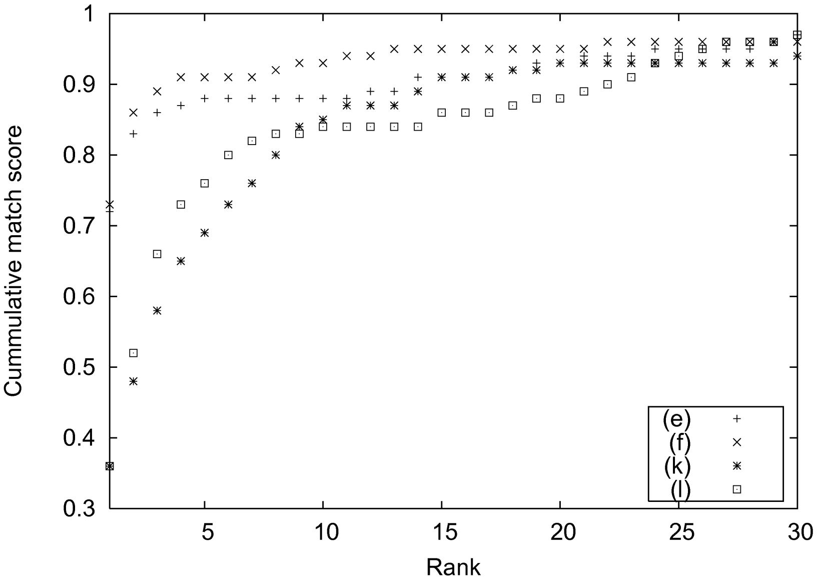
(a)

(b)

Figure 2.14: Cumulative match scores for the FERET database. **(a)** Performance on the fafb dataset (different pictures taken in the same day). **(b)** Performance on the duplicate datasets.



(a)



(b)

Figure 2.15: Cumulative match scores for the AR database. **(a)** Performance on the emotion datasets. **(b)** Performance on the occlusion datasets.

recognition rates [%]		our system	Phillips et al. (2000)	Tan et al. (2005)
FERET	fafb	95	95 (85)	92/-
	duplicate I	47	59 (40)	
	duplicate II	26	52 (22)	
AR	Emotion	91		95/82
	Em. duplicate	61		81/82
	Occlusion	73		96/81
	Occ. duplicate	36		56/51

Table 2.1: Rank 1 match scores (in %) of our system, compared to those reported in the literature. The middle column shows the scores for the best performing system Wiskott et al. (1997) of the official FERET evaluation, and in brackets the average score of all 13 systems evaluated. The next column shows the performance of the two systems (SOM-Face/LocPb) proposed in Tan et al. (2005). The probe sets from the FERET database are the same as those of Figure 2.14, while for the AR database, the three emotion sets (b,c,d and h,i,j, respectively) and the two types of occlusion (e,f and k,l) have been averaged.

Table 2.1 shows the performance of our system considering only the first match (i.e. cumulative match score for rank 1), and compares it to the recognition rates of the systems evaluated in Phillips et al. (2000), and to the performance of Tan et al. (2005). We can see that our system outperforms the average of the systems tested in the FERET evaluation, but does not reach the performance of the winner of this evaluation. Similarly, performance on the AR database is poorer than that of the better approach proposed in Tan et al. (2005).

We can conclude that while our system is definitely competitive, its performance does not reach that of top-notch, purely functionally motivated face recognition systems. However, we did not apply any parameter tuning to the system as tested here. For example, it turns out that performance of the model grows monotonously with input grid size, with our resolution of 20x20 points still being far from saturation. In fact, the winner of the FERET evaluation Wiskott et al. (1997) uses Gabor wavelets from every pixel of the input image! Other parameters that could be optimized include the relative contribution of different landmarks, or the strength of topological interaction among the control units.

2.5.4 Attention Experiments

Attention and priming have a great influence on the way visual information is processed. Vision in animals and humans is not a static, reproducible process, but it depends on the current state of the organism. While this causes frustration with physiologists and experimental psychologists—who have great trouble constraining laboratory settings enough to reproduce their results—it is the basis of flexible, situation-specific behavior. For example, if we are looking for or expecting to see a certain object, we react faster and more accurately when the object finally comes into sight than if we were just watching passively (Duncan 1984). This is called *semantic priming* or *object-based attention*. Similarly, when we focus our attention to a certain area in the visual field, information from this region is processed preferentially. This is called *spatial attention*. Some cases of visual information processing even require attention to work at all, for example serial search (for differences between the serial and parallel modes of visual search, see, e.g., Nakayama and Silverman, 1986 or Treisman and Sato, 1990) or detection of even large changes in images (*change blindness*, see, e.g. Simons and Rensink 2005).

Both spatial attention and object-based attention effects can be realized in our system. Preactivating a subset of the control units at the beginning of a ν cycle (equivalent to spatial attention) results in a bias in favor of a specific location. For a large input image containing several faces, the system then preferentially processes and recognizes objects at that position (see Figure 2.16). Note that this way of preactivating pathways instead of feature activity merely biases the system towards a certain decision without distorting the content that is being processed. This, however, must be expected in the model of Deco and Rolls (2004), which implements spatial attention by increasing activity at the input level. Furthermore, experimental findings (Luck et al. 1997) suggest that attention shifts effective receptive fields (that is, pathways) without changing activity within the afferent layer.

Similarly, the priming of objects is possible with the help of preactivating search images in the form of arbitrary combinations of facial features in the Gallery Layer, leading to preferential detection and recognition of a similar face among several others in the visual field (see Figure 2.17). In distinction, preactivation of object-representing nodes in feature-based systems would not generate an explicit search image at lower layers but a large unstructured activation of *all* features that potentially could give rise to the primed object. It is quite unlikely that this would result in useful object priming, at least not in a field like face recognition, where the target object is not defined by a pure enumeration of its features, but rather by its specific arrangement.

Instead of priming a specific face, it is also possible to bias a large group of faces representing a certain type of face, for example a “female” face. When all faces of women are preactivated in the Gallery, the initial activity in the Gallery Assembly changes (see Figure 2.18 from a neutral average face to a prototypical female face (prototypical for the database being used, that is). Figure 2.19 shows how the system, having been primed on female faces, has chosen and recognized one of the two women present in the input image (which one it chooses is not specified by us in this case, so it chooses the one that has more similarity with the average female face).

2.6 Discussion

The model presented in this chapter is fully neural and combines findings from psychophysics, imaging studies, and physiology, while still performing competitively on benchmark tests for face recognition. The basic building block of the system is a model of the cortical column that makes use of a population code for stimulus representation, thus allowing significantly faster computations than with rate codes of single neurons (see, e.g., van Vreeswijk and Sompolinsky 1998). An essential ingredient of the model are dynamic links, synaptic connections that are modulated by the activity of control units, whose activity in turn is controlled by signal com-

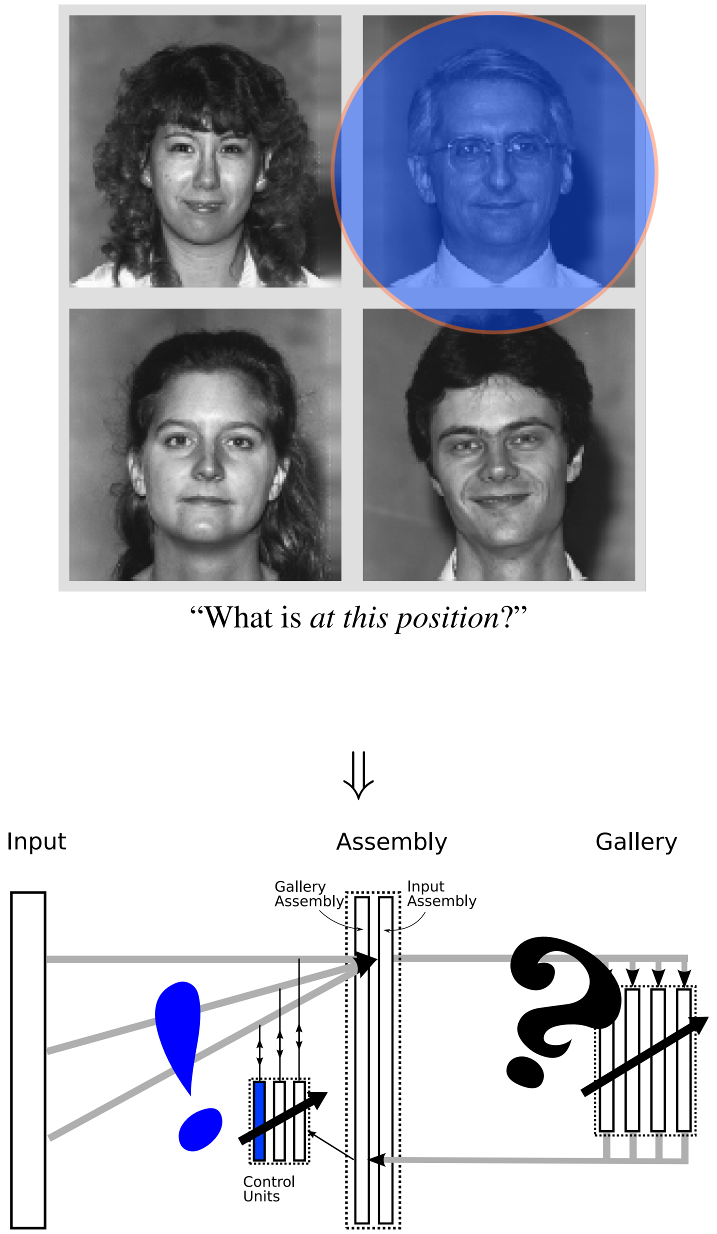


Figure 2.16: Spatial attention. The system's attention is guided to a certain position in the large input image (visualized by the circle around one of the faces) by preactivating the control units pointing to the respective area of the Input Layer. This weak preactivation is enough to bias the control units into matching the Assembly Layer with the attended face, and consequently, the person behind that face is recognized. Applying no attention results in a competition between the different input faces that can take several ν cycles, until the most salient face present in the Gallery is selected and recognized. In that sense, spatial attention speeds up processing.

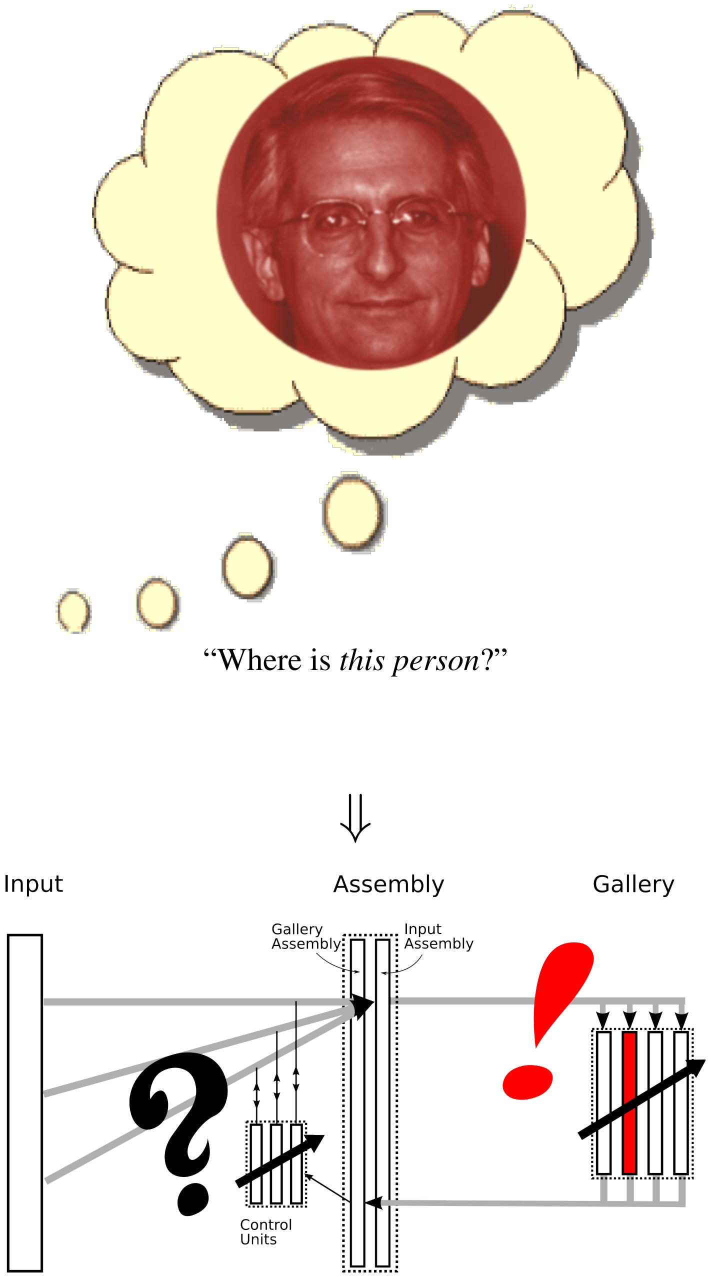


Figure 2.17: Object search. The system can be primed for a specific face by preactivating its respective Gallery units. This leads to the system searching the input image for a similar face and selecting and recognizing this face. If the face is not present in the input image, the system selects the most similar input object and recognizes it (which can result in a different set of Gallery units winning than originally preactivated).

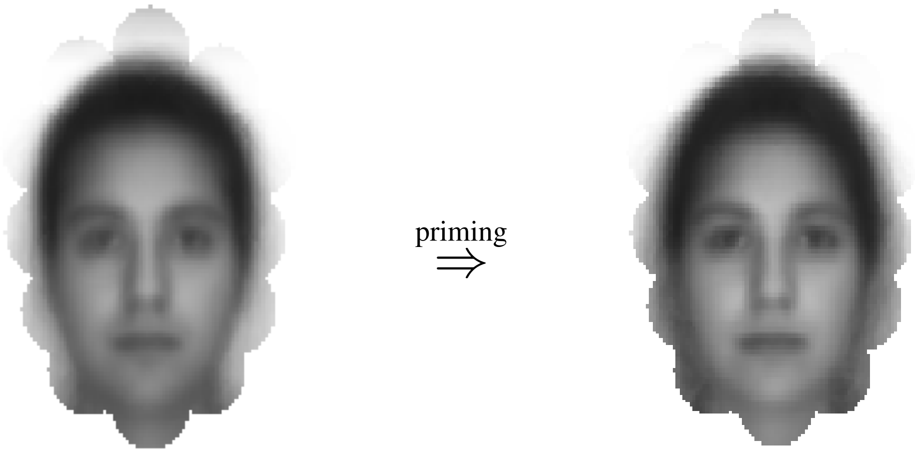


Figure 2.18: Activity of the Gallery Assembly after priming of female faces. When the Gallery units representing female faces are preactivated, the visual information present in the Gallery Assembly changes from the initial neutral average face to an average female face, before then becoming more dedicated in the subsequent recognition process.

parisons. There is strong experimental evidence that receptive fields of neurons are not static (see (Luck et al. 1997) and other references in the Introduction), suggesting the existence of dynamic links. For a discussion of further evidence for dynamic routing of information in the brain, see Section 3.2. Other models in the literature argue for similar concepts, like the control units of (Olshausen et al. 1993) or Sigma-Pi neurons (Weber and Wermter 2007). For possible physiological mechanisms of modulation of neural signals, see Section 3.4.2.

In anatomical terms, the different layers of our model can be interpreted as follows. The input layer represents incoming image information by Gabor wavelets, which resemble the receptive field properties in primary visual cortex (V1). The biological counterpart of our Assembly Layer would be an area like central or anterior inferotemporal cortex. Neurons here respond to stimuli from large parts of the visual field, and they code for complex shapes similar to the face parts represented by the Assembly Layer (Tanaka 1996, 2003). The fact that information about object position and scale can be read out from IT neurons (Hung et al. 2005), which disagrees with the assumptions made by pure pooling-models, points to the possibility of our control units residing there as well. Of course, in the cortex the mapping from V1 to IT does not happen directly, but via intermediate stages like V2 and V4. This is not accounted for in our current model, but will be included in future extensions. We have described previously how such a routing over several stages should look like (Wolfrum and von der Malsburg 2007b) and how it can develop ontogenetically (Wolfrum and von der Malsburg 2007a). Finally, the Gallery of our model might correspond to an area like the fusiform face area (FFA), which is specialized for face recognition (Kanwisher and Yovel 2006, Tsao et al. 2006). Note that the decision *that* something is a face is not modeled by us. Also in the brain, this appears to happen outside of FFA. Summerfield et al. (2006) find neurons in medial frontal cortex that are selectively active

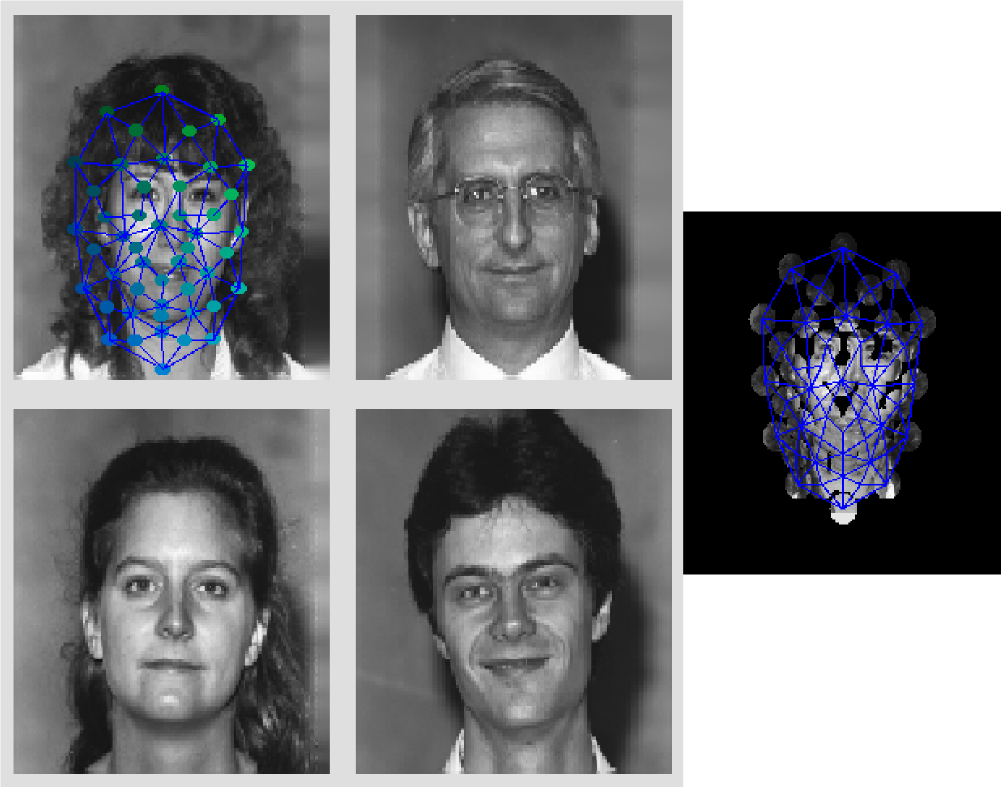


Figure 2.19: Final result of a priming experiment. The Gallery units representing female faces were preactivated. This biases the system towards detecting female faces. The matching process locks onto that part of the input image that resembles most the female average face represented by the Gallery Assembly, and subsequently recognizes this person. The face graph landmarks in the input image (left) are positioned at the population averages of the active control units, indicating the correspondence the system has found. The right image shows the information present in the Input Assembly.

when subjects have to make a face vs. non-face decision, independently of face *identification*. Likewise, prosopagnosia patients recognize objects as faces but cannot identify them (Zhao et al. 2003). As discussed before, face recognition is special because faces have a generic shape, but recognition from thousands of individuals requires high sensitivity to detailed differences. This might become possible through competitive interaction, which in fact is the mechanism by which recognition happens in our Gallery Layer, in the small and compact FFA (Kanwisher 2006). Apart from faces, there is evidence suggesting that FFA can also serve as an area of expertise (Tarr and Gauthier 2000, Gauthier et al. 2000) for other object classes. In the same sense, our model is not confined to face recognition, but could be used for recognition of any kind of object type that has a prototypical shape and requires high sensitivity to small differences among objects.

Our system is generative in the sense that it explicitly represents both the recognized object and the extrinsic properties to which it is invariant (especially object position). This has several consequences. First, the model requires top-down weights to re-generate the input, and these weights are identical to the bottom-up weights through which the Gallery units are activated. This existence of (not necessarily identical) weights in both directions is typical for generative models. The influential model of Olshausen and Field (1997) for learning V1 receptive fields makes use of both “analysis” and “synthesis functions”, the Helmholtz machine (Dayan et al. 1995) has distinct “recognition” and “generative weights”, and also current models for object recognition like (Murray and Kreutz-Delgado 2007) use explicit bottom-up and top-down weights. It is hard to imagine a generative model without this dual representation. Let me note that this aspect of our model reflects anatomical reality, with cortical top-down connections being at least as strong if not stronger than feedforward connections. The top-down pathways that are usually used to reconstruct the current percept can also send down information coding for an *expected* percept. In that way, the system can be prepared for certain objects or locations, replicating typical attentional and priming effects. We explored these capabilities in Section 2.5.4.

Finally, object representations in generative models are *explicit*. A cardinal cell of a typical feature-based model represents objects or properties implicitly, as the activity of the cell does not express the structure of the object, which is only implicit in the synaptic patterns that define its firing condition. An explicit representation, on the other hand, cannot only detect presence of the object, but can also reproduce the appearance of the object under many different conditions, and may, e.g., relay this rich information to recipients elsewhere in the brain. An explicit representation therefore forms not just a sample point in appearance space but represents a whole space of variations. However, in distinction to the general, all-purpose representation in the retina, the ultimate goal of the visual system is to lead to explicit representations that encode the meaningful parts of a scene, like object identities and their properties, in a combinatorial and invariant code that is useful for subsequent action planning or motor control. It is these explicit representations that give us the feeling of being in direct contact with the visual reality out there.

Where does our model stand in the light of these distinctions? The representation in the Input Layer is explicit and uncommitted, and its generality is restricted only by the incomplete sampling of Gabor space. The representation in individual units in the Gallery Layer is implicit, highly specific and completely committed (to individual landmarks in individual faces). Given its activity state, the Gallery Layer creates, via its output connections to the Gallery Assembly, an explicit representation of a face, which for low levels of the inhibition is still uncommitted to an individual (see Figure 2.11, upper-most image), and at the end of the selection process corresponds to an explicit representation fully committed to one individual face. While models of object recognition that explicitly reconstruct the input are common in probabilistic modeling (conceptually discussed e.g. in Yuille and Kersten 2006), it is still unclear, how such reconstructions are realized by the brain. Hopefully this work can contribute to deepening our insight in this respect.

Surely, the model still leaves open a number of problems for future work. As is, the model is invariant only to translation and needs to be generalized to changing scale and orientation (which will require dynamic relinking of feature connections, see Sato et al. 2007, 2008) as well as other image transformations such as changing illumination and perspective deformation. Another important extension of the system is autonomous learning of the contents of the Gallery.

An unrealistic aspect of the model as presented here are the direct dynamic links from all positions in the Input Layer to all positions of the Input Assembly. This would require unrealistic number of fibers converging on any target unit. Therefore, we have investigated how the fiber convergence numbers and the overall number of neural elements required for correspondence finding can be reduced by distributing the process over an architecture of several stages. We turn to this topic in the next chapter.

3 Switchyards—Routing Structures in the Brain

In the previous chapter, we discussed a mechanism for correspondence finding and recognition. For simplicity, we assumed a given, rather simple all-to-all connectivity for correspondence finding between Input and Assembly Layer. In this chapter, we will investigate in detail how such a connectivity structure should look like to be biologically more realistic.

We first introduce the idea of multistage routing (Section 3.1) and discuss physiological evidence for it (Section 3.2). We proceed to derive an optimized architecture for routing in Section 3.3. We have termed these optimal architectures *switchyards*, referring to the networks of tracks and switches that are used to route and reassemble railway trains. In Section 3.4, the resulting networks are discussed in functional and physiological terms, before the chapter concludes with Section 3.5. Parts of the material presented in this chapter has been published in (Wolfrum and von der Malsburg 2007b).

3.1 Multi-Stage Routing

As discussed in Chapter 1, an important capability of biological vision systems is invariant object recognition. The same object seen at different position, distance, or under rotation leads to entirely different retinal images which have to be perceived as the same object. Only one of these variances, translation, can be compensated by movements of the eye. The approximate log-polar transform which takes place in the mapping from retina to cortex (Schwartz 1977), on the other hand, replaces some kinds of transformations (scale, rotation) by others (translation on the cortex) and therefore does not fully explain invariant recognition, either. Invariant recognition, and how the visual system performs it, remains a topic far from being understood.

Invariance does *not* mean insensitivity to the spatial arrangement of visual information. While object recognition in our brain is invariant with respect to the above-mentioned transformations, it is very sensitive to small differences in the retinal activity pattern arising from, say, seeing both of your twin sisters shortly after each other. We believe that the only way a brain can solve these two competing problems realistically is to have a general, object-independent mechanism that compensates variance transformations without distorting image information. Thus the image can be conveyed in a normalized frame of reference to higher brain areas for recognition. Such a mechanism requires a routing network providing physical connections between all locations in the visual input region (V1) to all points in the target area (like IT). In addition, neural machinery is required to control these connections.

The necessity of dynamic information routing was appreciated early on in vision research (Pitts and McCulloch 1947). Especially the finding that in posterior parietal cortex a frame of reference transformation takes place from retinal coordinates in primary visual areas to head-centered coordinates in higher areas (e.g., cf. Duhamel et al. 1997) has received a lot of modeling attention (e.g. Pouget and Sejnowski 1997, Zipser and Andersen 1988). For object recognition, on the other hand, the idea of dynamic routing is much less common, maybe because here its importance is not as evident as for coordinate transforms. Nevertheless, there have been models for routing in object recognition. In a sense, correspondence-based models for recognition like the one discussed in Chapter 2 dynamically route visual information, alas in a very basic way, by connecting input and memory domains in an all-to-all manner. This, however, is biologically not very plausible. To allow correspondence finding at acceptable resolution in the central parts of the visual field alone would require each point of the input to be connected to $\approx 100\,000$ points in the memory domain. This contradicts the fact that the number of inputs to typical neurons is rather on the order of $1\,000$ (Cherniak 1990). Therefore, architectures have been proposed that distribute the process of visual information routing over several stages. These include the famous Shifter Circuits of Olshausen et al. (1993) or the SCAN model (Postma et al. 1997).

3.2 Physiological Background of Dynamic Routing

In spite of their essential advantages, systems that employ dynamic information routing are not as widespread in the modeling community as traditional neural networks with static connections. One reason for this may be that they require certain functional elements that static neural nets do not have. These are

1. a gating mechanism that allows neural signals to be modulated by the activity of other neurons and
2. neural units that perform this task of modulating other activities.

Although maybe more challenging to implement technically, networks with modulatory interactions between neurons are vastly more powerful than traditional neural networks (Durbin and Rumelhart 1989, Koch 1999). Moreover, there are plausible physiological explanations of how this additional functionality may be realized in the brain, as we will discuss in the following.

A gating mechanism requires that two neurons can *multiplicatively* influence the activity of a target neuron. Or, in other words, the activity of one neuron *modulates* the information transmission from another neuron to a target. Experimental evidence for such multiplicative or modulatory interactions between neurons abounds. Gabbiani et al. (2002), e.g., have verified that specific neurons of locusts perform a multiplication of two input signals. A very prominent example in higher animals are the dopaminergic modulations of cortical input to the striatum (e.g., Freund et al. 1985, Nicola et al. 2000). Functionally, multiplicative interaction can be achieved by the input function of the target neuron having the general form $\sum_i \sum_j w_{ij} x_i x_j$, x representing the activities of the afferent neurons. Due to the subsequent multiplication and summing of inputs, such neurons are called *Sigma-Pi* neurons (e.g., Durbin and Rumelhart 1989). Physiologically, such multiplication of inputs can be implemented by shunting inhibition at the same synapse (Volgushev et al. 1996, Kubota et al. 2007) or by supra-linear interaction of apical and distal inputs (Larkum et al. 1999, Schaefer et al. 2003) to a neuron. Another possibility for multiplication of neural signals is a nonlinear *activation* function. Tal and Schwartz (1997) note the fact that LIF neurons transform synaptic inputs approximately logarithmically into firing rates. Adding the *outputs* of two LIF neurons therefore gives the logarithm of the products of their inputs. For a wider review of cerebral gating mechanisms, refer to, e.g., (Salinas and Sejnowski 2001).

The other necessary ingredient for routing networks are units that control the information flow through the routing network. Similarly to the control units of Chapter 2, they have to compare visual signals in the input and the memory domains, as well as incorporate the decisions of neighboring control units. In a multi-stage routing network, they additionally have to factor in the activity at intermediate stages. These requirements are met by the pulvinar nucleus in the thalamus, which is reciprocally connected to all stages of the ventral stream. Lesion studies in the pulvinar (Desimone et al. 1990) suggest that it is involved in directing visual attention, which would be a typical task of control units. On a similar note, the importance of the dorsal stream for processing the position of objects in the visual field suggests that control units might be located somewhere in that area and interact with the ventral stream from there. However, another, and maybe simpler explanation is that control units reside in the ventral stream itself, mixed with those units that represent and process feature information. This would make sense since it makes integration of information from and feedback of routing commands to the ventral stream much easier. While this idea contradicts the textbook view that the dorsal stream takes care of object position and the ventral stream is only responsible for object identity, there is growing evidence that these roles are not as separate as previously thought. Hung et al. (2005) have found that object position and scale can be read out from neurons in inferotemporal cortex (a high area of the ventral stream), while Konen and Kastner (2008) report representations of object identity in the dorsal stream.

Evidence that points specifically to the existence of multi-stage routing circuits as defined below is that the tangential spread of the basal dendrites and the number of dendritic branches and spines of pyramidal neurons increase along the ventral pathway (Lund et al. 1993, Elston and Rosa 1998, 2000). This fits well with architectures like the one of Figure 3.1. For discussions of evidence for dynamic information routing in general, see Sects. 1.2 and 2.6.

3.3 Optimized Architectures for Routing

What has been missing in previous models of multi-stage routing is the consideration of efficiency in terms of required neural resources. Different routing architectures require different numbers of intermediate feature-representing nodes (we will refer to them simply as *nodes*) and node-to-node connections (*links* from now on; if we mean both links and nodes, we will use the term *units*). Since we will not discuss here *how* connections in a routing circuit are controlled, we simply assume that the maintenance of a link and its control by a neural control unit have the same cost as feature nodes (for a deviation from this assumption, see the second part of Section 3.3.1). It is likely that cortical architectures have evolved which minimize this cost for the organism. Below, we therefore derive and analyze the routing network structure that minimizes the sum of all required units, both nodes and links. We call these optimal structures *switchyards* in reference to the networks of tracks and switches that are used to route and reassemble railway trains.

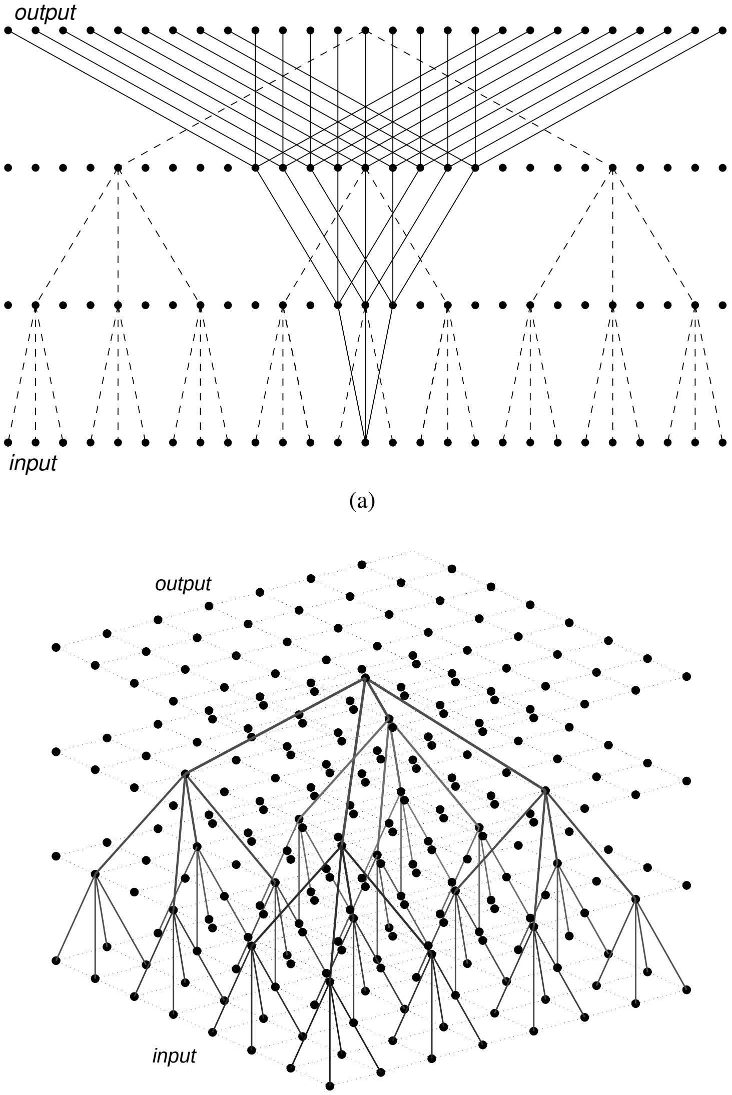
In our analysis we will focus on two situations. In Section 3.3.1 we discuss routing between two cortical regions of identical size. This corresponds to perception of an already coarsely segmented object. In Section 3.3.2 we consider the architecture that must be present in real biological vision systems: Routing from a large input domain to a much smaller output domain, the first corresponding to the whole visual field, the second to a small higher-level target area engaged in object recognition.

3.3.1 Routing Between Two Regions of the Same Size

Let us define a routing architecture with as few assumptions as possible:

- Input and output stages both consist of n image points. Each image point is represented by one feature unit (the extension of this to more than one feature per image point will be discussed below).
- The routing between input and output is established via $k-1$ intermediate layers of n feature units each. This number of layers is not predefined but will serve as the optimization parameter for minimizing required neural circuitry.
- Nodes of adjacent feature layers can be connected. For every such connection there exists one dynamic, neural unit that controls information flow in both directions. These units resemble the *control units* of (Olshausen et al. 1993). We assume here that one link (including its control unit) imposes the same "maintenance cost" as one feature node. If these costs are not identical, this can be accounted for with the parameter α introduced below.

Under these assumptions, what is the minimal architecture providing for each input node one separate pathway to every output node? For $k = 1$, the situation is clear: without any intermediate layer, every input node must be connected to all n output nodes. With intermediate layers, however, we can make use of a combinatorial code to achieve full connectivity, similar to "butterfly" computations used in the fast Fourier transform (Cooley and Connor 1965): we assume that each input unit is only connected to l nodes of the adjacent intermediate layer (see solid lines in Figure 3.1a). Each of these l nodes has in turn connections to l nodes of the following layer, and so on, until the output stage is reached. This method yields for every input node l^k pathways to the output stage, which are unique and lead all to different output nodes *if* we make sure that no two separate pathways merge again on the way to the output stage. An anatomically plausible way to meet this functional requirement is to let the spacing between target points increase exactly by the factor l from one link layer to the next, as shown in Figure 3.1a. The two-dimensional case is analogous, except that here the groups of nodes projecting to the same target are two-dimensional patches of l units with adequate spacing in between (see Figure 3.1b).



(b)

Figure 3.1: Architectures for routing networks. (a) The one dimensional case, with $n = 27$ and $k = 3$, thus $l = n^{\frac{1}{k}} = 3$. All feature nodes are shown (dots), but only selected links (lines), the others being shifted versions (with circular boundary conditions) of the shown links. All connections from one input node to the whole output stage are shown as solid lines, the connectivity between one output node and the whole input as dashed lines. (b) The two dimensional case, with $n = 64$, $k = 3$, and $l = 4$. Only the downward connections from a single output node are shown.

The connectivity described here agrees with the anatomical finding that loosely speaking, the spread of neuronal connections increases along the visual hierarchy. Perkel et al. (1986), for example, found a higher divergence of direct projections between V1 and V4 than between V1 and V2 or V3, respectively. In (Tanigawa et al. 2005), a four times larger spread of horizontal axons in inferotemporal cortex than in V1 was reported. Note, however, that the specific connectivity of the routing network is irrelevant for the results derived in the following. The only requirement is that the pathways of every input node be unique and lead to different output nodes.

In order to reach the whole output stage with these pathways, their number must equal the number of output nodes:

$$n = l^k.$$

From this we get the necessary neuronal fan-out at each stage as

$$l = n^{\frac{1}{k}}. \tag{3.1}$$

Let us now calculate how many nodes are needed to realize the routing architecture. Having $k-1$ intermediate layers means that a total of $(k+1)n$ feature nodes is required. All of these nodes, except those of the output layer, have l links to the next stage, resulting in a total of $knl = kn^{\frac{k+1}{k}}$ links. So the total number of units as a function of k and n is

$$N(k,n) = (k+1)n + kn^{\frac{k+1}{k}}. \tag{3.2}$$

As we can see in Figure 3.2, this number changes drastically with the number of intermediate layers being used. A direct all-to-all connectivity without any intermediate layers ($k = 1$) is most expensive because the number of required links scales quadratically with n in this case. For a very large number of intermediate layers, on the other hand, the decrease in the required fan-out l is outweighed by the linear increase in nodes caused by additional layers. As we can see, there is a unique value k_{opt} for which the number of required units attains a minimum. To determine k_{opt}, we calculate the derivative of N with respect to k and set it to zero:

$$\frac{\partial N}{\partial k} = n + n^{\frac{k+1}{k}} - k \, \ln n \frac{1}{k^2} n^{\frac{k+1}{k}} = n\left(1 + n^{\frac{1}{k}} - n^{\frac{1}{k}} \frac{\ln n}{k}\right) \overset{!}{=} 0.$$

This is satisfied for

$$\frac{\ln n}{k} = n^{-\frac{1}{k}} + 1. \tag{3.3}$$

With the ansatz

$$k_{opt} = c \, \ln n \tag{3.4}$$

(3.3) becomes independent of n:

$$\frac{1}{c} = e^{-\frac{1}{c}} + 1. \tag{3.5}$$

Solving this numerically we obtain

$$k_{\text{opt}} \approx 0.7822 \, \ln n. \tag{3.6}$$

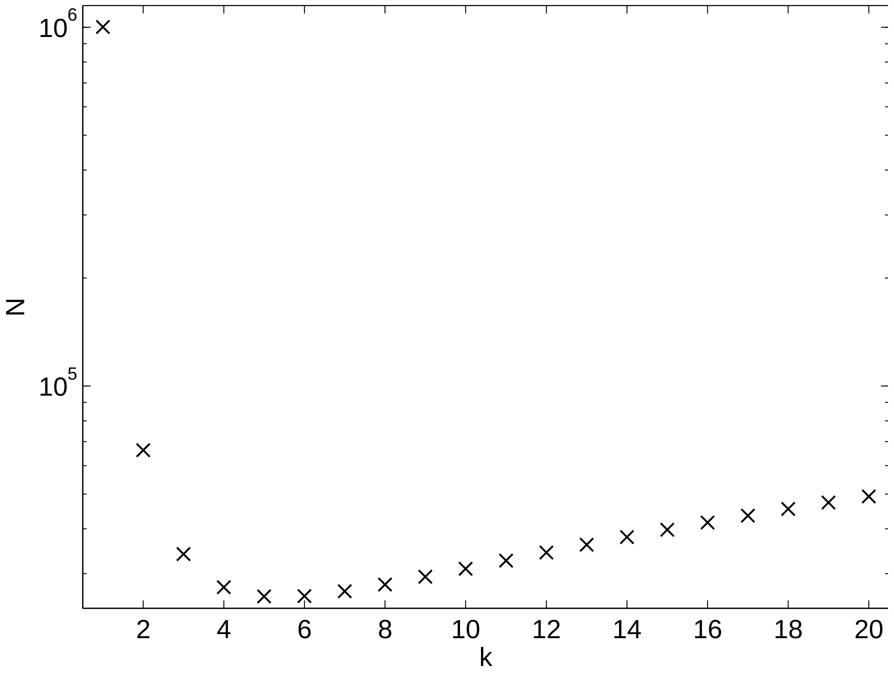


Figure 3.2: The number of required units for a routing architecture between two layers depends strongly on the number $k-1$ of intermediate layers being used. The values shown here are for input and output stages of $n = 1000$ image points each.

The fact that k_{opt} scales logarithmically with n is not surprising by itself. Such a scaling behavior lies at the heart of many techniques that have to permute or operate on a group of nodes simultaneously, like permutation networks or the fast Fourier transform (Cooley and Connor 1965). Even in random graphs Erdös and Rényi (1959) the network diameter (corresponding somewhat to our number of layers k) scales logarithmically with the number of nodes. This general logarithmic scaling behavior is independent of the specific fanout (or degree) at each node. A different fanout only changes the basis of the logarithm, which is equivalent to changing the prefactor in the logarithmic relation. Here, however, minimizing the number of components of the network leads to a *specific* logarithmic scaling, or phrased differently: the prefactor c in k_{opt} is unique. This goes hand in hand with the existence of a unique optimal fanout

$$l_{\text{opt}} = n^{\frac{1}{k_{\text{opt}}}} = e^{\frac{1}{c}}. \tag{3.7}$$

We will discuss this finding further in Section 3.5.

More than one feature per image point

So far, we have neglected the routing of visual information when there are several feature cells at one image point. Instead of a dense pixel array, visual information in V1 is represented by a pattern of “hypercolumns” of lower density. However, each hypercolumn contains cells responsive to many different local properties of the input, such as wavelet-like features (Gabors) at different orientations and spatial frequencies, different colors or specificity for one eye or the other.

It may not be necessary to route these features independently of each other to higher areas, so one might assume that only one active link is needed to route many feature units in one image location. On the other hand, certain feature types do require individual treatment. For example, for full orientation invariance, units of one orientation specificity of the input would need connections to all orientation specificities of the output domain. The truth probably lies somewhere in between these two extremes, as suggested in (Zhu and von der Malsburg 2004): image points are not routed individually, but in small assemblies through collective links called “maplets”. For every group of nodes, there exist several such maplets, responsible for routing at different scales and orientations without requiring individual links for all features in all positions.

Since the focus of this analysis is not on a specific routing architecture, but on finding the optimal number of layers for a very general architecture, we will merge the above arguments into a single factor $\alpha \geq 1$ representing the number of feature nodes that are controlled by a single link. If necessary, the parameter α can also be used to account for unequal expense assumed for feature units versus link units.

Instead of n independent feature nodes, we now have

$$n_\alpha = \frac{n}{\alpha} \tag{3.8}$$

groups of nodes, each containing α nodes. With this, the number of units in the routing circuit (3.2) changes to

$$N(k,n) = \alpha(k+1)n_\alpha + k n_\alpha^{\frac{k+1}{k}}. \tag{3.2'}$$

Setting the derivative of (3.2') to zero leads to

$$\frac{\ln n_\alpha}{k} = \alpha n_\alpha^{-\frac{1}{k}} + 1 = 0. \tag{3.3'}$$

The new ansatz

$$k_{opt} = c \ln n_\alpha \tag{3.4'}$$

yields

$$\frac{1}{c} = \alpha e^{-\frac{1}{c}} + 1, \tag{3.5'}$$

which we can solve numerically for explicit values of α. In Figure 3.3 we see that c—and with it k_{opt}—only changes by a factor of 2 over a reasonably large range of α.

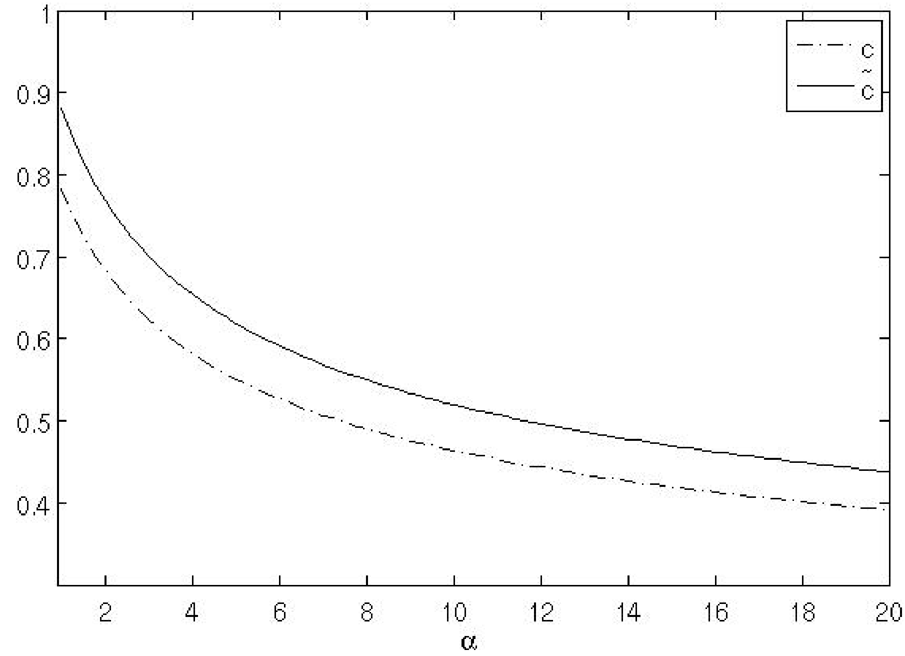


Figure 3.3: The prefactors c and $\tilde{c}$ define k_{opt} through (3.4') and (3.13) in the cases of routing to an output of the same size or much smaller size, respectively.

Having determined the number of layers k_{opt} that minimizes the required neural circuitry for given n and α, we can calculate the size N_{opt} of this minimal circuitry. Inserting (3.4') into (3.2') yields

$$N_{opt}(n) = n + \left(\alpha + e^{\frac{1}{c}}\right)(c n_\alpha \ln n_\alpha). \tag{3.9}$$

This means that for large n the number of units of the optimal routing architecture between two layers of n_α image points scales with

$$N_{opt}(n_\alpha) \propto n_\alpha \ln n_\alpha, \tag{3.10}$$

as expected from classical network theory. This result holds also for routing of only a single feature ($\alpha = 1$) per point.

3.3.2 Routing Circuit with Different Sizes of Input and Output Layer

Let us now discuss routing from the whole visual field to a comparatively small cortical output region. We assume that an attentional mechanism singles out, in the input domain, a region that is to be mapped to the output region.

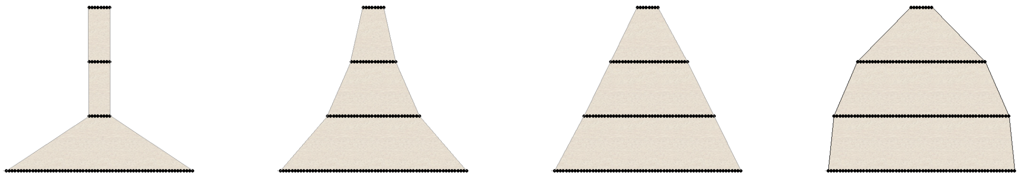


Figure 3.4: Possible forms of tapered networks. All these ways of decreasing layer size towards the output are possible in principle. For our analysis we choose a linear decrease (center) as the most parsimonious solution.

We do not claim here that invariant recognition is perfect over the whole visual field (there are studies showing that this is not the case, like Dill and Fahle, Cox et al., 1998, 2005), but object recognition *is* possible to some degree even at high retinal eccentricities, although of course impaired by the poor resolution at these parts of the retina. In any case, the basic problem remains the same as before: neural connections must exist between all parts of the visual input region and a target area.

Computationally, the situation is very similar to the one discussed in the previous section and leads to a generalization of the results derived there. We now want to connect an input stage of n units with an output stage that is smaller by the factor m and contains only $\frac{n}{m}$ units. As before, the routing is established via $k-1$ intermediate layers, and groups of α nodes can be routed collectively. With the same argument as in Section 3.3.1 we see that now each group has to make

$$l = \left(\frac{n_\alpha}{m}\right)^{\frac{1}{k}} \tag{3.11}$$

connections to the next higher layer in order to connect every group of input nodes with every output group. Figure 3.5 shows parts of the architecture required for routing from a 125 node input to an 8 node output stage. Note that due to the different input and output sizes downward fan-out now has to be higher than the upward fan-out l.

Differently from before, the size of intermediate layers is not well-defined now. In principle an architecture is conceivable where already after the first routing stage the size of the feature layers is reduced to that of the output stage (Figure 3.4, left). Although this still allows all input units to be connected to all output points, it would require a comparatively high number of downward connections for the feature units in the first intermediate layer. Moreover, this decrease of layer sizes clearly contradicts anatomical reality. Other possibilities include geometric

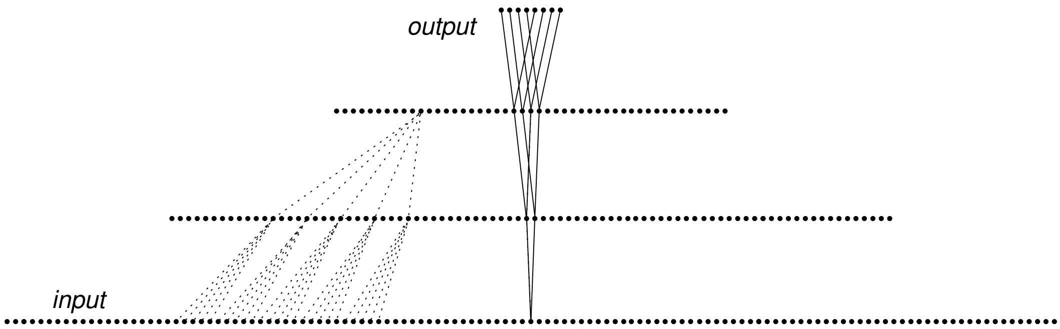


Figure 3.5: Routing network for an input of $n = 125$ and an output of $\frac{n}{m} = 8$ nodes with $k = 3$ link layers and linear decrease of layer size. Consequently, the upward fan-out is $l = 2$. The full lines show the links connecting an input node with the full output stage. Downward connectivity is $l_{down} = n^{\frac{1}{3}} = 5 > l$ (shown exemplarily for a node on the second level of the architecture by dotted lines).

or linear decrease of intermediate layers (Figure 3.4, center) or the sub-linear tapering on the right side, which was used in (Olshausen et al. 1993). We will assume here that the number of nodes changes linearly from the input to the output layer. This is supported by measurements of the average sizes of primary visual areas in humans (Dougherty et al. 2003). Note, however, that the same paper reports variance of V1 sizes of more than 100% between different individuals. In general there seems to be little undisputed data on this question in the literature. Given this uncertainty in the anatomical data, the simplest possible assumption is probably best for this kind of general discussion.

With a linear decrease in size, the number of feature units in layer κ ($\kappa = 0$ for the input and $\kappa = k$ for the output layer) is

$$f_\kappa = n - \frac{\kappa}{k}\left(n - \frac{n}{m}\right).$$

The number F of the feature encoding units of all layers is then

$$F = \sum_{\kappa=0}^{k} f_\kappa = (k+1)n - \left(n - \frac{n}{m}\right)\frac{k(k+1)}{2k} = \frac{n}{2}\frac{m+1}{m}(k+1).$$

Adding the links emanating upwards from all but the top-most layer, we get the total number of units as

$$\begin{aligned} N(n,k) &= F + \frac{1}{\alpha}\left(F - \frac{n}{m}\right) l \\ &= \frac{n_\alpha}{2}\frac{m+1}{m}\left[\alpha(k+1) + \left(k + 1 - \frac{2}{m+1}\right)\left(\frac{n_\alpha}{m}\right)^{\frac{1}{k}}\right]. \end{aligned} \tag{3.12}$$

Setting the derivate with respect to k to zero leads to

$$-\alpha\left(\frac{n_\alpha}{m}\right)^{-\frac{1}{k}} = 1 + \frac{\ln\frac{n_\alpha}{m}}{k^2}\left(\frac{2}{m+1} - k - 1\right).$$

With the ansatz

$$k_{opt} = \tilde{c} \ln \frac{n_\alpha}{m} \tag{3.13}$$

this turns into

$$-\alpha e^{-\frac{1}{\tilde{c}}} = 1 + \frac{1}{\tilde{c}^2 \ln \frac{n_\alpha}{m}} \left(\frac{2}{m+1} - 1 \right) - \frac{1}{\tilde{c}}.$$

For large input/output ratio m, the term $\frac{2}{m+1}$ becomes negligible, so that $\tilde{c}$ depends only on the number of independently routed output nodes $\frac{n_\alpha}{m}$ and not on m itself:

$$-\alpha e^{-\frac{1}{\tilde{c}}} \approx 1 - \frac{1}{\tilde{c}^2 \ln \frac{n_\alpha}{m}} - \frac{1}{\tilde{c}}. \tag{3.14}$$

Numerical analysis of (3.14) shows, however, that $\tilde{c}$ changes by less than 10% when $\frac{n_\alpha}{m}$ is varied over 3 orders of magnitude. So we can say that, like c in Section 3.3.1, $\tilde{c}$ only depends on the parameter α. Figure 3.3 shows that $\tilde{c}$ takes on similar but slightly higher values than c.

Calculating the size of the derived routing circuit by plugging (3.13) into (3.12) yields

$$N_{opt} = \left(\alpha + e^{\frac{1}{\tilde{c}}} \right) \frac{n_\alpha}{2} \left(\tilde{c} \ln \frac{n_\alpha}{m} + 1 \right) \tag{3.15}$$

for large m. Although the relation is a bit different from the one derived for equal input and output domains (3.9), the scaling with $n_\alpha \ln n_\alpha$ (since $n_\alpha \gg m$) remains the same.

3.4 Interpretation of Results

Let us now discuss functional and physiological implications of the architecture derived in Section 3.3.

3.4.1 Difference to Sorting Networks

It is important to note that the routing architectures derived here are different from sorting networks known in computer science. A sorting network takes a number of scalar input signals and outputs them at the final stages ordered by their values. At each intermediate node, (usually) two signals converge and are compared by a logic unit to be passed on in an ordered fashion. Switchyards, on the other hand, shift and reorganize signals by a scheme imposed from outside (via the control units), without internally comparing signals. The intermediate nodes ($\widehat{=}$ feature units) do not have any active role but are pure relay stations that combine any signals they receive and pass them on (this may be a reason to use max-like operations at the feature units in future models). Therefore, switchyards run the risk of having *conflicts* in their routing, which is impossible by definition for sorting networks. A conflict occurs whenever the signal at two or more feature nodes at one stage is sent to the same feature node of the next stage. Here, the signals are inevitably mixed and cannot be separated any more. Note, however, that a conflict is problematic only if it mixes two totally unrelated signals. In some cases, e.g. when downscaling visual information, “conflicts” are unavoidable and actually desirable, since in this case high resolution information of neighboring points needs to be averaged ($\widehat{=}$ mixed) into a single point.

The question in which situations conflicts arise in routing networks has not been investigated in depth so far. For switchyards, no conflicts occur if pure shift or rotation operations are carried out. Scalings produce conflicts exactly if the scaling factor is a multiple of the fanout l of the switchyard. Whether a conflict can occur between the paths emanating from two input nodes also depends on their proximity. Figure 3.6 shows the number of conflicts that can occur between two nodes, scaled by the absolute number of possible paths that can be taken. While for two directly neighboring input nodes about 24% of possible choices of output nodes will lead to conflicts somewhere along the path, this number decreases to 0 for very distant nodes if we discount the degenerate case of identical output nodes. More precisely speaking, paths emanating from input nodes with a distance

$$d \geq \frac{n}{l} \tag{3.16}$$

can never collide no matter what output nodes are chosen (assuming the output nodes are not identical). Most conflicts in switchyards will anyway arise from the paths of two neighboring input nodes merging already at the first routing stage. This is not very problematic since owing to the nature of visual information, it does not mix totally independent signals, but signals that are very related anyway. Thus the merging merely acts as a kind of low pass filter. This kind of low pass behavior is even necessary to perform correspondence finding in switchyards, as we will see in Chapter 5.

Although switchyards are well suited for routing of visual information, sorting networks outperform them in raw power of re-routing since they can realize any kind of input-output mapping. One reason for this lies in the aforementioned fact that their intermediate nodes are active logical units instead of passive relay nodes. Moreover, sorting networks require larger circuits for given input and output sizes. Our switchyards are of size $\mathcal{O}(n \log n)$ with a relatively small prefactor. Functional sorting networks, on the other hand, scale at least with $\mathcal{O}(n(\log n)^2)$, more common and easier to implement ones even with $\mathcal{O}(n^x)$ with $x = 1.5$ or higher (for an extensive treatment, see Knuth 1997). Ajtai et al. (1983) describe a sorting network of order $\mathcal{O}(n \log n)$. However, this network is not used in practice since the prefactor in the size is huge and the explicit formulation of quite complicated expander graphs would be required.

We conclude that sorting networks and switchyards are not directly comparable. While the former can re-arrange signals arbitrarily, they have to use active logic elements at all relay nodes and are relatively large. Switchyards, on the other hand, have limited routing capabilities which are somewhat tailored for visual information routing, and they only scale with $\mathcal{O}(n \log n)$.

3.4.2 Physiological Interpretation

In Section 3.3 we found that the optimal number of link layers in a routing circuit is given by

$$k_{opt} = c \, \ln n_\alpha$$

and

$$k_{opt} = \tilde{c} \ln \frac{n_\alpha}{m}$$

for routing to an output stage of identical size and of much smaller size, respectively. So in both cases, k_{opt} is proportional to the natural logarithm of the number of independently routed *output*

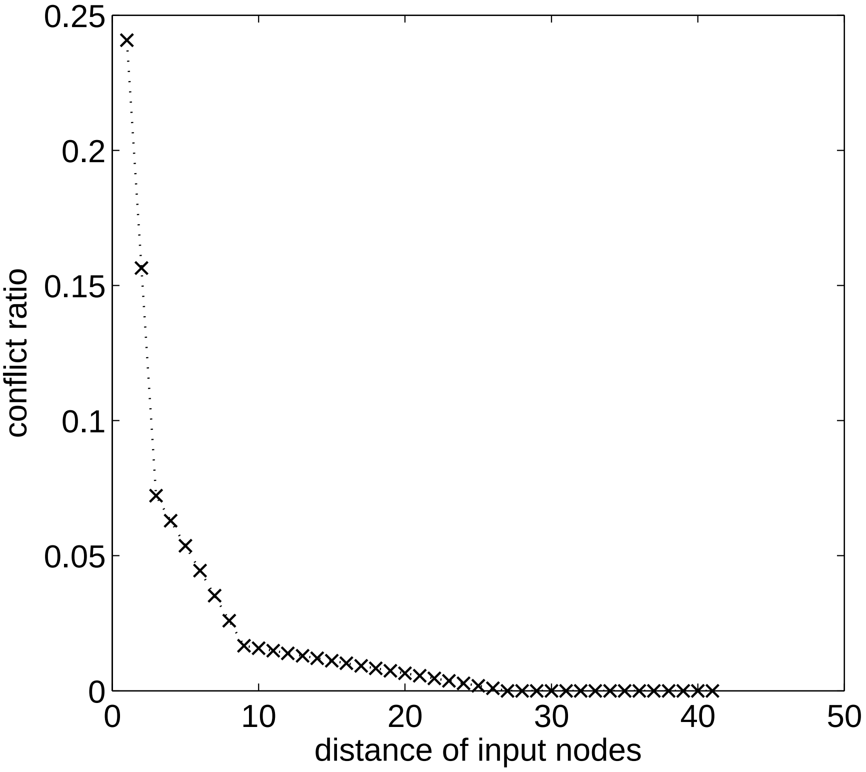


Figure 3.6: Number of possible conflicts as a function of distance of input nodes. Shown here for a network with $n = 81$ input and output nodes, fanout $l = 3$, and $k = 4$ routing stages. The abscissa marks the separation of two input nodes from which all possible pathways to arbitrary but different output nodes are evaluated. The absolute position of the input nodes is irrelevant since switchyards are symmetric to shifts. On the ordinate, we plot the number of pathway constellations that lead to conflicts, relative to the overall number of possible pathway combinations. For a distance larger or equal than $\frac{n}{l}$ $(= 27$ for this case), no conflicts can occur any more.

nodes. The well defined prefactors c and $\tilde{c}$ are very similar (cf. Figure 3.3), depend only on α, and do not vary too much over large ranges of α.

How do those results match the facts in the human brain? A good starting point is the optic nerve, which is known to contain $\sim 10^6$ fibers for humans and other primates (Potts et al. 1972). Since the optic nerve is the bandwidth bottleneck of the visual system, it is safe to assume that it contains no redundant information. The number of neurons in V1, however, is by far higher than the number of optic nerve fibers, mainly for two reasons. First, the cortex most probably employs a population coding strategy in order to reduce noise and increase transmission speed. This means that several neurons together (perhaps the ≈ 100 of a cortical minicolumn, cf. Section 2.3) represent one of our abstract feature units. Second, visual information is represented in an overcomplete code in V1 (Olshausen and Field 1997), increasing the number of feature units over the number of optic nerve fibers. Nevertheless, the information represented in V1 cannot be higher than that transported by the optic nerve, so that overcomplete groups of units can be routed collectively. We will therefore assume that the number of feature encoding units is of the same order as the number of fibers in the optic nerve, keeping in mind that an overcomplete basis in V1 may be accounted for by a correspondingly higher value of α.

From the primary visual area V1, visual information is routed retinotopically along the ventral pathway to a target region in inferotemporal cortex (IT). Psychophysical evidence (van Essen et al. 1991) suggests that about 1000 feature nodes are sufficient to represent the contents of the two dimensional "window of attention", and therefore the size of this target region, at any given time. One may assume that there exist multiple such target regions in parallel in IT, which are used for different object recognition tasks.

How would our routing architecture look for these numbers? For this, we still miss an estimate of the parameter α. Research in our lab has shown that representing an image with 40 Gabor wavelets in each image point preserves all necessary image information of gray scale images (Wundrich et al. 2004) and allows good object identification (Lades et al. 1993). To additionally include color and temporal information (direction of motion), this number would have to be roughly twice as high. This is in line with findings concerning the number of orientation pinwheels in the primate brain. (Obermayer and Blasdel 1997) report around 10^4 pinwheels for V1 of the Macaque. Assuming a similar number for the human brain, we face the situation of an input region of the ventral stream containing 10^6 feature units clustered in some 10^4 pinwheels. If we assume that every pinwheel—as a first order approximation of the functional "hypercolumn"—contains the full set of visual features for a certain input location on one retina, it follows that the number of these distinct features is of the order 100. Inputs from the two eyes are treated independently here, so that successful stereoscopic fusion can be achieved for arbitrary depths by activating the right routing links.

While we have two agreeing estimates of the number of feature units per resolution point, coming from computer vision and physiology, the number α of features that can be routed together is difficult to estimate. It depends on the kinds of invariance operations that are actually realized in the routing circuit, as discussed in the second part of Section 3.3.1. We assume α to lie in the approximate range of 2 to 5. For these values, the optimal number of layers for routing $(k + 1)$ from a 10^6 node input to a 1000 node output ranges from 4.3 to 5.8. Figure 3.7 shows these values, as well as the number of units required for the full circuit when using the optimal

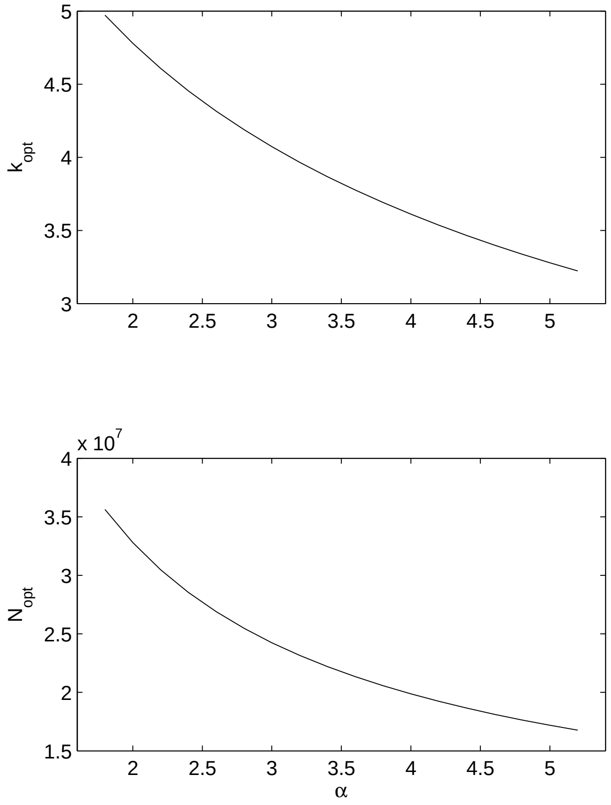


Figure 3.7: Routing from an input stage of 10^6 to an output stage of 1000 nodes. The upper part displays the optimal number of link layers as a function of α. Below we see the total number of required units using the optimal number of layers from above.

number of layers.

The ventral pathway comprises the areas V1, V2, V4, and IT. IT in turn consists of posterior, central, and anterior parts. In our setting it may make sense to take into account this additional subdivision, since the receptive field sizes of these three parts are very different (Tanaka et al., 1993; see also Figure 4 in Oram and Perret, 1994), suggesting that they form different stages of the routing hierarchy. Visual information is relayed from the lateral geniculate nucleus in a rather clear sequential order V1→V2→V4→PIT→CIT→AIT, finally being combined with other signal streams in the superior temporal polysensory area (STP). Note that there exist at least equally strong feed*back* connections between the layers, indicating the importance of recurrent processes in vision. The number of 4–6 distinct cortical stages (depending on whether we regard IT as one or three stages) lies clearly in the range derived for our optimal circuit above. It is therefore possible that the ventral pathway indeed performs computationally optimal information routing. At this point, however, this is only a hypothesis, due to the great uncertainties in the available data. More explicit interpretations would be possible if α could be narrowed down further (also by quantifying the "overcompleteness" of V1) and if the stages involved in the routing were known for certain. Also, more psychophysical work on the information content in the window of attention would be desirable.

3.5 Discussion

We have seen in Section 3.3 that under some very general assumptions there exists a clear optimality condition on the number of layers required to build a routing architecture with minimal neural resources. This number depends on the size of the target region as well as the number of independently routed feature types. Within the given uncertainties, the derived numbers agree well with physiological data.

Constraining the design of a routing architecture by an optimality condition, as we did, has the advantage of imposing an additional requirement to an otherwise underconstrained problem. While the Shifter Circuit of Olshausen et al. (1993) addresses several anatomical and physiological facts, there is no experimental or theoretical justification for some of the parameter values chosen, among them the exact doubling of link spacing from layer to layer. In the absence of experimental results dictating these values, we think it best to follow some global optimality condition like proposed above.

We are well aware that our very general assumptions can be refined in several ways, possibly changing the derived quantitative results:

- We avoided on purpose a detailed discussion of the kind of feature-to-feature connectivity that may be in place to achieve scale and orientation invariance. This is being addressed in ongoing work at our institute and will help to narrow down the parameter α.

- Routing architectures with many numbers of layers are a disadvantage to the organism in terms of longer reaction times and more complicated routing dynamics. This additional cost has not been considered here, its influence would bias the biological routing architecture in favor of fewer stages than derived here.

The analysis carried out in Section 3.3 also leads to experimental predictions. One such prediction arises from the fact that the notion of a static receptive field becomes meaningless if one embraces an active routing process. During attention focusing and recognition, this process would choose a certain routing path and deactivate all alternative pathways. For a unit at the output stage in the hierarchy (IT), this would change the functional receptive field from a very broad region to a narrow and specific location. A unit at a medium stage of the hierarchy might even be bypassed by the currently established routing pathway. There is ample evidence for the behavioral plasticity of receptive fields (Moran and Desimone 1985, Connor et al. 1993), and recent findings (Murray et al. 2006) show that even the size of representation in V1 can change with an object's perceived size (suggesting a scale invariant routing process that already starts in the mapping from LGN to V1). However, these findings are often interpreted as the result of a diffuse "attention modulation" mechanism, without taking the possibility of an explicit routing process seriously. In the light of the rather specific geometric changes of receptive fields implied by the presence of such a process, it should be possible to design attention experiments that can clearly prove or refute the routing hypothesis.

While the above predictions are general implications of any multi-stage routing process and have been discussed similarly before (Olshausen et al. 1993), the quantitative results obtained here make some more specific predictions. An interesting feature of the minimal architecture, already mentioned in Section 3.3.1, is that the number of links emanating from one node (see (3.1) or (3.11)) is independent of network size:

$$l_{\text{opt}} = n_\alpha^{\frac{1}{k_{\text{opt}}}} = \exp\left(\frac{\ln n_\alpha}{\tilde{c}\ln n_\alpha}\right) = \exp\left(\frac{1}{\tilde{c}}\right). \tag{3.17}$$

l_{opt} is surprisingly low (between 3 and 9 for the range of α shown in Figure 3.3). This number should not be confused with the full number of connections that a cortical neuron makes, which is known to be several thousand. First, here we only count the connections necessary for information routing, not those involved in other kinds of processing or communication. Second, as mentioned above, the functional units discussed here are abstract "image points", which in the cortex are probably made up of ≈ 100 spiking neurons (like a cortical minicolumn). Single neurons in such a group would have to devote the majority of their connections to homeostatic within-group connections (cf. Lücke and von der Malsburg 2004), which do not appear on our level of abstraction. Nevertheless, the small fanout necessary for optimal routing is an interesting feature and shows that by including the number of control units into our optimization we have implicitly also minimized the required connectivity of the routing architecture.

The optimal number of layers (equations 3.4' and 3.13), on the other hand, scales logarithmically with network size:

$$k_{opt} = \tilde{c}\ln n_\alpha.$$

This means that if more visual information has to be routed, the number of routing stages increases, while the local properties (number of connections that each node has to make) remain the same. Consequently, for species processing different amounts of visual information, the ventral streams should contain different numbers of routing stages. While the optic nerve of primates contains on the order of 10^6 fibers (Potts et al. 1972), the number is 10^5 for the rat

(Fukuda et al. 1982), $2 \cdot 10^5$ for the cat (Hughes and Wässle 1976), and $2.4 \cdot 10^6$ for the adult chicken (Rager and Rager 1978). If we assume, as we did before, that the number of optic nerve fibers is a measure for the number of input units of the ventral stream, *and* if the number of output (IT) units changes by the same factor, then a rat would optimally have 2.3 layers less, a cat 1.6 layers less, and a chicken 0.9 routing layers more than a primate. The differences might be smaller, however, if the size of the output stage does not change as strongly as the number of optic nerve fibers, since k_{opt} depends on the number of output units. Although anatomical comparisons across species will be difficult, it may be interesting to investigate different brains with regard to this question.

Although the switchyards derived in this chapter are useful for visual information routing and may be optimal in some sense, looking at those patterns provokes the question “how can such complex structures possibly arise in the brain?”. We will address this justified question in the following chapter.

4 Ontogenesis of Switchyards

In the previous chapter, we derived an optimized architecture for multi-stage routing of visual information. In order to be optimal in terms of number of neural elements, this network needs to have a very specific connectivity (cf. Figure 3.1). If we want to argue that such architectures are actually employed by the brain for information routing (which we do!), we must be able to explain how they develop in an animal.

One obvious assumption might be that this connectivity is directly encoded in the genes. For this to be possible, however, the raw information of the genome is too small. The human brain, for example, contains on the order of 10^{11} neurons. If we assume that each of them is only connected to 100 other neurons, describing this connectivity alone would require more than $4 \cdot 10^{13}$ bytes (40 terabytes). This is much more than the less than 1 gigabyte of information contained in the human genome (approx. 3200 megabases (Morton 1991) with 2 bit each). Consequently, the structure of the brain and most likely also single routing networks cannot be directly encoded in the brain, but must emerge during development of the organism in a self-organized way.

Studying self-organizational processes in the brain also has another, more indirect advantage. Sometimes researchers propose sophisticated models that can do “universal computation” or are “as powerful as a Turing machine”. But although interesting in their own right, these models will not provide insight into how the brain works if there is no explanation of how they could develop in a living organism. In that sense the study of self-organization helps the search for candidate models of brain function by providing constraints to the otherwise large space of possible models that could in principle explain a certain function.

Models of development of brain function can either be based on autonomous mechanisms that do not require external information, or they can incorporate learning processes where sensory input shapes the neural circuits. While many models of the development of visual function rely on learning processes happening postnatally, we argue here that basic visual functions like content-independent routing of information should already be operational at birth. The newborns of many hoofed animals, for example, can stand up and walk around directly after birth, indicating that at least rudimentary orientation in visual space is already up and running.

We therefore propose here an ontogenetic mechanism for the development of routing structures like the switchyards described in the previous chapter. We start out with a short overview of growth mechanisms in the brain (Section 4.1), before we proceed to describe the proposed model in detail (Section 4.2). Section 4.3 shows results and analyzes the behavior of the model. We discuss other potential mechanisms for the growth of switchyards (Section 4.4) before concluding with Section 4.5.

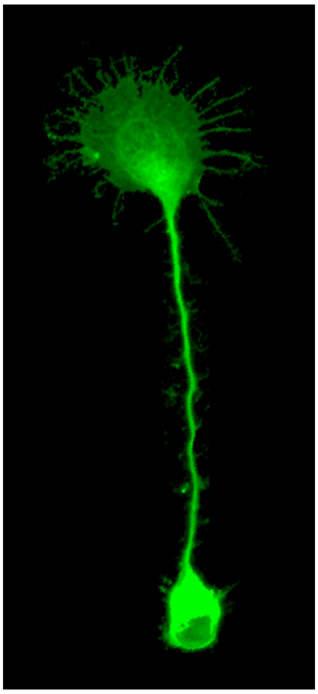


Figure 4.1: Image of a rat superior cervical ganglion neuron with an axon growing upwards. The axon ends in a growth cone which regulates future direction and strength of growth. Diameter of the growth cone is approximately $10 \mu m$. Courtesy of Geoff Goodhill.

4.1 Ontogenetic Plasticity Mechanisms in the Brain

Over the last 50 years, a lot of interest has focussed on the mechanisms that shape connectivity structures in the brain. Basically, we can distinguish two categories of mechanisms.

- Chemical interaction between neurons and also among fibers influences the direction and speed at which axonal fibers grow.
- Once contact has been established between two neurons, their connections can grow or shrink depending on the firing activity of both neurons.

An early idea for chemical interaction, the *chemoaffinity hypothesis* by Sperry (1963), assumes that axonal projections have a way of uniquely identifying their dedicated target location in the brain area they are growing towards. More recent findings, however, suggest that it is rather gradients of chemical markers that guide axon growth. When growing, an axon produces a *growth cone* with many little protrusions called *filopodia*. These filopodia can sense the gradient of chemical concentrations in their vicinity and will let the whole growth cone grow towards or withdraw from the source depending on whether the chemical has an attractive of repulsive effect on them (see Figure 4.1). For example, the retino-tectal projection in chicken is thought to be shaped by two opposing gradients of the substances “EphrinB” and “Wnt3” (Schmitt et al. 2006).

Another chemical mechanism is provided by attraction or repulsion of neighboring fibers (e.g., Holt and Harris 1993). Repulsive forces between fibers may be helpful in spreading connections homogeneously over the whole target tissue, while attractive forces could support the growth of parallel connections. See Table 4.1 for a non-exhaustive collection of substances involved in axon guidance.

Ligand(s)	Receptor(s)	Functional Roles
Netrins / Unc6	DCC/Unc40	Attracts commissural axons to floorplate
	Unc5	Repels trochlear/cranial motor neurons
	Neogenin	Attracts retinal ganglion axons to optic disk Thalamocortical / corticothalamic Cell Migration
Semaphorins	Neuropilins	Repulsion of sympathetic and sensory axons
	Plexins	Cortical axon/dendrite guidance
Slits	Robos	Repulsion from midline Axonal branching
Ephrins	Eph	Map formation: retinotectal, hippocampal-septal Nervous system segmentation/patterning

Table 4.1: A sample of chemical substances involved in axon guidance. Geoff Goodhill, personal communication.

Activity dependent mechanisms, on the other hand, play a role in refining connectivity once an axon has made contact to a target neuron. Most likely, they follow similar principles as the plasticity mechanisms that can be observed in the mature brain. These include the Hebb rule, anti-Hebbian, homeostatic mechanisms, or spike timing dependent plasticity. Experiments have shown that darkness or the blocking of neural activity by TTX (Olson and Meyer 1991) or disturbing correlated firing with stroboscopic light (Schmidt and Buzzard 1993) result in altered topographic projections, underlining the relevance of activity-based mechanisms. Consequently, for many years there was the notion that activity dependent mechanisms were required during ontogeny to produce the correct fine-structure of connections. However, recent findings suggest that under some circumstances chemical mechanisms alone may be sensitive enough. For example, Rosoff et al. (2004) have shown that growth cones can detect concentration differences as small as a single molecule across their spatial extent.

Below, we present an ontogenetic mechanism that is purely based on interactions of chemical markers. However, we are more interested in the functional properties of the mechanism than claiming a specific biological implementation, and in Section 4.4 we show how the same behavior can be achieved using only activity-induced plasticity.

4.2 A Model for the Growth of Routing Networks

As we discussed in Chapter 3, switchyards connect a layer of n input nodes via k routing stages to an output layer of $\frac{n}{m}$ nodes (see Figure 4.2). How can ontogeny produce such routing circuits

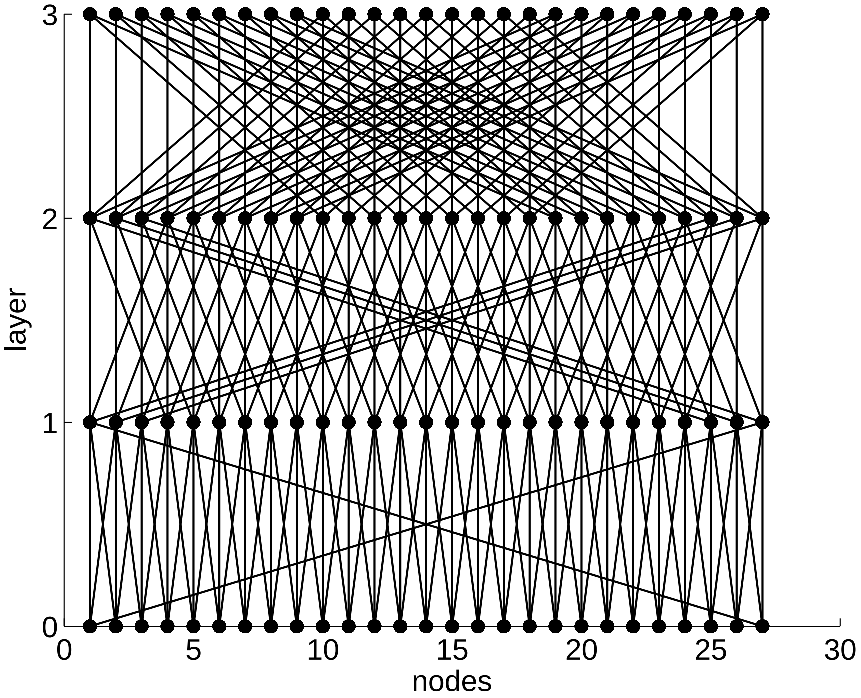


Figure 4.2: Switchyard architecture as derived in Chapter 3. The $n = 27$ nodes of the input layer 0 are connected to all 27 nodes of output layer 3 via 2 intermediate layers and $k = 3$ stages of links.

in the brain? Especially the large gaps necessary between links on higher stages are difficult to explain with traditional learning rules. We will investigate below whether chemical markers can help forming such structures. For simplicity and ease of visualization we restrict ourselves here to the case of one-dimensional feature layers. However, the mechanism derived below can also create three-dimensional networks connecting two-dimensional sheets of neurons, as we see in Section 4.3.2.

Let $C^{\kappa}_{i,j}$ denote the strength of the link between node i in layer κ to node j of layer $\kappa+1$. $C^{\kappa}_{i,j}$ can vary between 0 and 1, with 0 representing an absent link and 1 a fully grown one. We will refer to all links of one stage by the $n \times n$ matrix C^{κ}. When we make a statement that refers to the links of all k stages we will leave out the superscript κ (this also applies for other variables introduced below).

We describe the growth of the links not directly in terms of C but of an unbounded variable U, which codes for the real links via the sigmoid function

$$C = \frac{1}{1+e^{-sU}} \quad , \tag{4.1}$$

where s defines the steepness of the sigmoid. We let U start out at a homogeneous negative value with some noise added (see Section 4.3 for a discussion of robustness to noise), so that all links C are initially close to 0. The growth of U then follows the differential equation

$$\dot{U} = F^{\text{norm}} \times F^{\text{marker}} \times F^{\text{top}}, \tag{4.2}$$

where × denotes elementwise multiplication. The three terms have the roles of restraining local growth of connections (F^{norm}), keeping similarities of chemical markers on both sides of a link low (F^{marker}), and introducing topological interactions (F^{top}). Thanks to the multiplicative combination, no “tuning” of the relative contributions of the terms is required; the mechanism works for different network sizes without need for adjusting many parameters.

The term

$$F^{\text{norm}}_{i,j} = d - \sum_{\tilde{j}} C_{i,\tilde{j}} \tag{4.3}$$

is a factor that tends to keep the sum of all efferent links from any position i close to a desired value d. Once the combined link strengths exceed d, F^{norm} turns negative, thus letting the respective link shrink. Thus F^{norm} introduces competition among the links extending from any one neuron. Physiologically, this can be easily realized by a global signal in the soma of the neuron that keeps track of the overall strength of all its projections.

The term

$$F^{\text{top}}_{i,j} = \beta(C_{i-1,j-1} + C_{i+1,j+1}) + G_{i,j} \tag{4.4}$$

combines two different topological influences, their relative strength weighted by the parameter β. The first part adds cooperation between parallel neighboring links. This could be realized by cooperative fiber-fiber interactions as discussed in Section 4.1. While this term is not strictly necessary for the mechanism to work, it improves noise robustness of the whole process (see

Section 4.3.1). The second term G favors the growth of links to the corresponding position in the next layer ($i - j = 0$) over links to faraway positions. We assume it here to be a bounded hyperbolic function of position difference of the two end nodes:

$$G_{i,j} = \frac{\gamma}{|i - j| + \gamma}, \tag{4.5}$$

with γ defining the steepness (see Figure 4.3). G is necessary to tell the ontogenetic mechanism how to align the coordinate systems of the layers it is connecting. A possible way of implementing this term is to first allow development of a point-to-point mapping (e.g. through the mechanism of Willshaw and von der Malsburg, 1979), which then serves as a guidance for the growth of a routing connectivity. This means that the axons have already found their coarse target location, while the *axon collaterals*, the branched terminals of the axon, have not yet finalized their connections.

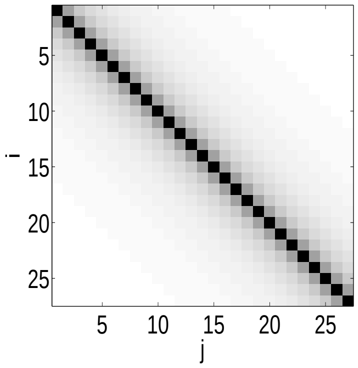


Figure 4.3: The term G helps to align the coordinate systems of subsequent layers by favoring links between nodes with corresponding positions (middle diagonal) over links between distant nodes. $\gamma = 0.6$ like in the simulations of Section 4.3.

The term F^{marker} represents the "heart" of our ontogenetic mechanism. It makes a link's change sensitive to the similarity of chemical markers in the two nodes it connects. These markers are channeled from the input layer to higher levels by the very connectivity C whose growth in turn they influence. We assume each node of the input layer to contain a different type of chemical marker t_i (for a discussion of the plausibility of this and possible alternatives, see Section 4.4). In matrix notation this means that the marker distribution in layer 0 is the identity matrix, $M^0 = I_{N \times N}$, with the marker types on the 1st and the node location on the 2nd dimension. Markers are then transported to higher layers via the existing links C:

$$M^{\kappa+1} = M^{\kappa} C^{\kappa} \ . \tag{4.6}$$

To calculate F^{marker}, we first define a similarity term

$$F^{\text{sim},\kappa}_{i,j} = \sum_{\mu} M^{\kappa}_{\mu,i} (M^{\kappa+1}_{\mu,j} - C^{\kappa}_{i,j} M^{\kappa}_{\mu,i}) \ , \tag{4.7}$$

which is the similarity (dot product) of the marker vector on the presynaptic side with that portion of the marker vector on the postsynaptic side that was *not carried there by the link itself* (μ is an index for the marker type). Therefore, the similarity term signals to the link how well the routes between the part of input space it "sees" and its target node are already being served by other links (see Figure 4.4). The role of F^{marker} is to let a link grow only if its similarity term is not too large. We therefore set

$$F_{i,j}^{\text{marker}} = 1 - H(F_{i,j}^{\text{sim}} - \alpha) \ , \tag{4.8}$$

with $H(\cdot)$ denoting the Heaviside function and a fixed parameter α.

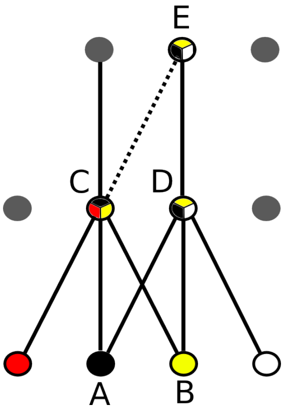
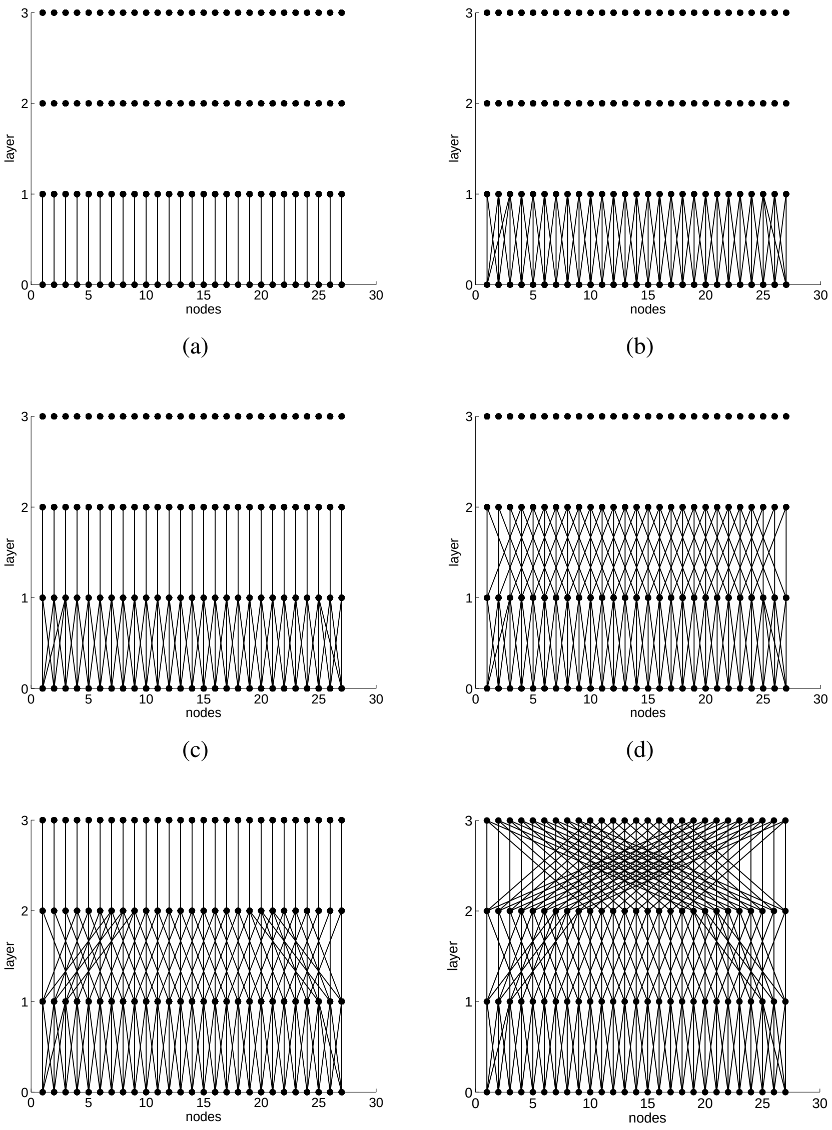


Figure 4.4: Role of the similarity term. Already well-established links (solid lines) carry markers from input nodes A and B to intermediate nodes C and D, and from D to E. Therefore, a weak link C-E (dotted line) finds a marker distribution at its target E that is similar to the one at its origin C. This similarity keeps it from growing. Functionally, this mechanism prevents formation of redundant alternative routes between two points (e.g., A-C-E would be an alternative to A-D-E).

4.3 Results

As example, we choose to investigate the growth of networks containing $k = 3$ link stages. We assume $d = 3$ as target number of links per node (see Eq. 4.3). With 3 link stages, this means that $n = d^k = 27$ input and output nodes can be connected. Equation (4.2) is integrated using the Euler method and the following parameter settings: $s = 30$ (steepness of sigmoid), $\alpha = 0.5$ (threshold for marker similarity), $\beta = 0.6$ (strength of neighbor interaction), $\gamma = 0.6$ (steepness of the hyperbolic term G). A delayed onset of growth at higher stages improves the final results. We choose a delay of 15% and 30% of overall simulation time for the middle and the highest stage, respectively.

When the process starts at the lowermost routing stage, the homeostatic term F^{norm} induces general growth of connections at this first stage. Under the influence of F^{top}, neurons first



(a) (b) (c) (d) (e) (f)

Figure 4.5: Snapshots of the growth process.

produce central links to the position in the next layer that corresponds to their own (Fig 4.5a. After this, adjacent links are produced, until F^{norm} stops the growth when the overall strength of each neuron's outward connections has reached its desired value d, here $d = 3$ (see Figure 4.5b). Note that the outermost neurons produce more links slanted to one direction, because they cannot spread their connectivity in both directions. Shortly after, growth sets in at the second stage and produces central links there (Figure 4.5c). The next links here, however, do not grow directly adjacent to the central one. This is because the marker concentrations in the respective nodes are already too similar, so that F^{marker} prevents growth of links here. This situation is visualized in Figure 4.4. The first node in the target layer that does not have any markers in common with the source node is exactly at a distance d from the central location. This is where the next links grow in the second routing stage (Figure 4.5d). Due to the network of links established by the first two routing stages, the chemical markers in the last intermediate layer are spread over a larger distance than in the previous layer. Consequently, once the first central links (Figure 4.5e) have brought markers to the output layers, marker composition in those two layers are similar over larger distances than before. Therefore, the first non-central links at the final stage arise at a distance of d^2 (Figure 4.5e).

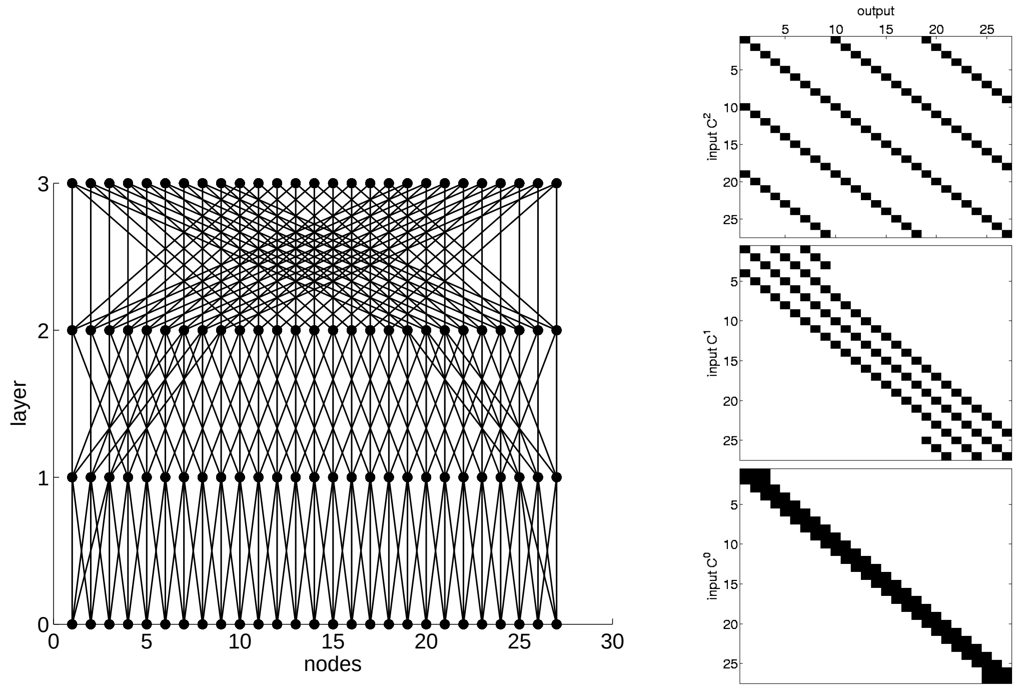
The final result of the ontogenetic mechanism is shown in Figure 4.6. Note how the distance between links increases from 1 to 3 to 9 from bottom to top, thus producing non-redundant full connectivity. We can see in Figure 4.6a that the resulting network differs qualitatively from the manually calculated one of Figure 4.2: there are no wrap-around links (i.e. links from a node on one side of the feature layer to the opposite side of the next layer). Instead, these links appear on the other side of the central link (cf. Figure 4.6b). Interestingly, this new structure produces the same perfect all-to-all connectivity as the one arising from theoretical considerations in Chapter 3, while being biologically more plausible.

The mechanism can also grow routing structures between larger feature layers. For this we only have to adjust the target number of links per node d, without changing any of the other parameters. Figure 4.7 shows simulation results for $d = 5$, i.e. $n = d^3 = 125$ nodes per layer. We see that qualitatively the resulting structure is similar to the one obtained for $d = 3$, except that now each node makes 5 connections to the next layer, with appropriate spacings of 1, 5, and 25 nodes.

4.3.1 Noise Robustness

However, we also see that the structure in Figure 4.7 is not as clean as the one in Figure 4.6, with several links not going to the "correct" targets. This means that while some pairs of nodes at the input layer and at the output layer of the switchyard are connected via two different routes, other such pairs may have no connection. Consequently, the input-output connectivity of the switchyard, which is defined by the concatenation of all routing stages, $\prod_{\kappa} C^{\kappa}$, is not perfectly homogeneous. For the structure shown in Figure 4.7, the strengths of the input-output connections have mean value $\mu \approx 1$ (which is the optimal value), but a standard deviation of $\sigma \approx 0.15$, which would be zero for a perfect connectivity.

The reason for the uneven final structure lies in the noise that was introduced to the initial link strengths. We chose the initial values of U randomly from the interval $[-16.5 \,..\, -15]$,



(a) (b)

Figure 4.6: Results for $n = 27$ nodes per layer and a target number of links $d = 3$. Note that this connectivity does not have "wrap-around" links, i.e. links between one end of the presynaptic layer and the opposite end of the postsynaptic layer. **(a)** Connection structure of the full network. **(b)** Matrices $C^{\prime\kappa}$ of the full network of (a) shown separately.

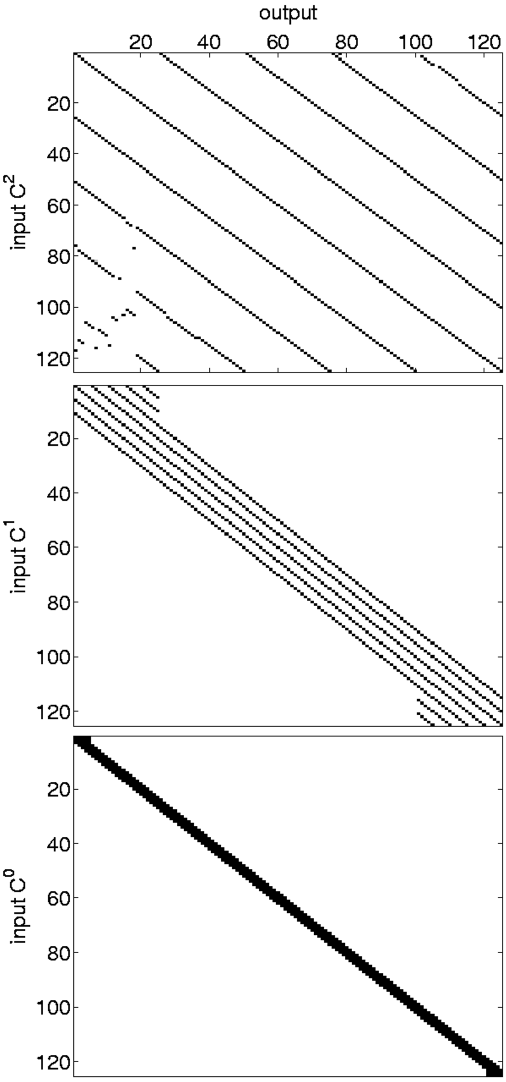


Figure 4.7: Resulting connection matrices C^{κ} for $n = 125$ nodes per layer and a target number of links $d = 5$. The initial values of U contained 10% of additive noise.

which means that they contain 10% of additive uniformly distributed noise. Further simulations have shown that the mechanism generally results in a flaw-less connectivity only if the initial conditions contain less than $\approx 5\%$ of noise. The growth of smaller networks is far less sensitive to noise. For $n = 27$, up to 20% of additive noise in the initial conditions practically always results in the correct final connectivity (cf. Figure 4.8).

A reason for this relatively high robustness to noise lies in the topological cooperation between neighboring links (the first component of the term F^{top}). In matrix notation, topological interaction at a stage κ is equal to the connectivity matrix C^κ convolved with an oriented kernel G^{top}:

$$F^{\text{top,inter}} = T * C^\kappa, \quad T = \frac{1}{2}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.9}$$

Previous models (e.g. the model for retino-tectal projections of Häussler and von der Malsburg, 1983) have employed unoriented topologic interaction between links. This was assumed to be biologically more plausible, because it only requires source and target distance, not the orientation of the links (but cf. the fiber-fiber interactions mentioned in Section 4.1 for a possible source of oriented interaction).

Here we assess these two cases (oriented and unoriented interaction) and a third case of no topological cooperation at all in terms of noise robustness. In the third case we replace topological interaction by a scalar constant equal to the mean of topological interaction in the other cases. For unoriented interaction, we use a normalized Gaussian kernel

$$T^{\text{unoriented}} = \begin{bmatrix} 0.0672 & 0.1828 & 0.0672 \\ 0.1828 & 0 & 0.1828 \\ 0.0672 & 0.1828 & 0.06721 \end{bmatrix}$$

with 0 at the central location. As can be seen in Figure 4.8, oriented cooperation is most robust, allowing noise levels of 20% before significant deterioration. Unoriented cooperation produces noisy results from 10% on. When no topological interaction at all is used, final results become noisy already for noise levels of 4%. For every condition (noise level, form of topology), we did 300 runs. A paired t-test on the results shows that the differences are statistically highly significant ($P < 0.001$, but much smaller in most cases). So we can conclude that topological cooperation is not strictly necessary for the mechanism to work, but it improves noise robustness of the ontogenetic process.

4.3.2 Growth of Three-Dimensional Networks

We have focused most of our attention on the ontogenesis of flat switchyards between one-dimensional layers. However, the same mechanism can in principle also explain the emergence of routing structures connecting two-dimensional layers of nodes. Figure 4.9 shows a growth process of a switchyard between layers of $9 \times 9 = 81$ nodes. The parameters used in Figure 4.9 are identical to those of Section 4.2; as the distance between two positions i and j required to calculate $G_{i,j}$ (see Eq. 4.5), we used the Manhattan distance, i.e. the L_1-norm. Unfortunately, these parameters do not produce perfect results in the case of three-dimensional switchyards.

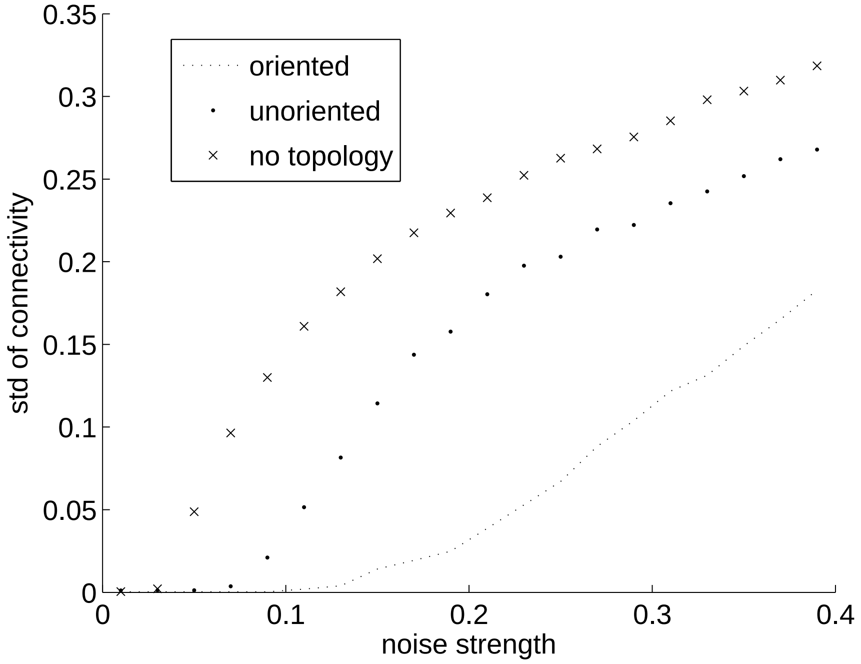
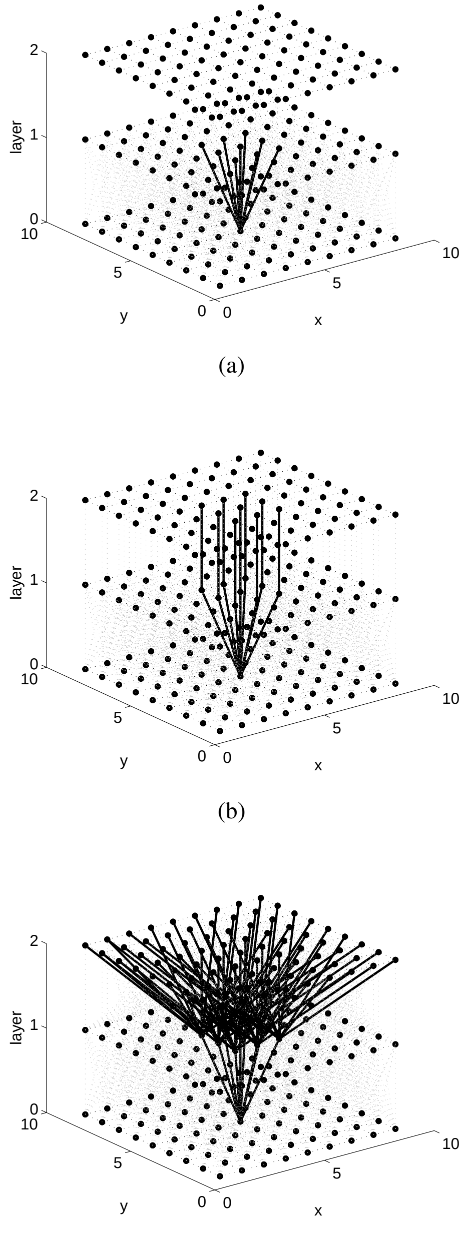


Figure 4.8: Noise robustness of the ontogenetic mechanism. The figure shows the normalized standard deviation of the resulting input-output connectivity (defined as the concatenation of all routing stages, see text) over relative noise strength in the initial values of U. For a perfect connectivity between input and output of the switchyard, this std should be zero. Cases shown are oriented topological interaction like F^{top} of the proposed mechanism, unoriented, Gaussian interaction, and no topological interaction at all. Values shown are for a switchyard size of $n = 27$ and have been averaged over 300 runs for each point. Differences between the curves are statistically highly significant.



(a)

(b)

(c)

Figure 4.9: Growth of three-dimensional networks. Dark lines show the pathways emanating from the central node of the input layer, for three instances during the growth process. All other connections are shifted versions of those (with circular boundary conditions) and are just hinted at by gray dotted lines. Parameters were $n = 81$, $d = 9$.

As Figure 4.10a shows, not all output nodes are connected to the full input layer in the final network. This changes when the growth process is made fully symmetric by removing boundary conditions in the growth terms. For this, both components of F^{top} have to be adapted. The topological cooperation between neighboring parallel links must be extended to "wrap-around" cooperation between links on opposite ends of a layer that would be neighbors if boundaries of a layer were connected. Likewise, the term G encouraging straight links must not be cut at the boundaries, but must wrap around to the other side of the layer. For such closed boundary conditions, the mechanism produces perfect connectivities also in the three-dimensional case (Figure 4.10b).

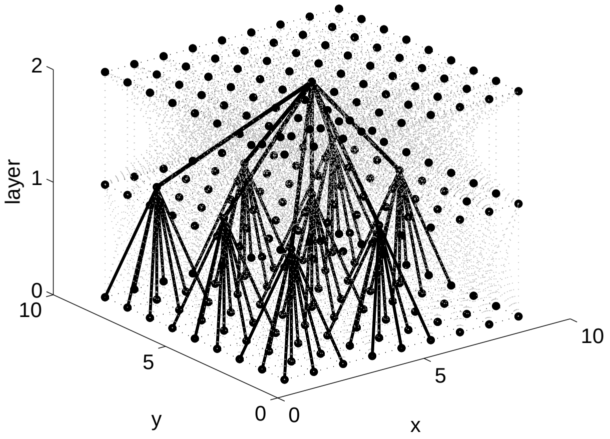
One aspect that may be responsible for the less robust growth of three-dimensional networks is the appearance of a rotational symmetry that does not exist for two-dimensional networks. While for one-dimensional layers there are exactly 2 nodes that have a certain distance l from a given node, for two-dimensional layers there is a whole ring of such nodes that sit on the edge of a square centered around that central node. Links to such equidistant nodes receive the same growth signal $G_{i,j}$, which will result in identical growth if everything else is equal. For one-dimensional layers this is no problem, since two links growing to nodes at distance l in opposite directions will not disturb each other. For two-dimensional layers, however, the candidate targets sitting densely on the edge of a square will compete, because a link to one of them would inhibit the growth of others through F^{marker}. Consequently, the decision where links will grow has to happen through spontaneous symmetry breaking in this case. If this symmetry breaking does not happen in a consistent, compatible fashion along the circle of candidate targets, a faulty connectivity will result. Such "symmetric", consistent symmetry breaking is more likely to happen if growth conditions are perfectly symmetric, too, which may explain why wrap-around boundary conditions result in better connectivities.

4.4 Other Potential Mechanisms

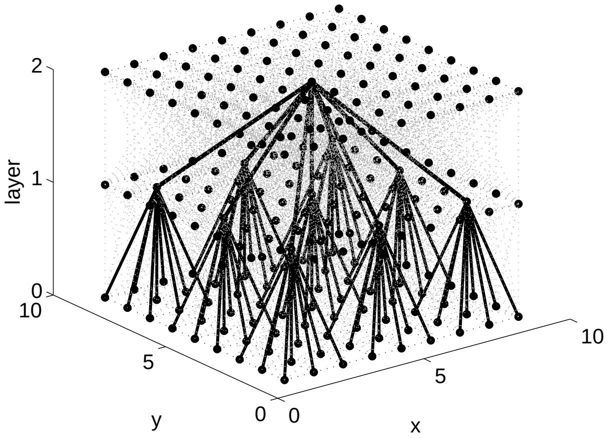
An unrealistic assumption of the growth model as presented in Section 4.2 is the existence of a unique chemical marker at every input node. It is unlikely that such a large number of distinct marker substances is available in an organism. However, other mechanisms are conceivable that are functionally similar or equivalent but do not require this assumption.

One possibility is a mechanism that operates with only a handful of chemical markers. The concentration of each such marker would be maximal at a certain input node and would fall of monotonously to both sides of that node. Consequently, neighboring nodes already at the input layer would have overlapping marker concentrations to some extent. This means, however, that already at the first routing stage the similarity term between some nodes would be > 0. In order to nevertheless allow tightly bundled links to grow at this first stage, the threshold α has to be raised high enough that this minimal similarity does not affect F^{marker} yet. With this parameter adjustment, a mechanism using only a few chemical markers should be qualitatively similar to the original version.

Instead of trying to find a mechanism with a limited number of markers, however, it is also possible to avoid the use of chemical markers altogether and reformulate the original mechanism



(a)



(b)

Figure 4.10: Three-dimensional results without (a) and with wraparound (b) boundary conditions.

of Section 4.2 completely in terms of activity correlation. For this we assume that all input nodes are spontaneously active, $\xi^0_\mu(t)$ representing the activity time course of input node μ. We define a scalar product

$$<\xi,\zeta>=\frac{1}{T}\int_t^{t+T}\xi(\tau)\zeta(\tau)d\tau \tag{4.10}$$

to quantify the similarity between the activities ξ and ζ of two nodes. We require that activities of input nodes are random, uncorrelated with each other, and have constant average energy. Without loss of generality, we can assume this average energy to be normalized such that the scalar product with itself is 1. Under these assumptions, the activities of the input nodes form an orthonormal basis:

$$<\xi^0_\mu,\xi^0_{\mu'}>=\delta(\mu,\mu'). \tag{4.11}$$

Nodes at higher stages are activated by the input layer via their afferent connections. At the first intermediate stage, this activity

$$\xi^1_i=\sum_\mu C^0_{\mu,i}\xi^0_\mu$$

results from the direct links leading from the input stage to node i; for arbitrary nodes of higher stages, it is a general superposition of input activities defined by the connectivity of this node to the input layer:

$$\xi^\kappa_i=\sum_\mu m^\kappa_{\mu,i}\xi^0_\mu, \tag{4.12}$$

$m^\kappa_{i,\mu}$ representing the coefficients of this expansion. This expansion is unique due to the ξ^0_t being orthogonal. The similarity of the activities of two nodes in subsequent layers is then given as

$$<\xi^\kappa_i,\xi^{\kappa+1}_j>=\sum_\mu\sum_{\mu'} m^\kappa_{i,\mu}\xi^0_\mu m^{\kappa+1}_{j,\mu'}\xi^0_{\mu'}, \tag{4.13}$$

which simplifies to

$$<\xi^\kappa_i,\xi^{\kappa+1}_j>=\sum_\mu m^\kappa_{i,\mu}m^{\kappa+1}_{j,\mu} \tag{4.14}$$

due to the orthonormality of the input layer activities (Eq. 4.11). Comparing (4.14) to the term $\sum_\mu M^\kappa_{\mu,i}M^{\kappa+1}_{\mu,j}$, which represents chemical marker similarity in (4.7), reveals that the mechanisms for expressing similarity between two nodes are equivalent whether chemical markers or activity correlations are used. Consequently, we can define F^{sim} and then F^{marker} in exactly the same way as done in (4.7) and (4.8), only that now the functional mechanism is neural activity instead of chemical markers. The terms $F^{\text{norm}}_{i,j}$ (defined in Eq. 4.3) and $F^{\text{top}}_{i,j}$ (defined in Eq. 4.4) can remain unchanged since they do not involve chemical markers at all. In this way, the whole growth mechanism has been reformulated in terms of activity correlation without functional changes.

4.5 Conclusion

We have presented a neurally plausible ontogenetic mechanism modeling the formation of routing circuits in the brain. The mechanism requires only signals that are available locally at the source and/or target of the respective connection. Interestingly, its results turn out to be biologically more plausible than equivalent connectivities derived from mathematical analysis in Chapter 3 (compare Figs. 4.2 and 4.6a). While the mechanism is useful to understand the development of certain wiring structures of the brain, it may also turn out to have technological applications like the automatic wiring of computer networks.

Previous ontogenetic models have mostly focused on retino-tectal projections (e.g., Willshaw and von der Malsburg 1979, Weber et al. 1997). One of the few exceptions are Linsker's models for the development of receptive fields in V1 (Linsker 1986). Therefore the study of more complex structures like the switchyards investigated here promises to yield important new insights, but on the other hand it also has much poorer experimental foundations than the intensively studied retino-tectal projection. A lot of anatomical and physiological work on the ontogenesis of connections in cortex (as opposed to subcortical areas) is necessary to provide a tighter scaffold for future models.

So far, we have focussed on the optimal structure (Chapter 3) and growth (this chapter) of switchyards. In the final chapter, we will investigate how such structures can be used in combination with the ideas of Chapter 2 to make progress towards realistic models of visual object recognition.

5 Putting the Pieces Together: Recognition with Switchyards

In the previous chapters, we have developed several building blocks for a neural, correspondence-based vision system. In Chapter 2, we discussed how patterns can be compared by finding corresponding points through an all-to-all connectivity, and we showed how these principles can be extended into a neural face recognition system. In Chapters 3 and 4, we investigated how neural patterns can be connected more efficiently by using multi-stage switchyards, ignoring, however, the question how these connections may be controlled dynamically. In this chapter, we will combine these ideas. In Section 5.1, we study how correspondences between patterns can be found if the neural representations of these patterns are not directly connected, but only indirectly via a switchyard of several routing stages. In Section 5.2, we will show how this idea can be extended into a system that can match *and* recognize patterns from a gallery.

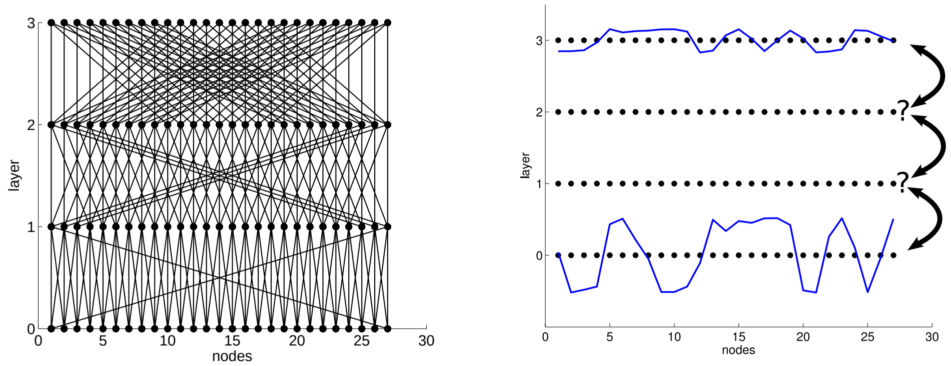
Why should it be necessary to match patterns of neural activity via intermediate stages? The short answer to this is (neural) economy. As discussed in detail in Chapter 3, it would be unrealistically expensive in terms of neural circuitry to connect all potential matching partners via direct connections. Therefore it will be a common situation that the brain has to match patterns that are only indirectly connected via intermediate layers. Switchyards (like the one in Figure 5.1a) can provide such a connectivity using a minimal number of links and intermediate feature nodes.

We believe that this is part of a more general computational principle of the brain. Matching two patterns means finding a mapping or transformation from one "input" pattern to another "target" pattern. If the patterns are connected directly (cf. Sect 2.1), the dynamic links between them have to express this transformation in a single step. However, if the patterns are connected by a switchyard, then each routing stage only represents a partial transformation, all of which are carried out in series to produce the full mapping. In that sense, multiple stages can generate a huge space of transformations through a *combinatorial code* of their simple mappings. This is computationally advantageous to the direct connectivity case, where every single of these possible transformations needs to be represented explicitly.

But while being more parsimonious in terms of required neural circuitry, using multiple stages to connect patterns introduces additional problems when trying to match these patterns. When patterns are connected directly, dynamic links between them can directly evaluate the similarities of image points and thus establish a global match between the patterns. But when the patterns are connected only indirectly via multiple stages, then the dynamic links of the first routing stage do not have a clear matching partner for the input pattern, and while these early dynamic links have not converged, later routing stages will not receive the properly transformed input pattern to which they can match the target pattern (see Figure 5.1). In the following we discuss under what conditions and how matching via multiple stages can work nevertheless.

5.1 Matching of Two Patterns

We consider the case of two patterns connected by a switchyard of connections as introduced in Chapter 3. In our experiments in this section and Section 5.2, we use a switchyard with $k = 3$ routing stages and a layer size $n = 27$ as shown in Figure 5.1a. On top of the purely static connectivity investigated there, we now assume, however, that each connection in the switchyard is governed by a control unit and thus forms a dynamic link which becomes active based on feature similarity of the points it connects and cooperation with its neighboring links (see Section 2.1 for details). Additionally, the intermediate feature representing layers of the switchyard now consist of two separate layers to accommodate separate streams of information flowing upward and downward (see Figure 5.2). Following the nomenclature introduced in Chapter 2 (cf. also Figure 2.7), we call the layers representing bottom-up information *Input Assembly* and the layers representing top-down information *Gallery Assembly*. Like in Chapter 2, both feature and control units follow the modified evolution equation (2.1), without competition for the feature units and with competition in the case of control units. Control units govern both upstream and downstream information flow, and they are activated by evaluating the similarity between features in the Input Assembly Layer right below and the Gallery Assembly Layer right above them (again, see Figure 5.2, and also cf. Figure 2.7, where the same entangled principle is used):



(a) (b)

Figure 5.1: Challenge of matching patterns via a switchyard. **(a)** The switchyard used in the experiments in this and the following section has $k = 3$ routing stages and a layer size $n = 27$. Consequently, the fanout at every node is $l = 27^{\frac{1}{3}} = 3$. **(b)** Here, only the feature nodes of the switchyard are shown, together with the patterns present at input and output of the circuit. The challenge is to match these patterns via a succession of routing stages although a priori not all of the stages have direct access to active patterns that can be used for correspondence finding.

$$I_{\text{sim}}(\mathcal{C}^{\kappa}_{ij}) = \mathcal{IA}^{\kappa}_{j}\,\mathcal{GA}^{\kappa+1}_{i}, \tag{5.1}$$

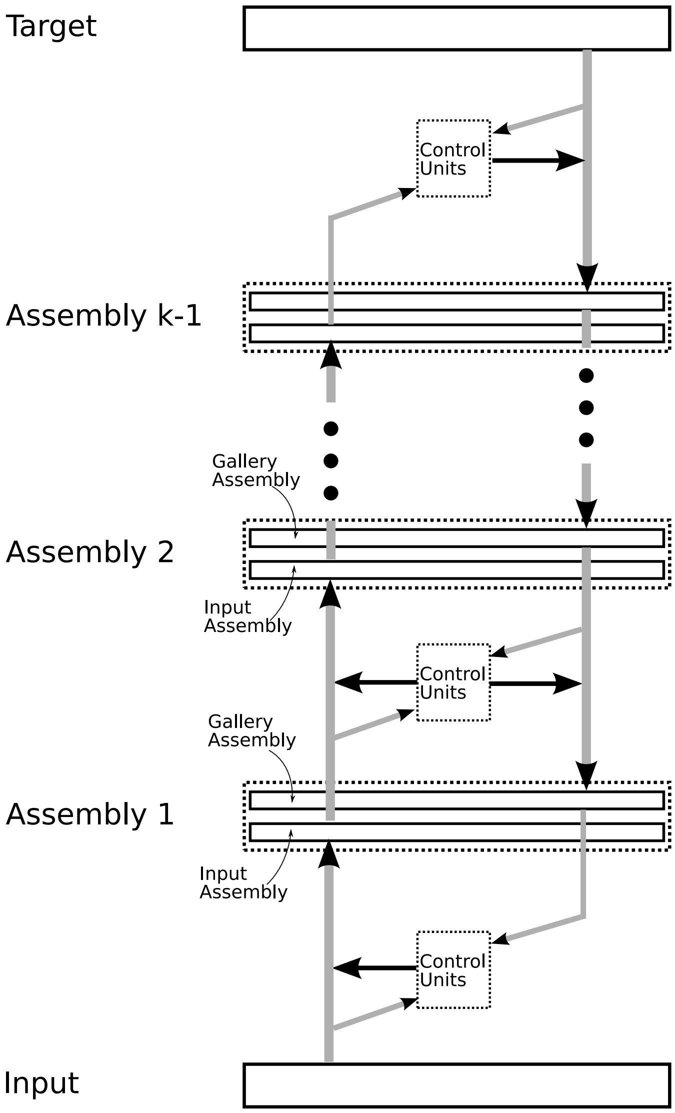


Figure 5.2: Information flow in the dynamic switchyard. The input pattern is propagated upstream via the Input Assembly layers, while the target pattern is routed down through the Gallery Assembly layers. All routing is governed by the control units, which in turn are driven by comparison of the activities of the Input Assembly directly below to the Gallery Assembly directly above them.

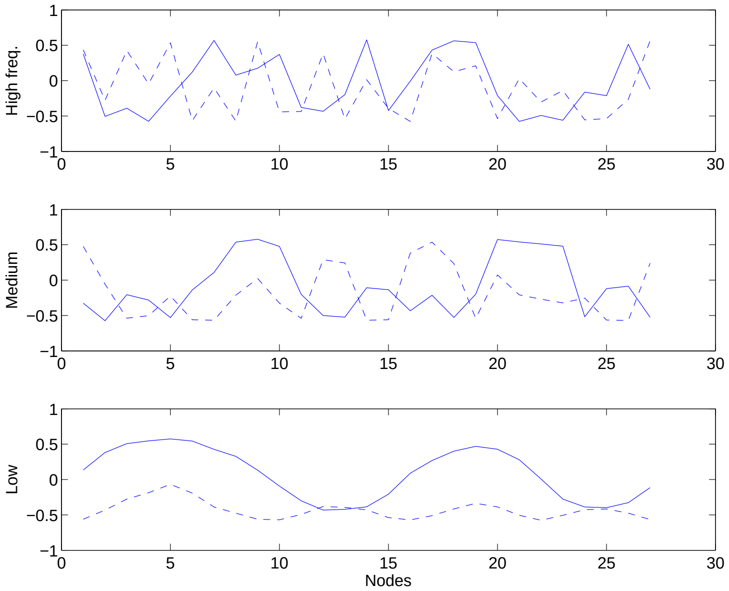
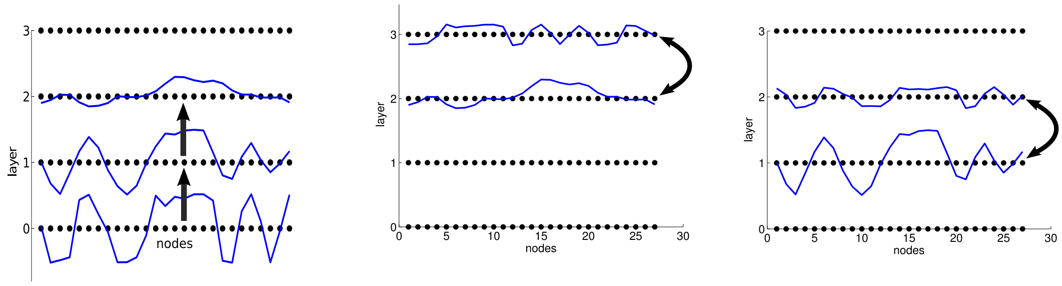


Figure 5.3: Typical pattern used for matching via a switchyard. At each position, the pattern consists of 6 components. Every two of them are the real (solid line) and imaginary (dashed line) components of a complex number of unit length. This ensures that the energy of the features is equal at each point. The arguments or angles defining these 3 complex unit vectors (via $z = \exp(i\phi)$) are drawn randomly and independently from a uniform distribution between 0 and 2π for each point and each channel. While the angle pattern of one channel is left unchanged, the others are convoluted with Gaussian filters of width $\sigma = 1$ and $\sigma = 4$ before being converted to complex numbers to produce medium and low frequency signals.

with $\mathcal{C}_{ij}^{\kappa}$ representing the control unit between Input Assembly ($\mathcal{IA}$) unit j in layer κ and Gallery Assembly ($\mathcal{GA}$) unit in layer $\kappa + 1$. Additionally, control units of neighboring parallel links cooperate with interaction strength $c_{\text{top},\mathcal{C}} = 1.2$ (see Section 2.4.2 for details on topological interaction among control units).

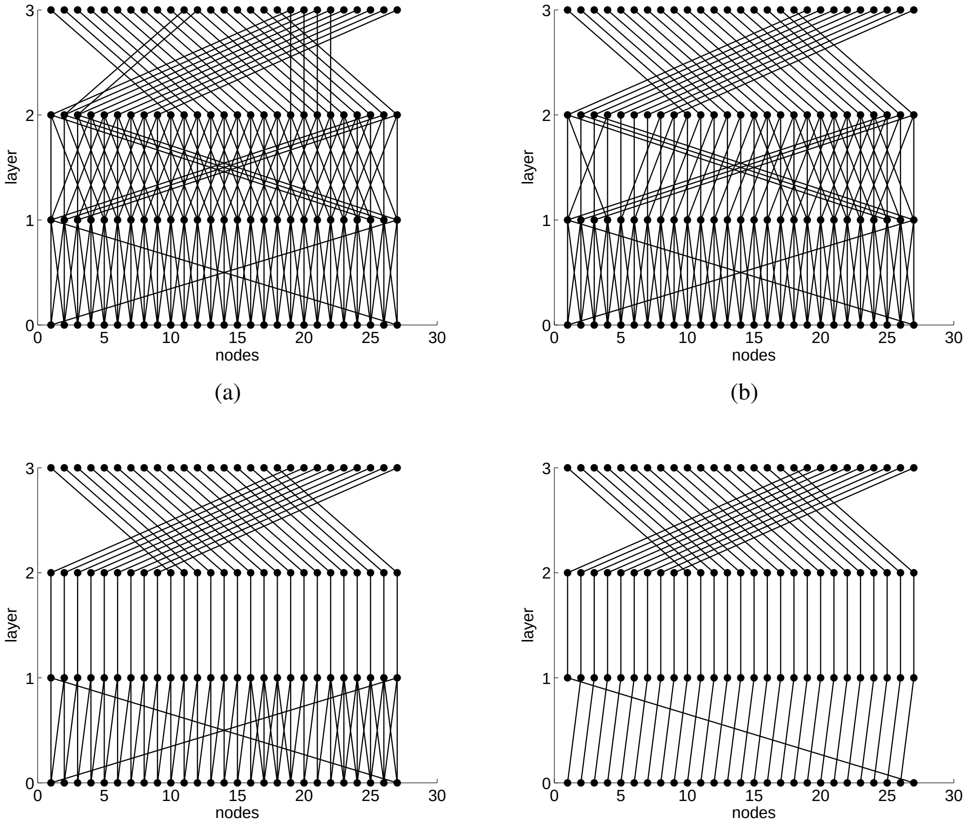
How can such a switchyard self-organize into a state that represents a match between the patterns at its input and its output stage? We will focus here on position-invariant matching of one-dimensional visual patterns (like those shown in Figure 5.3) as a proof of principle. In this case, the first routing stage of a switchyard with its tight bundles of connections represents small, but very accurate translations of visual information. The following stages carry out farther reaching but coarser translations, while the highest stage is responsible for global shifts at the scale of the whole image. If these transformations between two patterns are to be estimated more or less independently, both patterns need to contain significant information over a large range of spatial frequencies: low frequencies to match the patterns coarsely at the final routing stage, down to high resolution information to decide on the fine transformations happening at the first routing stage. This condition of having sufficient energy at all frequencies is satisfied for most natural images, but we will discuss the consequences of degenerate cases in Section 5.3.



(a) (b) (c)

Figure 5.4: The principle of matching in a switchyard. **(a)** Initially, all links in the switchyard are open. The input pattern is propagated upwards and blurred on the way. **(b)** Matching starts between the target pattern at the output of the switchyard and the blurred version of the input pattern at the highest intermediate stage. **(c)** Once the links at the highest stage have converged, they send down a properly translated version of the target pattern (9 points to the right in the case shown here) to the next routing stage, where it is matched with a less blurred version of the input pattern to determine the medium-range correspondences. This process travels downstream until the very fine correspondences of the lowest routing stage have been established.

Nevertheless, we are still confronted with the problem of trying to find this match in a neural fashion via a switchyard of several stages, each of which can only “see” and match the information directly adjacent to it (see Figure 5.1). However, when all connections are open, the successive stages of a switchyard act as low pass filters with different cut-off frequencies. Thanks to this property, a switchyard can actively match visual patterns that have sufficient energy at all frequencies (like those shown in Figure 5.3) as described in the following. Initially, all control units of the switchyard are active (cf. Chapter 2), which means that all links in the switchyard can pass information. Since in this state the early stages of the switchyard act as a low pass, a very blurred version of the input pattern reaches the highest intermediate stage (see Figure 5.4a). Competition between the control units sets in first at the highest routing stages and initiates a correspondence finding process between this low-pass version of the input pattern and the target pattern at the output of the switchyard (Figure 5.4b). Owing to the significant low frequency components of the patterns, this blurred version of the input is sufficient to find a very coarse estimate of the correct match with the target pattern, just as accurate as it can be represented by the sparse, far-reaching connectivity provided by the highest routing stage. Once the dynamic links of the highest stage have settled onto this coarse match, they can pass down to the highest Gallery Assembly Layer a version of the target pattern that is already shifted to the approximate position of the input pattern. This pattern can now be matched at a medium routing stage with a less blurred version of the input pattern to establish the medium range translations between the patterns (Figure 5.4c). Finally, a version of the target pattern that is nearly at the same location as the input pattern is passed down to the lowermost intermediate stage, where it is matched with the original input pattern to represent the finest details of the mapping in the activities of the dynamic links of the first routing stage. Snapshots from a simulation of the matching process are shown in Figure 5.5.



(a) (b)

(c) (d)

Figure 5.5: Matching via a switchyard. The input pattern is shifted by 8 image points to the right with respect to the target pattern (as in Figure 5.1b). Images show the activity of the dynamic links at different stages of the matching process. **(a)** Matching at the highest routing stage has just started. **(b)** The highest routing stage has converged to a shift of 9 image points, while the process has begun at the middle stage. **(c)** The middle stage has produced a straight match, and matching has set in at the lowest stage. **(d)** All dynamic links have converged, representing a transformation that maps the input pattern 8 points to the left.

Comparison to Olshausen

In his Phd thesis, Bruno Olshausen (1994) derived analytically how a routing circuit (his famous *Shifter Circuit*) can autonomously focus on a Gaussian blob. This process is related to the pattern matching via switchyards described above, and we will see that both processes are functionally nearly equivalent. There are some important differences, however, which make Olshausen's implementation less realistic.

To analytically derive a mechanism for blob focussing, we define an error function E_{blob} that quantifies the mismatch of the output pattern of the routing circuit I^k with a target blob G. If we restrict ourselves to circuits with two routing stages (as Olshausen did), the error function is defined as (the following formulae have been adapted from Chapter 2 of (Olshausen 1994) to the nomenclature used in this dissertation)

$$E_{\text{blob}} = -\sum_i \mathcal{I}_i^2 G_i, \tag{5.2}$$

with $\mathcal{I}_i^2$ the activity at the i^{th} position of the 2nd (=output) stage of the routing circuit. Note that a Shifter Circuit routes information only in the feedforward direction. We assume here that visual information is propagated linearly and without delay by the routing circuit:

$$\mathcal{I}_i^2 = \sum_j \mathcal{C}_{ij}^1 \mathcal{I}_j^1 = \sum_j \sum_k \mathcal{C}_{ij}^1 \mathcal{C}_{jk}^0 \mathcal{I}_k^0, \tag{5.3}$$

with $\mathcal{C}^\kappa$ representing the control unit activities at stage κ. With this, the derivative of (5.2) with respect to the control units at the second routing stage is

$$\frac{\partial E_{\text{blob}}}{\partial \mathcal{C}_{ij}^1} = -\mathcal{I}_j^1 G_i, \tag{5.4}$$

and the derivative with respect to the lower routing stage results as

$$\frac{\partial E_{\text{blob}}}{\partial \mathcal{C}_{jk}^0} = -\sum_i \mathcal{I}_k^0 \mathcal{C}_{ij}^1 G_i. \tag{5.5}$$

Olshausen then concludes his derivation by defining the blob focussing dynamics through gradient descent on the error function, leading to

$$\tau \dot{\mathcal{C}}_{ij}^1 = \mathcal{I}_j^1 G_i \tag{5.6}$$

and

$$\tau \dot{\mathcal{C}}_{jk}^0 = \sum_i \mathcal{I}_k^0 \mathcal{C}_{ij}^1 G_i. \tag{5.7}$$

In our approach to matching in a routing circuit, on the other hand, there exist separate streams of upward and downward information flow. In steady-state (adiabatic solution of the fast dynamics of (2.2)) the upward stream is described by

$$\mathcal{IA}_i^{\kappa+1} = \sum_j \mathcal{C}_{ij}^\kappa \, \mathcal{IA}_j^\kappa \tag{5.8}$$

and the downward stream by

$$\mathcal{GA}_j^\kappa = \sum_i \mathcal{C}_{ij}^\kappa \, \mathcal{GA}_i^{\kappa+1}, \tag{5.9}$$

starting with

$$\mathcal{IA}^0 = I$$

as the input pattern and

$$\mathcal{GA}^k = G$$

as the target pattern (in this case a Gaussian blob). As discussed above, the dynamics of our control units are driven by—besides topological cooperation—the similarity of the Input Assembly location right below and the Gallery Assembly location right above the link (see Eq. 5.1). In the case of a two-layer switchyard, this yields

$$I_{\text{sim}}(\mathcal{C}^1_{ij}) = \mathcal{IA}^1_j \, G_i, \tag{5.10}$$

which is identical to the right-hand-side of dynamics (5.6), and

$$I_{\text{sim}}(\mathcal{C}^0_{jk}) = I_k \, \mathcal{GA}^1_j, \tag{5.11}$$

which is equivalent to the rhs of (5.7) since

$$\mathcal{GA}^1_j = \sum_i \mathcal{C}^1_{ij} \, G_i$$

(compare Eq. 5.9).

So in principle, both approaches produce equivalent results when applied to blob focussing, if we leave aside details of the underlying dynamics. But there is one fundamental difference between both systems that has important implications: Shifter Circuits only route visual information in one direction, from the input to the output stage. The top-down stream that exists in our switchyards is not explicitly implemented. However, this feedback information is required by the dynamics (5.7), which means that the control units at the lower stages of a Shifter Circuit would need additional, dedicated connections to both the target pattern G and the control units of all stages above them. This provokes the same critique as backpropagation training of multilayer perceptrons: it requires non-local information and therefore is biologically less plausible than a solution using only locally available signals. One possibility to reduce the number of those intricate connections at least a bit would be to do away with the direct connections to the target pattern by hardwiring its values as weights into the connections to the control units of higher stages. This would entail, however, that the circuit now could only use a predefined Gaussian blob as target pattern. While this may allow segmentation in very simplified situations, it will—as already noted by Olshausen (1994)—usually not provide enough details to perform model-based recognition on top of a routing circuit. Moreover, the idea of an explicit downward stream of information agrees with findings that visual imagination evokes activity at “early” visual areas, and it follows our general commitment to regard vision as a generative process (see discussions in Chapter 1 and Section 2.6).

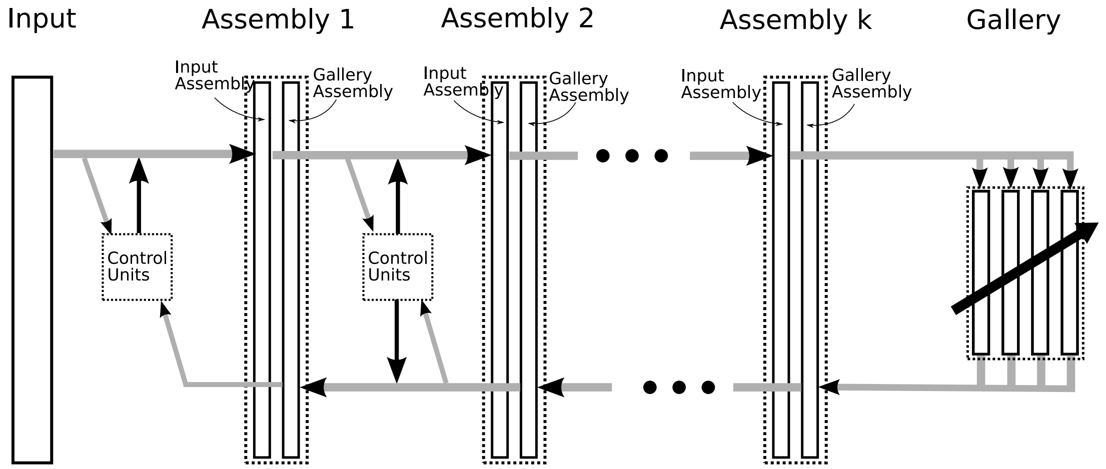


Figure 5.6: Information flow in the full recognition system. The input pattern is propagated upstream via the Input Assembly Layers of the switchyard to finally activate those units in the Gallery representing the most similar pattern. Simultaneously, signals from the Gallery are sent to the Gallery Assembly k, where they form a superposition that is routed downstream through the lower Gallery Assembly Layers. All routing is governed by the control units, which in turn are driven by comparison of the activities of the Input Assembly directly below to the Gallery Assembly directly above them. Competition among the control units results in the activation of a unique mapping between Input and Gallery, while competition among Gallery units forces the system to "recognize" one of the patterns stored in the gallery.

5.2 Recognition from a Gallery of Patterns

In this section, we will investigate how switchyards can be incorporated into an invariant object *recognition* system. In principle, this is a straightforward combination of the ideas derived in Section 5.1 and the system developed in Chapter 2. We will demonstrate the essential functionality of the approach with the recognition of one-dimensional patterns from a gallery. Position invariance is achieved by the same 3-stage switchyard used in Section 5.1. The system performs recognition from a gallery containing 10 "random" patterns (see Figure 5.7). Topological cooperation (refer to Section 2.4.2 for details) among neighboring parallel links in the switchyard is $c_{\mathrm{top},\mathcal{C}} = 1.2$, cooperation among neighboring Gallery units representing the same pattern has the same value $c_{\mathrm{top},\mathcal{G}} = 1.2$.

We provide an input pattern to the system that is identical to one of the patterns of the gallery, except that it is shifted by several image points. The input pattern enters the system through the Input Layer of the switchyard (see Figure 5.6) and is propagated upwards through the successive Input Assembly Layers. Since initially all control units have a small, non-zero activity, a very lowpass-filtered version of the input reaches the $(k-1)$th Input Assembly, as discussed in Section 5.1 (cf. also Figure 5.4a). The function of the target pattern of Section 5.1 is now taken over by the Gallery Assembly k, which receives input from all Gallery units. Different

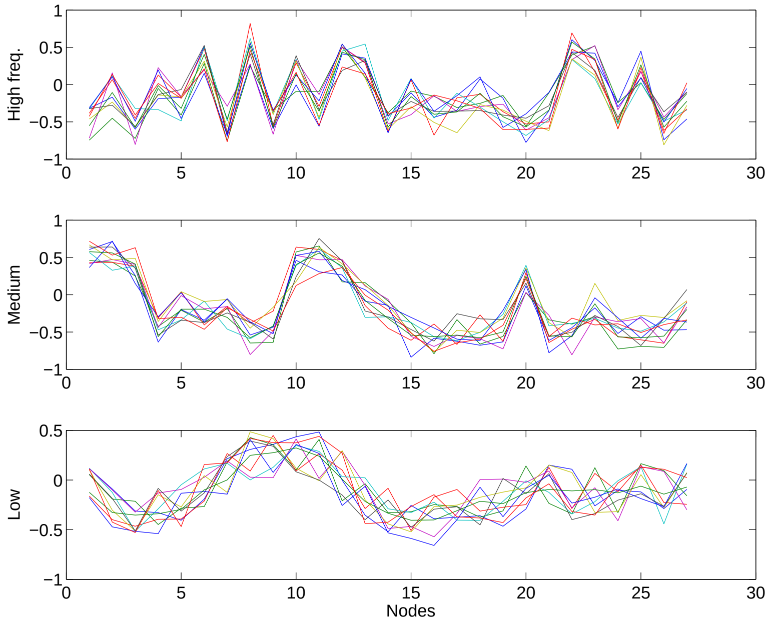
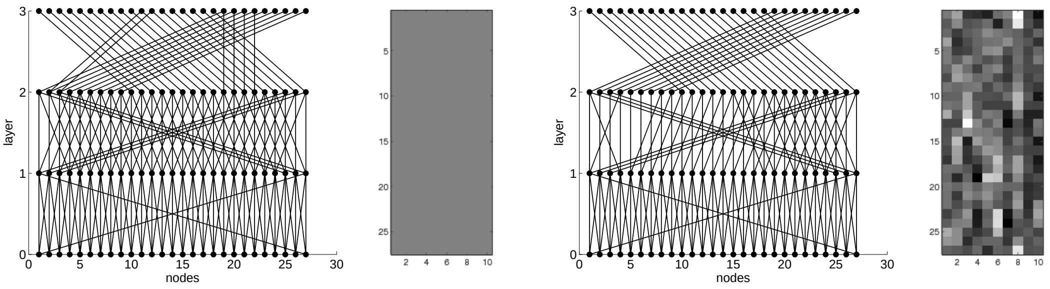


Figure 5.7: Patterns stored in the gallery. The gallery contains 10 six-dimensional patterns which are defined as described in Figure 5.3. Only the three components corresponding to the real parts of the phase angles are shown here. All patterns are generated from the same template (to emulate the structural similarity of different images coming from the same class of objects), but contain 20% of individual, uncorrelated white noise for all three channels.

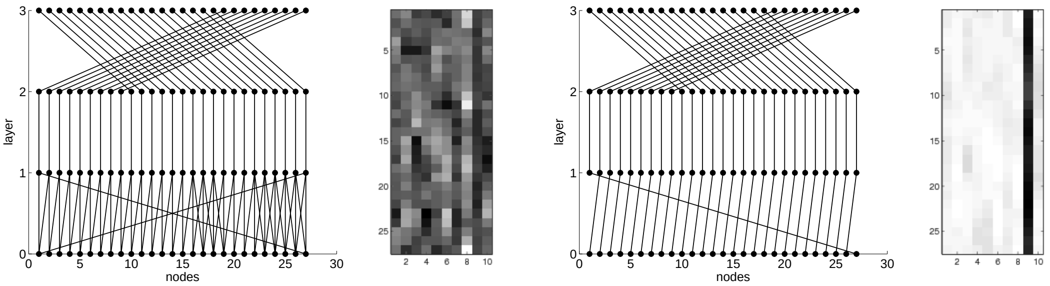
than before, however, this target pattern will vary over the recognition process as the activities of the Gallery units change. Initially, all units in the Gallery are equally active, just like the control units of the switchyard. Therefore the highest Gallery Assembly will initially contain an average of the patterns stored in the Gallery. Since those patterns are created from the same template, this average will contain very similar low-frequency components as the pattern at the Input Assembly $k-1$. This low-frequency information is sufficient to find the rough, long-ranging correspondences that are controlled by this highest routing stage (see Figure 5.8a, also compare Figure 5.4b). Once the control units at this highest stage have converged, the average Gallery pattern can be propagated downwards to Gallery Assembly $k-1$ under the correct long-range translation. There it is used by the second-highest routing stage to find the medium-range correspondences with the version of the input pattern present in Input Assembly $k-2$. The matching process continues downstream through all routing stages until also the very fine correspondences between Input and Gallery Assembly 1 have been established.

In the same way that the decision at the highest routing stage allows the correct downward propagation of the pattern in the highest Gallery Assembly, it also allows the blurred input pattern at Input Assembly $k-1$ to be routed forward under the correct long-range translation to



(a) $t = 0.3\,T$ ($T\widehat{=}$ full simulation time)

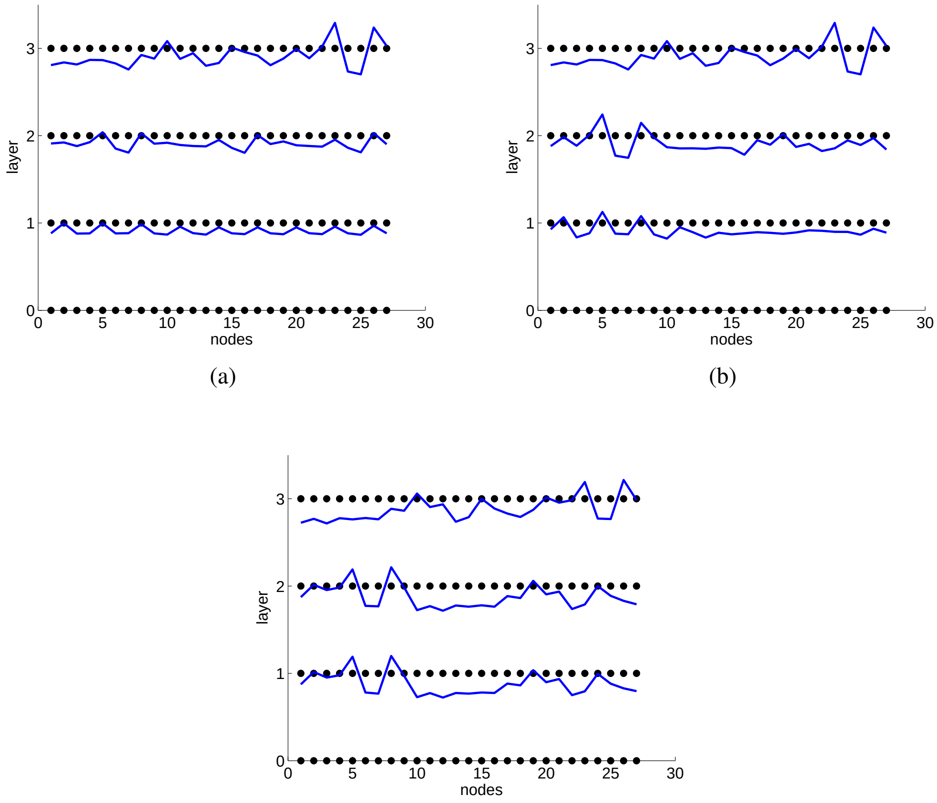
(b) $t = 0.5\,T$



(c) $t = 0.7\,T$

(d) $t = 0.9\,T$

Figure 5.8: The process of pattern recognition in the full system. In the example shown here, the system is confronted with an input pattern that is identical with the 9th pattern in the gallery, but shifted by 8 points to the right. The left side of each subfigure shows the active links in the switchyard connecting the Input with the highest Assembly stage. The right side shows the activation of the Gallery Layer units. Black represents a strong pathway or strong activity. **(a)** The dynamics of the Gallery start later than those of the highest stage of the switchyard (with a delay of 40% of the full simulation time T). The highest control units of the switchyard have already a clear tendency for a shift of 9 image points, while all Gallery units are still in their initial homogeneous state. **(b)** Activity in the gallery has set in, but no trend for any specific pattern is noticeable yet. The highest stage of the switchyard has mostly converged, while at the medium stage the central links are starting to dominate. **(c)** Matching at the lowest stage is in progress, and a tendency for the ninth gallery pattern is developing. **(d)** The switchyard has settled to a shift of 8 points, and the Gallery has almost converged to the correct pattern.



(a)

(b)

(c)

Figure 5.9: Signals generated by the Gallery and propagated down the Gallery Assembly stream. **(a)** Initially, the uppermost Gallery Assembly Layer contains an average of all Gallery patterns, while the lower layers only receive lowpass versions of this signal. **(b)** After the links of the highest routing stage have converged, they route a sharp version of the Gallery average to the middle Gallery Assembly. **(c)** In this image, both the routing process and the Gallery competition have mostly finished. The uppermost layer now contains a copy of the chosen Gallery pattern instead of the average of all of them (note the subtle differences). This pattern is precisely routed to the lower stages.

the highest Input Assembly Layer. From there it can serve as point-by-point input to the Gallery, activating more strongly those units that are similar to its own signal. While this similarity is not very crisp initially due to the blurred nature of the signal in the Input Assembly stream, it improves with every routing stage below that finds its correspondences and sends a sharper version of the input pattern upwards. The sharper the pattern in the highest Input Assembly, the clearer becomes the decision among the candidate patterns of the Gallery. This in turn changes the pattern sent down through the Gallery Assembly stream from an average of all Gallery patterns towards that pattern which is dominating the competition. This enhancement of the downstream target pattern is necessary especially for fixing the very fine correspondences at the lowest routing stage. If the recognition process is successful, the whole system ends up in a state where the control units of the switchyard represent the correct shift from Input to Gallery, while the one Gallery pattern remaining active is the one the system has “recognized” as being most similar to the input pattern (Figure 5.8d). Neighbor-neighbor cooperation between parallel links as well as between Gallery units representing the same pattern helps the system to perform robustly on this challenging “hen and egg” problem even in the presence of noise.

5.3 Conclusion

We have seen in this chapter how the ideas of the previous chapters can be combined into a position-invariant recognition system. We first investigated how the single stage correspondence finding of Section 2.1 can be extended to matching patterns via the intermediate stages of a switchyard. We then proceeded to combine these results with the object recognition ideas of Section 2.4. The resulting system combines the advantages of correspondence-based object recognition (see Chapter 1) with the biologically (and computationally) more plausible idea of achieving full connectivity via a switchyard of routing stages.

This claim that a system using several layers for correspondence finding is biologically more realistic is—additionally to the physiological evidence discussed in Section 3.2—also backed by the fact that the approach breaks down for visual patterns that are also specifically challenging for humans. For example, it is quite difficult for humans to recognize *random dot stereograms* (RDSs). RDSs are a superposition of two fine-grained patterns (see Figure 5.10) the difference of which encodes a virtual depth profile of the image. When merged properly, they give rise to a very strong sensation of depth. Computationally, recognition of RDSs is equivalent to merging two normal stereoscopic images, and both these tasks can be formulated as correspondence finding problems between two patterns as discussed throughout this thesis. Consequently, the single stage matching process of Section 2.1 would perform equally well and fast for RDSs as for other visual patterns. The difference between RDSs and “normal” visual information is, however, that the former do not contain any low-frequency cues that can be used to establish coarse correspondences between the two patterns. Therefore, the multi-stage matching process introduced in Section 5.1 would not be able to successfully match the information contained in a random dot stereogram, since it would lack the low-frequency correlations that are necessary to initiate matching at the higher routing stages in the first place. Likewise, many humans cannot recognize these random dot stereograms at all, and if they can, it is a slow process that requires

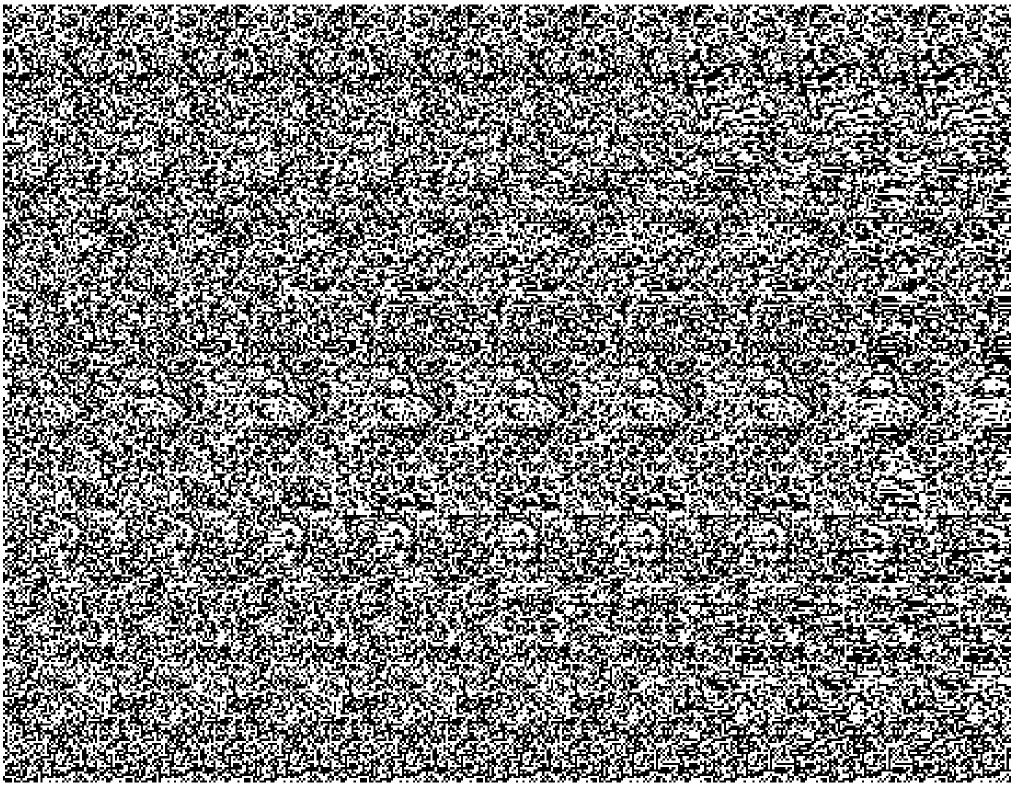


Figure 5.10: Random dot stereogram. When looking at this figure and trying to focus behind it by lightly squinting outwards, one may perceive a three-dimensional shape (in this image a relief of the Mandelbrot set).

patiently staring at the image and waiting for the brain to make sense of the correlations among the random dots.

6 Discussion and Outlook

In the introduction to this thesis, we claimed that the matching of spatially distributed patterns is central to the function of our brain, and we named two requirements for this: suitable connection structures between the units representing those patterns, and a basic mechanism—plus its biological realization—for matching patterns.

We presented this basic principle, Dynamic Link Matching (e.g. Bienenstock and von der Malsburg 1987), in Section 2.1 and showed in the following sections how it can be realized in a neurally plausible way and how it can be extended into a functioning, competitive face recognition system. Regarding connection structures, in Chapter 3 we derived a multi-stage architecture that provides (indirect) full connectivity between two patterns using a minimal number of neural resources. In Chapter 4, we investigated the growth of connections in the brain and found an ontogenetic mechanism that produces in a self-organized, autonomous way connection structures that are nearly identical to the architectures derived in Chapter 3.

But why should the *matching of patterns via intermediate stages* be a general principle of brain function, as we claimed at the beginning of the dissertation? Additionally to the arguments discussed in the introduction, the findings of this dissertation support this conviction in several ways.

First of all, an object recognition system built on these principles is inherently *generative*. It requires that information flows not only upstream from the input, but also downstream from an internal memory (see Figure 5.6), thus generating at "lower" stages explicit representations of its percepts and decisions in higher areas (see e.g. the patterns propagated down the Gallery Assembly stream in Figure 5.9, or the content of the Gallery Assembly in Figure 2.11 changing from an average to the recognized face). This entails that the system of Chapter 2 cannot only detect general faces, but can in a very natural way be biased e.g. for "female" faces or search for a specific individual. The system can exhibit attentional effects through priming of its routing units on the input side, as well as expectation through priming of its memory representations. In that sense, feedforward and feedback streams, external input and internal memory become two equally important sides of the same coin.

Another advantage is the *combinatorial coding of transformations* introduced by having multiple routing stages. When matching two patterns, the circuitry connecting them must be able to carry out a multitude of transformations to map the patterns onto each other in spite of their different extrinsic properties (position, scale, color, etc.). If the patterns are connected directly, then dynamic links between them must be able to express these transformations in a single step. However, if the patterns are connected by a switchyard, then each routing stage only represents a partial transformation, all of which are carried out in series to produce the full mapping. In that sense, multiple stages can generate a huge space of transformations through a combinatorial code of their simple mappings. This is computationally more economic than the direct connectivity case, where every single of these possible transformations needs to be represented explicitly. We saw in Chapter 3 that switchyards, which are capable of performing arbitrary translations as well as many types of scaling and rotation, are much cheaper in terms of required neural circuitry than if these transformations were to be carried out in a single step (cf. Figure 3.2). For realistic vision scenarios with even more types of transformations, it would be indispensable to carry them out in a circuit of sequential stages. Of course, the division of the matching process into several stages has its drawbacks. To allow it to function, the different stages have to be able to operate at least partially independently. If the initial loss of fine-scale spatial information makes the visual input insufficient for a successful matching at higher stages, then the multi-stage approach will not work, as we discussed in Section 5.3. For such problems, the brain probably resorts to serial processing of small patches of the input image at a time, thus effectively trying out all combinations of control units at the different stages and hence negating the advantages of multi-stage routing.

Finally, vision systems based on matching via several stages could offer a synthesis of the feature-based and correspondence-based “philosophies” in vision: Initially, input information flows upstream through the Input Assembly stages with all routing gates open (see Figure 5.4a). This first bottom-up sweep is very similar to the behavior of feature-based systems. If the free flow of information were replaced by a max-like pooling at each receiving feature node, such a system would initially respond similarly to models like those of Riesenhuber and Poggio (1999). The matching process that follows this first sweep, on the other hand, is inherently correspondence-based. When matching has finished on all routing stages, the final situation is given by a complete match across all stages that, different to feature-based approaches, does not lose information about the properties of the input pattern. In that sense, systems built on the principles of Chapter 5 could reconcile the findings of extremely fast, unconscious recognition (Thorpe et al. 1996) with the advantages of correspondence-based or generative systems (very exact recognition, reasoning about a percept, etc.).

Of course, the results of this dissertation can only be a starting point in this direction. Especially the unification in Chapter 5 of the principles derived in the preceding chapters is only demonstrated on toy examples. Apart from dealing with more realistic data here, the system needs to be extended towards being able to recognize images from more than a single category. For object categories with a clear topological layout, this should in principle be possible as sketched in Figure 6.1. The system would first coarsely match the input information at the highest Assembly stage with averages generated from the respective category memories to switch on the dynamic links to only one category. This would result in an average pattern in the top-down Gallery Assembly stream generated only from the chosen category. From then on, matching would proceed as described in Chapter 5 to determine the individual object within that category.

Another unrealistic aspect of the model so far is the flat structure of the gallery domain. As has been convincingly shown (Biederman 1987), many objects are recognized as ordered arrays of simpler sub-shapes. Just like we extended the correspondence-finding part of our system to multi-layered switchyards as opposed to the one-step matching of earlier systems, a hierarchy should therefore be introduced on the Gallery side. This would help the system to deal better with images that are not as strictly spatially defined as faces (e.g. objects made up of sub-components, or landscape scenes). Obviously, hierarchical structuring of a memory domain

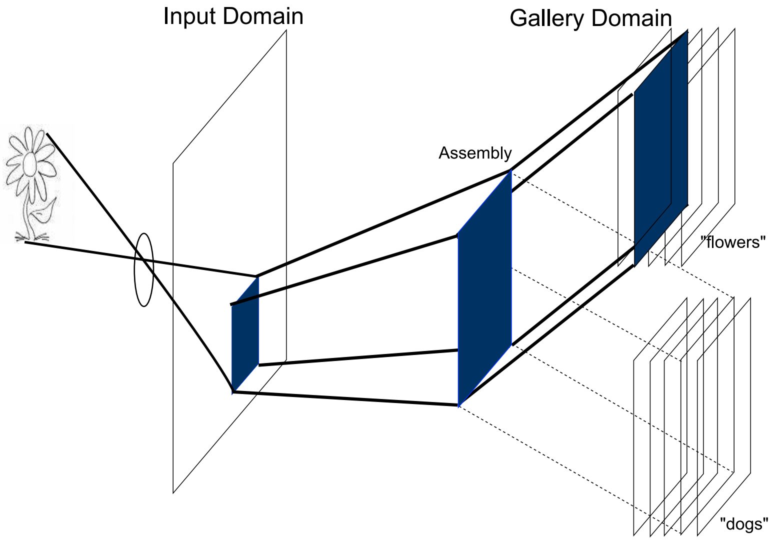


Figure 6.1: Sketch of a system recognizing objects from several categories.

is more challenging than structuring of the correspondence-finding process, where we could exploit the strictly geometric nature of early visual representations.

We hope that the building blocks provided by this dissertation—a biologically plausible approach to recognition by finding correspondences between layers of units and an approach to break pattern matching into a switchyard of several stages—can help to tackle these problems.

Appendix

A Self-Normalization Properties of Columnar Dynamics

The dynamics of the column model introduced in Section 2.3 possess automatic activity normalization, as shown in the following. Since the ratio $\frac{T}{\tau}$ is very large, i.e. the time constant is much shorter than the overall simulation time, the unit activities are close to the adiabatic state, i.e. $\frac{d}{dt}x_i \approx 0$.

For the case of $\nu = 0$ (linear representation), the unit dynamics are given by (2.2):

$$\tau \frac{d}{dt} x_i = I_i - x_i \sum_{j=1}^{K} I_j x_j .$$

The steady state of this is

$$x_i = \frac{I_i}{\sum_j I_j x_j}. \tag{App.1}$$

Multiplying both sides with x_i yields

$$x_i^2 = \frac{I_i x_i}{\sum_j I_j x_j},$$

and the sum of this term over all i is

$$\sum_i x_i^2 = 1, \tag{App.2}$$

i.e. for $\nu = 0$ the column activity is normalized to a 2-norm or Euclidean norm of 1. From squaring (App.1) and summing over all units, we get

$$\sum_i x_i^2 = \frac{\sum_i I_i^2}{(\sum_j I_j x_j)^2},$$

and plugging in the normalization property (App.2) yields

$$\left(\sum_j I_j x_j \right)^2 = \sum_j I_j^2 .$$

Therefore, the steady state of a feature unit becomes independent of the other unit activities in the column and is an explicit function of the column inputs:

$$x_i = \frac{I_i}{\sqrt{\sum_j I_j^2}}. \tag{App.3}$$

In the case of $\nu = 1$ (WTA behavior), the unit dynamics are

$$\tau \frac{d}{dt} x_i = x_i I_i - x_i \sum_{j=1}^{K} I_j x_j,$$

so summing over all units gives

$$\tau \sum_i \frac{d}{dt} x_i = \sum_i x_i I_i - \sum_i x_i \sum_j x_j I_j = \left(1 - \sum_i x_i\right) \sum_i x_i I_i = 0 \tag{App.4}$$

in steady state. If there is any activity in the column, i.e. not all unit activities are zero simultaneously, this requires

$$\sum_i x_i = 1, \tag{App.5}$$

which means that for $\nu = 1$ the column activity self-normalizes to a 1-norm (also Manhattan norm in this case, since unit activities are strictly positive) of 1. Consequently, the interaction term $\sum_j I_j x_j$ is the average activity-weighted input to the column. This means that only those unit activities grow whose input is higher than this weighted mean input to the column, otherwise they decay. This lets the weighted input average grow, because the bias shifts towards strong inputs. Eventually, all unit activities decrease to 0 except for the unit with the strongest input, whose activity approaches 1. For the final steady state we can show this by setting the time derivative for a single unit to 0:

$$x_i \left(I_i - \sum_{j=1}^{K} I_j x_j \right) = 0. \tag{App.6}$$

Here we see that for any i, we either have $x_i = 0$, or $I_i = \sum_j I_j x_j$, which can only be true for $x_i = 1$ and *all* other $x_j = 0$ (except for the degenerate case of two or more of the I_i being exactly identical, which would presumably be solved in the brain by spontaneous symmetry breaking). So in the final state, one unit will have activity 1 with all other unit activities at 0.

B Gabor Transform

The model described in Chapter 2 represents input images by a grid of Gabor features. At each grid point, it uses a set of Gabor wavelets that appropriately sample orientation (over 8 orientations) and spatial frequency (over 5 scales) space. If V is an image with $V(\vec{z})$ denoting the gray-value of a pixel at the geometric position $\vec{z} = \binom{x}{y}$, the filter responses $R_i\left(\vec{z}\right)$ are given by:

$$\begin{aligned} R_i\left(\vec{z}\right) &= \int V\left(\vec{z}\,'\right)\psi_i(\vec{z} - \vec{z}\,')d^2\vec{z}\,' \,, && \text{(App.7)} \\ \psi_i\left(\vec{\zeta}\right) &= \frac{k_i^2}{\sigma^2}\ \exp\left(-\frac{k_i^2\zeta^2}{2\sigma^2}\right)\left[\exp\left(i\vec{k}_i\vec{\zeta}\right) - \exp\left(-\tfrac{\sigma^2}{2}\right)\right], && \text{(App.8)} \end{aligned}$$

where we choose

$$\sigma = 2\pi \tag{App.9}$$

to approximate the shape of receptive fields found in primary visual cortex, and where the wave vector is parameterized as

$$\vec{k}_i = \begin{pmatrix} k_{ix} \\ k_{iy} \end{pmatrix} = \begin{pmatrix} k_\rho \cos\varphi_\mu \\ k_\rho \sin\varphi_\mu \end{pmatrix}, \quad k_\rho = 2^{\left(-\frac{\rho+2}{2}\right)}\pi, \quad \varphi_\mu = \mu\tfrac{\pi}{8}, \tag{App.10}$$

with orientation parameter $\mu = 1, .., 8$ and scale parameter $\rho = 1, .., 5$. That is, $(R_1(\vec{z}), \ldots, R_{40}(\vec{z}))$ is a vector of Gabor filter responses in which each entry corresponds to one of the 40 combinations of ρ and μ. As feature values we use the magnitude

$$\mathcal{J}_i^p = |R_i(\vec{z}_p)|, \tag{App.11}$$

thus ignoring Gabor phase, to model complex cell responses (Hubel and Wiesel 1977).

Bibliography

Adler, A. and Schuckers, M. E.: 2007, Comparing human and automatic face recognition performance, *IEEE Trans Syst Man Cybern B Cybern* **37**(5), 1248–1255.

Ajtai, M., Komlós, J. and Szemerédi, E.: 1983, An 0(n log n) sorting network, *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pp. 1–9.

Anderson, C. H. and Van Essen, D. C.: 1987, Shifter circuits: A computational strategy for dynamic aspects of visual processing, *Proceedings of the National Academy of Sciences of the United States of America* **84**, 6297–6301.

Arathorn, D. W.: 2002, *Map-Seeking Circuits in Visual Cognition—A Computational Mechanism for Biological and Machine Vision*, Stanford University Press.

Bak, P.: 1996, *How nature works: the science of self-organized criticality*, Springer-Verlag.

Bar, M. and Biederman, I.: 1999, Localizing the cortical region mediating visual awareness of object identity, *PNAS* **96**, 1790–179.

Berg, A. C., Berg, T. L. and Malik, J.: 2005, Shape matching and object recognition using low distortion correspondence, *Proc. CVPR*, pp. 26–33.

Biederman, I.: 1987, Recognition-by-components: a theory of human image understanding, *Psychol Rev* **94**(2), 115–147.

Biederman, I. and Kalocsai, P.: 1997, Neurocomputational bases of object and face recognition, *Phil. Trans. Roy. Soc. B* **352**, 1203–1219.

Bienenstock, E. and von der Malsburg, C.: 1987, A neural network for invariant pattern recognition, *Europhysics Letters* **4**(1), 121–126.

Bosch, A., Zisserman, A. and MuÃśoz, X.: 2008, Scene classification using a hybrid generative/discriminative approach, *IEEE Trans Pattern Anal Mach Intell (PAMI)* **30**, 712–727.

Bundesen, C. and Larsen, A.: 1975, Visual transformation of size, *J Exp Psychol Hum Percept Perform* **1**(3), 214–220.

Buxhoeveden, D. P. and Casanova, M. F.: 2002, The minicolumn hypothesis in neuroscience, *Brain* **125**, 935–951.

Chater, N., Tenenbaum, J. B. and Yuille, A.: 2006, Probabilistic models of cognition: Conceptual foundations, *Trends in Cognitive Sciences* **10**(7), 287–291.

Cherniak, C.: 1990, The bounded brain: Toward quantitative neuroanatomy, *Journal of Cognitive Neuroscience* **2**(1), 58–68.

Connor, C. E., Gallant, J. L., Preddie, D. C. and van Essen, D. C.: 1993, Responses in area v4 depend on the spatial relationship between stimulus and attention, *Journal of Neurophysiology* **75**, 1306–1308.

Cooley, J. W. and Connor, J. W. C. E.: 1965, An algorithm for the machine calculation of complex fourier series, *Mathematics of Computation* **19**, 297–301.

Cox, D., Meier, P., Oertelt, N. and DiCarlo, J. J.: 2005, 'breaking' position-invariant object recognition, *Nature Neuroscience* **8**(9), 1145–1147.

Csurka, G., Dance, C. R., Fan, L., Willamowski, J. and Bray, C.: 2004, Visual categorization with bags of keypoints, *Proc. ECCV workshop on Statistical Learning in Computer Vision*, pp. 59–74.

Dantzker, J. L. and Callaway, E. M.: 2000, Laminar sources of synaptic input to cortical inhibitory interneurons and pyramidal neurons, *Nat Neurosci* **3**(7), 701–707.

Daugman, J. G.: 1980, Two-dimensional spectral analysis of cortical receptive field profiles, *Vision Res* **20**, 847–856.

Dayan, P., Hinton, G. E., Neal, R. M. and Zemel, R. S.: 1995, The helmholtz machine, *Neural Comput* **7**(5), 889–904.

Debruille, J. B., Guillem, F. and Renault, B.: 1998, Erps and chronometry of face recognition: following-up seeck et al. and george et al, *Neuroreport* **9**(15), 3349–3353.

Deco, G. and Rolls, E. T.: 2004, A neurodynamical cortical model of visual attention and invariant object recognition, *Vision Res* **44**(6), 621–642.

DeFelipe, J., Hendry, M. C. and Jones, E. G.: 1989, Synapses of double bouquet cells in monkey cerebral cortex, *Brain Research* **503**, 49–54.

Desimone, R., Wessinger, M., Thomas, L. and Schneider, W.: 1990, Attentional control of visual perception: cortical and subcortical mechanisms, *Cold Spring Harb. Symp. Quant. Biol.* **55**, 963–971.

Dill, M. and Fahle, M.: 1998, Limited translation invariance of human visual pattern recognition, *Perception and Psychophysics* **60**(1), 65–81.

Dougherty, R. F., Koch, V. M., Brewer, A. A., Fischer, B., Modersitzki, J. and Wandell, B. A.: 2003, Visual field representations and locations of visual areas v1/2/3 in human visual cortex, *Journal of Vision* **3**, 586–598.

Douglas, R. J. and Martin, K. A.: 2004, Neuronal circuits of the neocortex, *Annual Review of Neuroscience* **27**, 419–451.

Douglas, R. J., Martin, K. A. and Witteridge, D.: 1989, A canonical microcircuit for neocortex, *Neural Computation* **1**, 480–488.

Duda, R., Hart, E. and Stork, D.: 2001, *Pattern Classification*, 2nd edn, John Wiley and Sons.

Duhamel, J., Bremmer, F., BenHamed, S. and Graf, W.: 1997, Spatial invariance of visual receptive fields in parietal cortex neurons, *Nature* **389**, 845–848.

Duhamel, J. R., Colby, C. L. and Goldberg, M. E.: 1992, The updating of the representation of visual space in parietal cortex by intended eye movements, *Science* **255**(5040), 90–92.

Duncan, J.: 1984, Selective attention and the organization of visual information, *J Exp Psychol Gen* **113**, 501–517.

Durbin, R. and Rumelhart, D. E.: 1989, Product units: A computationally powerful and biologically plausible extension to backpropagation networks, *Neural Computation* **1**(1), 133–142.

Eigen, M.: 1971, Selforganization of matter and the evolution of biological macromolecules, *Naturwissenschaften* **58**, 465–523.

Elston, G. N. and Rosa, M. G.: 1998, Morphological variation of layer iii pyramidal neurones in the occipitotemporal pathway of the macaque monkey visual cortex, *Cereb Cortex* **8**(3), 278–294.

Elston, G. N. and Rosa, M. G.: 2000, Pyramidal cells, patches, and cortical columns: a comparative study of infragranular neurons in teo, te, and the superior temporal polysensory area of the macaque monkey, *J Neurosci* **20**(24), RC117.

Erdös, P. and Rényi, A.: 1959, On random graphs, *Publicationes Mathematicae* **6**, 290–297.

Favorov, O. V. and Diamond, M.: 1990, Demonstration of discrete place-defined columns, segregates, in cat SI, *Journal of Comparative Neurology* **298**, 97 – 112.

Favorov, O. V. and Kelly, D. G.: 1994, Minicolumnar organization within somatosensory cortical segregates II, *Cerebral Cortex* **4**, 428 – 442.

Fei-Fei, L., Fergus, R. and Perona, P.: 2003, A bayesian approach to unsupervised one-shot learning of object categories, *Proc. of the Ninth IEEE Intern. Conf. Computer Vision*, pp. 1134–1141.

Feldman, J. A.: 1982, Dynamic connections in neural networks, *Biol Cybern* **46**(1), 27–39.

Fergus, R., Perona, P. and Zisserman, A.: 2003, Object class recognition by unsupervised scale-invariant learning, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*.

Fiser, J. and Biederman, I.: 2001, Invariance of long-term visual priming to scale, reflection, translation, and hemisphere, *Vision Research* **41**, 221–234.

Freund, T., Bolam, J., BjÃűrklund, A., Stenevi, U., Dunnett, S., Powell, J. and Smith, A.: 1985, Efferent synaptic connections of grafted dopaminergic neurons reinnervating the host neostriatum: a tyrosine hydroxylase immunocytochemical study, *J. Neurosci.* **5**, 603–616.

Fukuda, Y., Sugimoto, T. and Shirokawa, T.: 1982, Strain differences in quantitative analysis of rat optic nerve, *Experimental neurology* **75**, 525–532.

Fukushima, K., Miyake, S. and Ito, T.: 1983, Neocognitron: A neural network model for a mechanism of visual pattern recognition, *IEEE Transactions on Systems, Man and Cybernetics* **13**(5), 826–834.

Gabbiani, F., Krapp, H. G., Koch, C. and Laurent, G.: 2002, Multiplicative computation in a visual neuron sensitive to looming, *Nature* **420**, 320–324.

Gauthier, I., Skudlarski, P., Gore, J. C. and Anderson, A. W.: 2000, Expertise for cars and birds recruits brain areas involved in face recognition, *Nat Neurosci* **3**(2), 191–197.
URL: *http://dx.doi.org/10.1038/72140*

Gerstner, W.: 2000, Population dynamics of spiking neurons: fast transients, asynchronous states, and locking, *Neural Computation* **12**(1), 43–89.

Graf, M.: 2006, Coordinate transformations in object recognition, *Psychol Bull* **132**, 920–945.

Gray, C. M. and Singer, W.: 1989, Stimulus-specific neuronal oscillations in orientation columns of cat visual cortex, *Proceedings of the National Academy of Sciences of the USA* **86**(5), 1698–1702.

Greenberg, D. S., Houweling, A. R. and Kerr, J. N. D.: 2008, Population imaging of ongoing neuronal activity in the visual cortex of awake rats, *Nat Neurosci* .
URL: *http://dx.doi.org/10.1038/nn.2140*

Häussler, A. and von der Malsburg, C.: 1983, Development of retinotopic projections: an analytic treatment, *J. Theoret. Neurobiol.* **2**, 47–73.

Holt, C. and Harris, W.: 1993, Position, guidance, and mapping in the developing visual system, *J. Neurobiol.* **24**, 1400–1422.

Hubel, D. H. and Wiesel, T. N.: 1977, Functional architecture of macaque visual cortex, *Proceedings of the Royal Society of London - B* **198**, 1 – 59.

Hughes, A. and Wässle, H.: 1976, The cat optic nerve: Fibre total count and diameter spectrum, *The Journal of Comparative Neurology* **169**, 171–184.

Humphreys, G. and Heinke, D.: 1998, Spatial representation and selection in the brain: Neuropsychological and computational constraints, *Visual cognition* **5**, 9–47.

Hung, C. P., Kreiman, G., Poggio, T. and DiCarlo, J. J.: 2005, Fast readout of object identity from macaque inferior temporal cortex, *Science* **310**(5749), 863–866.
URL: *http://dx.doi.org/10.1126/science.1117593*

Joachims, T.: 1998, Text categorization with support vector machines: Learning with many relevant features, *Proc. ECML-98, 10th European Conference on Machine Learning*, pp. 137–142.

Johnson, J. S. and Olshausen, B. A.: 2003, Timecourse of neural signatures of object recognition, *J Vis* **3**(7), 499–512.
URL: *http://dx.doi.org/10:1167/3.7.4*

Jolicoeur, P.: 1985, The time to name disoriented natural objects, *Mem Cognit* **13**(4), 289–303.

Jones, E. G.: 2000, Microcolumns in the cerebral cortex, *Proceedings of the National Academy of Sciences, USA* **97**, 5019 – 5021.

Jones, J. and Palmer, L.: 1987, An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex, *Journal of Neurophysiology* **58**, 1233–1258.

Kanwisher, N.: 2006, Neuroscience. what's in a face?, *Science* **311**(5761), 617–618.
URL: *http://dx.doi.org/10.1126/science.1123983*

Kanwisher, N. and Yovel, G.: 2006, The fusiform face area: a cortical region specialized for the perception of faces, *Phil. Trans. R. Soc. B* **361**, 2109–2128.

Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P.: 1983, Optimization by simulated annealing, *Science* **220**(4598), 671–680.
URL: *citeseer.ist.psu.edu/kirkpatrick83optimization.html*

Knuth, D.: 1997, *The Art of Computer Programming*, Vol. 3, Addison-Wesley, chapter 5.3.4: Networks for Sorting, pp. 219–247.

Koch, C.: 1999, *Biophysics of computation: information processing in single neurons*, Oxford University Press.

Konen, C. S. and Kastner, S.: 2008, Two hierarchically organized neural systems for object information in human visual cortex, *Nat Neurosci* **11**(2), 224–231.
URL: *http://dx.doi.org/10.1038/nn2036*

Körner, E., Gewaltig, M.-O., Körner, U., Richter, A. and Rodemann, T.: 1999, A model of computation in neocortical architecture, *Neural Networks* **12**, 989–1005.

Kschischang, F. R., Frey, B. J. and Loelinger, H.-A.: 2001, Factor graphs and the sum-product algorithm, *IEEE Transactions on Information Theory* **47**, 498–519.

Kubota, Y., Hatada, S., Kondo, S., Karube, F. and Kawaguchi, Y.: 2007, Neocortical inhibitory terminals innervate dendritic spines targeted by thalamocortical afferents, *J Neurosci* **27**(5), 1139–1150. Comparative Study.

Kusunoki, M. and Goldberg, M. E.: 2003, The time course of perisaccadic receptive field shifts in the lateral intraparietal area of the monkey, *J Neurophysiol* **89**(3), 1519–1527.

Lades, M., Vorbrüggen, J., Buhmann, J., Lange, J., von der Malsburg, C., Würtz, R. and Konen, W.: 1993, Distortion invariant object recognition in the dynamic link architecture, *IEEE Transactions on computers* **42**, 300–311.

Lamme, V.: 2003, Why visual attention and awareness are different, *Trends Cogn Sci* **7**(1), 12–18.

Larkum, M., Zhu, J. and Sakmann, B.: 1999, A new cellular mechanism for coupling inputs arriving at different cortical layers, *Nature* **398**, 338–341.

Lawson, R. and Jolicoeur, P.: 1999, The effect of prior experience on recognition thresholds for plane-disoriented pictures of familiar objects, *Mem Cognit* **27**(4), 751–758.

Lazebnik, S., Schmid, C. and Ponce, J.: 2003, Affine-invariant local descriptors and neighborhood statistics for texture recognition, *Proc. ICCV*, pp. 649–655.

Lazebnik, S., Schmid, C. and Ponce, J.: 2006, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, *Proc. CVPR*, Vol. 2, pp. 2169–2178.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. and Jackel, L. D.: 1989, Backpropagation applied to handwritten zip code recognition, *Neural Computation* **1**(4), 541–551.

LeCun, Y., Huang, F. J. and Bottou, L.: 2004, Learning methods for generic object recognition with invariance to pose and lighting, *CVPR*, IEEE Computer Society, pp. 97–104.

Leung, T. and Malik, J.: 2001, Representing and recognizing the visual appearance of materials using three-dimensional textons, *International Journal of Computer Vision* **43**, 29–44.

Linsker, R.: 1986, From basic network principles to neural architecture (series), *Proceedings of the National Academy of Sciences* **83**, 7508–12, 8390–4, 8779–83.

Luck, S. J., Chelazzi, L., Hillyard, S. A. and Desimone, R.: 1997, Neural mechanisms of spatial selective attention in areas V1, V2, and V4 of macaque visual cortex, *J Neurophysiol* **77**(1), 24–42.

Lücke, J.: 2005, *Information Processing and Learning in Networks of Cortical Columns*, PhD thesis, Ruhr-University Bochum.

Lücke, J., Keck, C. and von der Malsburg, C.: 2008, Rapid convergence to feature layer correspondences, *Neural Computation* **20**(10), 2441–2463.

Lücke, J. and von der Malsburg, C.: 2004, Rapid processing and unsupervised learning in a model of the cortical macrocolumn, *Neural Computation* **16**, 501 – 533.

Lund, J. S., Yoshioka, T. and Levitt, J. B.: 1993, Comparison of intrinsic connectivity in different areas of macaque monkey cerebral cortex, *Cereb Cortex* **3**(2), 148–162.

Marr, D. and Poggio, T.: 1976, Cooperative computation of stereo disparity, *Science* **194**(4262), 283–287.

Martinez, A. and Benavente, R.: 1998, The AR face database, *Technical Report 24*, CVC.

Mel, B. W.: 1997, Seemore: Combining color, shape, and texture histogramming in a neurally inspired approach to visual object recognition, *Neural Computation* **9**, 777–804.

Mel, B. W. and Fiser, J.: 2000, Minimizing binding errors using learned conjunctive features, *Neural Computation* **12**(4), 731–762.

Messer, K., Kittler, J., Sadeghi, M., Hamouz, M., Kostin, A., Cardinaux, F., Marcel, S., Bengio, S., Sanderson, C., Poh, N., Rodriguez, Y., Czyz, J., Vandendorpe, L., McCool, C., Lowther, S., Sridharan, S., Chandran, V., Palacios, R. P., Vidal, E., Bai, L., Shen, L., Wang, Y., Yueh-Hsuan, C., Hsien-Chang, L., Yi-Ping, H., Heinrichs, A., Müller, M., Tewes, A., von der Malsburg, C., Würtz, R., Wang, Z., Xue, F., Ma, Y., Yang, Q., Fang, C., Ding, X., Lucey, S., Goss, R. and Schneiderman, H.: 2004, Face authentication test on the BANCA database, *Proceedings of the International Conference on Pattern Recognition, Cambridge*, Vol. 4, pp. 523 – 532.

Moran, J. and Desimone, R.: 1985, Selective attention gates visual processing in the extrastriate cortex, *Science* **229**, 782–784.

Morton, N.: 1991, Parameters of the human genome, *Proc. Natl. Acad. Sci. U.S.A.* **88**, 7474–7476.

Mountcastle, V. B.: 1997, The columnar organization of the neocortex, *Brain* **120**, 701–722.

Mountcastle, V. B.: 2003, Introduction (to a special issue on cortical columns), *Cerebral Cortex* **13**, 2 – 4.

Muresan, R. C. and Savin, C.: 2007, Resonance or integration? Self-sustained dynamics and excitability of neural microcircuits, *Journal of Neurophysiology* **97**, 1911 – 1930.

Murray, J. F. and Kreutz-Delgado, K.: 2007, Visual recognition and inference using dynamic overcomplete sparse learning, *Neural Comput* **19**(9), 2301–2352.
URL: *http://dx.doi.org/10.1162/neco.2007.19.9.2301*

Murray, S. O., Boyaci, H. and Kersten, D.: 2006, The representation of perceived angular size in human primary visual cortex, *Nature Neuroscience* **9**, 429–434.

Nakayama, K. and Silverman, G. H.: 1986, Serial and parallel processing of visual feature conjunctions, *Nature* **320**(6059), 264–265.
URL: *http://dx.doi.org/10.1038/320264a0*

Nicola, S., Surmeier, J. and Malenka, R.: 2000, Dopaminergic modulation of neuronal excitability in the striatum and nucleus accumbens, *Annu. Rev. Neurosci.* **23**, 185–215.

Obermayer, K. and Blasdel, G. G.: 1997, Singularities in primate orientation maps, *Neural Computation* **9**, 555–575.

Oliva, A. and Torralba, A.: 2006, Building the gist of a scene: the role of global image features in recognition, *Prog Brain Res* **155**, 23–36.
URL: *http://dx.doi.org/10.1016/S0079-6123(06)55002-2*

Olshausen, B.: 1994, *Neural Routing Circuits for Forming Invariant Representations of Visual Objects*, PhD thesis, California Institute of Technology.

Olshausen, B. A., Anderson, C. H. and van Essen, D. C.: 1993, A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information, *Journal of Neuroscience* **13**(11), 4700–4719.

Olshausen, B. A. and Field, D. J.: 1997, Sparse coding with an overcomplete basis set: a strategy employed by v1?, *Vision Research* **37**, 3311–3325.

Olson, M. and Meyer, R.: 1991, The effect of TTX-activity blockade and total darkness on the formation of retinotopy in the goldfish retinotectal projection, *J. Comp. Neurol.* **303**, 412–423.

Oram, M. W. and Perret, D. I.: 1994, Modeling visual recognition from neurobiological constraints, *Neural Networks* **7**, 945–972.

Perkel, D. J., Bullier, J. and Kennedy, H.: 1986, Topography of the afferent connectivity of area 17 in the macaque monkey: A double-labelling study, *The Journal of Comparative Neurology* **253**, 374–402.

Peters, A., Cifuentes, J. M. and Sethares, C.: 1997, The organization of pyramidal cells in area 18 of the rhesus monkey, *Cerebral Cortex* **7**, 405 – 421.

Peters, A. and Yilmaz, E.: 1993, Neuronal organization in area 17 of cat visual cortex, *Cerebral Cortex* **3**, 49 – 68.

Phillips, P., Flynnand, P., Scruggs, T., Bowyer, K., Chang, J., Hoffman, K., Marques, J., Min, J. and Worek, W.: 2005, Overview of the face recognition grand challenge, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 947–954.

Phillips, P., Grother, P., Micheals, R., Blackburn, D., Tabassi, E., and Bone, J.: 2003, Frvt 2002 evaluation report, *Technical Report 6965*, NISTIR. http://www.frvt.org/.

Phillips, P. J., Wechsler, H., Huang, J. and Rauss, P. J.: 1998, The FERET database and evaluation procedure for face recognition algorithms, *Image and Vision Computing* **16**(5), 295–306.

Phillips, P., Moon, H., Rizvi, S. and Rauss, P.: 2000, The FERET evaluation methodology for face-recognition algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(10), 1090–1104.

Pinto, N., Cox, D. D. and Dicarlo, J. J.: 2008, Why is real-world visual object recognition hard?, *PLoS Computational Biology* **4**(1), e27+.
URL: *http://dx.doi.org/10.1371/journal.pcbi.0040027*

Pitts, W. and McCulloch, W. S.: 1947, How we know universals: the perception of auditory and visual forms, *Bulletin of Mathematical Biophysics* **9**, 127–147.

Pollen, D., Lee, J. and Taylor, J.: 1971, How does the striate cortex begin the reconstruction of the visual world?, *Science* **173**, 74–77.

Postma, E., van den Herik, H. and Hudson, P.: 1997, SCAN: A Scalable Model of Attentional Selection, *Neural Netw* **10**(6), 993–1015.

Potts, A. M., Hodges, D., Shelman, C. B., Fritz, K. J., Levy, N. S. and Mangnall, Y.: 1972, Morphology of the primate optic nerve. i. method and total fiber count, *Investigative Ophthalmology & Visual Science* **11**, 980–988.

Pouget, A. and Sejnowski, T. J.: 1997, Spatial transformations in the parietal cortex using basis functions, *Journal of Cognitive Neuroscience* **9**(2), 222–237.

Rager, G. and Rager, U.: 1978, Systems-matching by degeneration i. a quantitative electron microscopic study of the generation and degeneration of retinal ganglion cells in the chicken, *Experimental Brain Research* **33**, 65–78.

Rao, R. P. and Ballard, D. H.: 1999, Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects, *Nat Neurosci* **2**(1), 79–87.

Riesenhuber, M. and Poggio, T.: 1999, Hierarchical models of object recognition in cortex, *Nat Neurosci* **2**(11), 1019–1025.

Ringach, D. L.: 2002, Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex, *Journal of Neurophysiology* **88**, 455 – 463.

Rockland, K. S. and Ichinohe, N.: 2004, Some thoughts on cortical minicolumns, *Experimental Brain Research* **158**, 265–277.

Rosenblatt, F.: 1961, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Spartan Books, Washington, D.C.

Rosoff, W. J., Urbach, J. S., Esrick, M. A., McAllister, R. G., Richards, L. J. and Goodhill, G. J.: 2004, A new chemotaxis assay shows the extreme sensitivity of axons to molecular gradients, *Nature Neuroscience* **7**, 678 – 682.

Salinas, E. and Sejnowski, T. J.: 2001, Gain modulation in the central nervous system: where behavior, neurophysiology, and computation meet, *Neuroscientist* **7**(5), 430–440.

Sato, Y. D., Jitsev, J. and von der Malsburg, C.: 2008, A visual object recognition system invariant to scale and rotation, *ICANN (accepted)*.

Sato, Y. D., Wolff, C., Wolfrum, P. and von der Malsburg, C.: 2007, Dynamic link matching between feature columns for different scale and orientation, *Proc. ICONIP 2007, Part I*, LNCS 4984, Springer, pp. 385–394.

Schaefer, A. T., Larkum, M. E., Sakmann, B. and Roth, A.: 2003, Coincidence detection in pyramidal neurons is tuned by their dendritic branching pattern, *J Neurophysiol* **89**(6), 3143–3154.
URL: *http://dx.doi.org/10.1152/jn.00046.2003*

Schiele, B. and Crowley, J.: 2000, Recognition without correspondence using multidimensional receptive field histograms, *International Journal of Computer Vision* **36**(1), 31–50.

Schmidt, J. and Buzzard, M.: 1993, Activity-driven sharpening of the retinotectal projection in goldfish: development under stroboscopic illumination prevents sharpening, *J. Neurobiol.* **24**, 384–399.

Schmitt, A., Shi, J., Wolf, A., Lu, C., King, L. and Zou, Y.: 2006, Wnt-Ryk signalling mediates medial-lateral retinotectal topographic mapping, *Nature* **439**, 31–37.

Schwartz, E. L.: 1977, Spatial mapping in primate sensory projection: analytic structure and relevance to perception, *Biological Cybernetics* **25**, 181–194.

Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M. and Poggio, T.: 2007, Robust object recognition with cortex-like mechanisms, *IEEE Trans Pattern Anal Mach Intell* **29**(3), 411–426. Evaluation Studies.

Simons, D. and Rensink, R.: 2005, Change blindness: past, present, and future, *Trends Cogn. Sci. (Regul. Ed.)* **9**, 16–20.

Singer, W.: 2003, Synchronization, binding and expectancy, *in* M. Arbib (ed.), *The handbook of brain theory and neural networks*, MIT Press, pp. 1136 – 1143.

Sivic, J., Russell, B., Efros, A., Zisserman, A. and Freeman, W.: 2005, Discovering objects and their location in images, *Proc. ICCV*, pp. 370–377.

Song, Y., Goncalves, L. and Perona, P.: 2003, Unsupervised learning of human motion, *PAMI* **25**(25), 1–14.

Sperry, R. W.: 1963, Chemoaffinity in the orderly growth of nerve fiber patterns and connections, *Proc Natl Acad Sci U S A* **50**, 703–710.

Summerfield, C., Egner, T., Greene, M., Koechlin, E., Mangels, J. and Hirsch, J.: 2006, Predictive codes for forthcoming perception in the frontal cortex, *Science* **314**(5803), 1311–1314.

Swain, M. and Ballard, D.: 1991, Color indexing, *International Journal of Computer Vision* **7**, 11–32.

Tal, D. and Schwartz, E. L.: 1997, Computing with the leaky integrate-and-fire neuron: Logarithmic computation and multiplication, *Neural Computation* **9**, 305–318.

Tan, X., Chen, S., Zhou, Z.-H. and Zhang, F.: 2005, Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft kNN ensemble, *IEEE Transactions on Neural Networks* **16**(4), 875– 886.

Tanaka, K.: 1996, Inferotemporal cortex and object vision, *Annu Rev Neurosci* **19**, 109–139.

Tanaka, K.: 2003, Columns for complex visual object features in the inferotemporal cortex: clustering of cells with similar but slightly different stimulus selectivities, *Cereb Cortex* **13**(1), 90–99.

Tanaka, K., Fujita, I., Kobatake, E., Cheng, K. and Ito, M.: 1993, Serial processing of visual object-features in the posterior and anterior parts of the inferotemporal cortex, *in* T. Ono, L. R. Squire, M. E. Raichle, D. Perrett and M. Fukuda (eds), *Brain mechanisms of perception and memory: From neuron to behaviour*, Oxford University Press, pp. 34–46.

Tanigawa, H., Wang, Q. and Fujita, I.: 2005, Organization of horizontal axons in the inferior temporal cortex and primary visual cortex of the macaque monkey, *Cerebral Cortex* **15**, 1887–1899.

Tarr, M. J. and Gauthier, I.: 2000, Ffa: a flexible fusiform area for subordinate-level visual processing automatized by expertise, *Nat Neurosci* **3**(8), 764–769.
URL: *http://dx.doi.org/10.1038/77666*

Thornton, T. L. and Gilden, D. L.: 2007, Parallel and serial processes in visual search, *Psychol Rev* **114**(1), 71–103.

Thorpe, S.: 1988, Identification of rapidly presented images by the human visual system, *Perception* **17**, A77.

Thorpe, S., Fize, D. and Marlot, C.: 1996, Speed of processing in the human visual system, *Nature* **381**(6582), 520–522.

Torralba, A., Murphy, K. P., Freeman, W. T. and Rubin, M. A.: 2003, Context-based vision system for place and object recognition, *Proc. ICCV*, pp. 273–280.

Treisman, A. and Sato, S.: 1990, Conjunction search revisited, *J Exp Psychol Hum Percept Perform* **16**(3), 459–478.

Troncoso, E., Muller, D., Korodi, K., Steimer, T., Welker, E. and Kiss, J. Z.: 2004, Recovery of evoked potentials, metabolic activity and behavior in a mouse model of somatosensory cortex lesion: role of the neural cell adhesion molecule (ncam), *Cereb Cortex* **14**(3), 332–341.

Tsao, D. Y., Freiwald, W. A., Tootell, R. B. H. and Livingstone, M. S.: 2006, A cortical region consisting entirely of face-selective cells, *Science* **311**, 670–674.

Valois, R. L. D. and Valois, K. K. D.: 1990, *Spatial Vision*, Oxford Psychology Series, Oxford Universitz Press.

van Essen, D. C., Olshausen, B., Anderson, C. H. and Gallant, J.: 1991, Pattern recognition, attention, and information bottlenecks in the primate visual system, *in* B. Mathur and C. Koch (eds), *Proceedings of the SPIE Conference on Visual Information Processing: from neurons to chips*, Vol. 1473, pp. 17–28.

van Vreeswijk, C. and Sompolinsky, H.: 1998, Chaotic balanced state in a model of cortical circuits, *Neural Computation* **10**, 1321–1372.

Volgushev, M., Vidyasagar, T. and Pei, X.: 1996, A linear model fails to predict orientation selectivity of cells in the cat visual cortex, *J. Physiol. (Lond.)* **496 (Pt 3)**, 597–606.

von der Malsburg, C.: 1981, The correlation theory of brain function, *Internal report, 81-2*, Max-Planck-Institut für Biophysikalische Chemie, Postfach 2841, 3400 Göttingen, FRG. Reprinted in E. Domany, J.L. van Hemmen, and K.Schulten, editors, *Models of Neural Networks II*, chapter 2, pages 95–119. Springer, Berlin, 1994.

Wang, D.: 2005, The time dimension for scene analysis, *IEEE Transactions on Neural Networks* **16**(6), 1401–1426.

Weber, C., Ritter, H., Cowan, J. and Obermayer, K.: 1997, Development and regeneration of the retinotectal map in goldfish: A computational study, *Phil. Trans. Roy. Soc. Lond. B.* **352**, 1603–1623.

Weber, C. and Wermter, S.: 2007, A self-organizing map of sigma-pi units, *Neurocomputing* **70**(13-15), 2552–60.

Wersing, H. and Körner, E.: 2003, Learning optimized features for hierarchical models of invariant object recognition, *Neural Comput* **15**(7), 1559–1588.
URL: *http://dx.doi.org/10.1162/089976603321891800*

Willshaw, D. J. and von der Malsburg, C.: 1979, A marker induction mechanism for the establishment of ordered neural mappings: its application to the retinotectal problem, *Philos Trans R Soc Lond B Biol Sci* **287**(1021), 203–243.

Wilson, H. R. and Cowan, J. D.: 1973, A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue, *Kybernetik* **13**, 55 – 80.

Wiskott, L.: 1999, The role of topographical constraints in face recognition, *Pattern Recognition Letters* **20**(1), 89–96.

Wiskott, L., Fellous, J.-M., Krüger, N. and von der Malsburg, C.: 1997, Face recognition by elastic bunch graph matching, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **19**(7), 775–779.
URL: *http://www.cnl.salk.edu/ wiskott/Abstracts/WisFelKrue97a.html*

Wiskott, L. and von der Malsburg, C.: 1996, Face recognition by dynamic link matching, *in* J. Sirosh, R. Miikkulainen and Y. Choe (eds), *Lateral Interactions in the Cortex: Structure and Function*, The UTCS Neural Networks Research Group, Austin, TX, http://www.cs.utexas.edu/users/nn/web-pubs/htmlbook96/, chapter 11. Electronic book, ISBN 0-9647060-0-8.
URL: *http://www.cnl.salk.edu/ wiskott/Abstracts/WisMal96c.html*

Wolfrum, P., Lücke, J. and von der Malsburg, C.: 2008, Invariant face recognition in a network of cortical columns, *Proc. International Conference on Computer Vision Theory and Applications*, Vol. 2, pp. 38–45.

Wolfrum, P. and von der Malsburg, C.: 2007a, A marker-based model for the ontogenesis of routing circuits, *Artificial Neural Networks – ICANN 2007*, Vol. 4669 of *LNCS*, Springer, pp. 1–8.

Wolfrum, P. and von der Malsburg, C.: 2007b, What is the optimal architecture for visual information routing?, *Neural Computation* **19**(12), 3293–3309.
URL: *http://dx.doi.org/10.1162/neco.2007.19.12.3293*

Wolfrum, P. and von der Malsburg, C.: 2008, Attentional processes in correspondence-based object recognition, *Proc. COSYNE*, p. 330.

Wolfrum, P., Wolff, C., Lücke, J. and von der Malsburg, C.: 2008, A recurrent dynamic model for correspondence-based face recognition, *Journal of Vision* . Accepted.

Womelsdorf, T., Anton-Erxleben, K., Pieper, F. and Treue, S.: 2006, Dynamic shifts of visual receptive fields in cortical area MT by spatial attention, *Nat Neurosci* **9**(9), 1156–1160.

Wundrich, I. J., von der Malsburg, C. and Würtz, R. P.: 2004, Image representation by complex cell responses, *Neural Computation* **16**(12), 2563–2575.
URL: *http://dx.doi.org/10.1162/0899766042321760*

Würtz, R. P.: 1995, *Multilayer Dynamic Link Networks for Establishing Image Point Correspondences and Visual Object Recognition*, Verlag Harri Deutsch, Thun, Frankfurt am Main.

Yoshimura, Y., Dantzker, J. L. M. and Callaway, E. M.: 2005, Excitatory cortical neurons form fine-scale functional networks, *Nature* **433**(7028), 868–873.

Yuille, A. and Kersten, D.: 2006, Vision as Bayesian inference: analysis by synthesis?, *Trends in Cognitive Sciences* **10**(7), 301–308.

Zhao, W., Chellappa, R., Phillips, P. J. and Rosenfeld, A.: 2003, Face recognition: A literature survey, *ACM Computing Surveys* **53**(4), 399–458.

Zhu, J. and von der Malsburg, C.: 2004, Maplets for correspondence-based object recognition, *Neural Networks* **17**(8-9), 1311–1326.

Zipser, D. and Andersen, R. A.: 1988, A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons, *Nature* **331**(6158), 679–684.
URL: *http://dx.doi.org/10.1038/331679a0*

Zusammenfassung in deutscher Sprache

Dieses Kapitel beinhaltet eine deutsche Zusammenfassung der Arbeit. Die Einteilung in Unterkapitel entspricht den Kapiteln der Dissertation, feinere Unterteilungen wurden allerdings nur zum Teil übernommen, um die Lesbarkeit der Zusammenfassung zu verbessern. Bei Fachbegriffen, zu denen es keine deutschen Entsprechungen gibt, wurden die englischen Bezeichnungen beibehalten.

1 Einleitung

Eine zentrale Aufgabe des Gehirns ist das Finden von Korrespondenzen zwischen Mustern, z.B. zwischen dem Muster, das beim Betrachten einer Szene auf der Retina entsteht, und Aktivitätsmustern im Gehirn, die Erinnerungen an diese Szene repräsentieren. Das Korrespondenzfinden sollte auch dann funktionieren, wenn das Bild verschoben oder verdreht auf die Retina fällt, d.h. der Abgleich sollte *invariant* gegenüber Veränderungen sein, die das Erscheinungsbild, aber nicht die Bedeutung der Muster betreffen.

Auch wenn das Korrespondenzfinden ein grundlegendes und für die verschiedensten Sinnesmodalitäten wichtiges Problem ist, liegt der Schwerpunkt der vorliegenden Arbeit auf *visuellen* Mustern. Daher wird zunächst ein Überblick über Objekterkennung gegeben. In der theoretischen Neurowissenschaft lassen sich die meisten Modelle zur Objekterkennung nach ihren zugrunde liegenden Prinzipien in *merkmalbasierte* oder *korrespondenzbasierte* Verfahren einteilen. Im merkmalbasierten Ansatz werden in einer Hierarchie immer komplexere Merkmale aus dem Bild extrahiert, wobei gleichzeitig durch *Pooling* über die vorhandenen Varianzen schrittweise Invarianz erreicht wird. Die Stärke merkmalbasierter Verfahren liegt in der Klassifizierung von Objekten und Szenen ohne eindeutige Topologie. Korrespondenzbasierte Verfahren dagegen erreichen Invarianz, indem sie auch unter eventuellen Transformationen die Korrespondenzen zwischen einem Bild und einem gespeicherten Vergleichsmuster ermitteln und explizit repräsentieren. Diese Verfahren eignen sich besonders für klar strukturierte Objekte wie Gesichter und können an diesen sehr feine Details unterscheiden.

Der anschließende Überblick über den Stand der Technik in maschineller Bilderkennung zeigt, dass die erfolgreichsten rein funktional begründeten Methoden (z.B. Peronas generative Modelle oder Verfahren auf der Basis von SIFT-Merkmalen) merkmalbasierte und korrespondenzbasierte Ideen vereinen. Weiterhin werden einige Grenzfälle der Bilderkennung aufgeführt, in denen korrespondenzbasierte Ansätze unerlässlich sind. Es werden physiologische und psychologische Argumente diskutiert, die teils für merkmalbasierte und teils für korrespondenzbasierte Prozesse im Gehirn sprechen. Insgesamt zeigt sich, dass sowohl aus funktioneller wie experimenteller Sicht beide Ansätze in einem vollständigen Sehsystem benötigt werden.

Dennoch fristen Modelle, die erklären könnten, wie korrespondenzbasierte Mustererkennung konkret neuronal realisiert ist, in der theoretischen Neurowissenschaft ein Schattendasein. Diesem Thema, der neuronalen Korrespondenzfindung zwischen räumlichen Mustern, widmet sich die vorliegende Arbeit. Hauptvoraussetzungen für neuronalen Musterabgleich sind erstens die Existenz geeigneter Verbindungsstrukturen, und zweitens ein genereller Korrespondenzfindungsmechanismus. Mit letzterem beschäftigt sich Kapitel 2, optimale Verbindungsstrukturen und ihre Entstehung werden in den Kapiteln 3 und 4 behandelt. Kapitel 5 führt diese Ideen zusammen.

2 Ein korrespondenzbasiertes neuronales Modell zur Gesichtserkennung

In Kapitel 2 wird ein korrespondenzbasiertes Modell zur positionsinvarianten Objekterkennung entwickelt. Die Frage nach realistischen Verbindungsstrukturen wird dabei auf die Folgekapitel verschoben und hier stattdessen zum Erreichen der Invarianz eine simple direkte Konnektivität angenommen (s.u.). Das Modell baut auf dem *Dynamic Link Matching* auf, ist allerdings im Gegensatz zu Vorgängermodellen wie *Elastic Graph Matching* als ein großes neuronales System realisiert, das keiner algorithmischen “Tricks” oder “Abkürzungen” bedarf. Im Gegensatz zu früheren neuronalen Ansätzen wie dem *Running Blob* kann es mit seiner kolumnaren Dynamik (s.u.) Objekte in neuronal realistischer Zeit erkennen.

Wir wenden das System hier auf Gesichter an. Diese haben den Vorteil, dass sie holistische, topologisch klar definierte Objekte sind, für die der korrespondenzbasierte Ansatz besonders geeignet ist. Andererseits kommt es bei der Unterscheidung von mehr als 1000 Gesichtern einer Datenbank auf sehr feine Nuancen an, was das Problem anspruchsvoll macht. Auch existieren gut etablierte Benchmarks, auf denen wir unser System testen. Im Prinzip sollte das System nicht nur für Gesichter, sondern für alle Arten von Objekten geeignet sein, die eine prototypische, holistische Form haben, aber für deren Erkennung feine Details unterschieden werden müssen.

Prinzip des Korrespondenzfindens

Unter anderem baut das hier entwickelte Objekterkennungssystem auf Dynamic Link Matching auf, einer Methode zum Finden von Korrespondenzen zwischen Mustern. Dabei wird angenommen, dass beide Muster durch Schichten neuronaler Einheiten dargestellt sind, und dass zwischen allen Punkten beider Schichten Verbindungen existieren. Diese Verbindungen (oder Links) können dynamisch an- oder abgeschaltet werden; ein aktiver Link zwischen zwei Punkten signalisiert, dass die entsprechenden Stellen der beiden Muster miteinander korrespondieren. Die Links werden in erster Linie aktiviert durch die Ähnlichkeit der Merkmale an den Punkten, die sie verbinden. Bereits dadurch können identische Muster auch unter beliebigen Transformationen erfolgreich abgeglichen werden, vorausgesetzt die Ähnlichkeitsfunktion ist invariant gegenüber diesen Transformationen. Ein erfolgreicher Abgleich wird dargestellt durch die gleichzeitige Aktivierung aller Links, die korrespondierende Punkte verbinden. Reale visuelle Muster sind allerdings niemals völlig identisch, was bei einem nur durch Ähnlichkeit getriebenen Abgleich zu vielen falschen Korrespondenzen führen würde. Zusätzlich wird daher ein Mechanismus benötigt, der dafür sorgt, dass alle gefundenen Punkt-Punkt-Korrespondenzen auch global untereinander konsistent sind, d.h. dass sie zusammen eine sinnvolle, topologieerhaltende Abbildung zwischen beiden Mustern darstellen. Dafür werden, zusätzlich zum Ähnlichkeitsinput, Kooperation zwischen benachbarten Links, die miteinander konsistent sind, und Konkurrenz zwischen sich widersprechenden Links eingeführt. Bei richtiger Ausbalancierung dieser drei "Zutaten" können auch realistische Bildmuster auf diese Art erfolgreich miteinander abgeglichen werden. Dieses Prinzip liegt sowohl dem in diesem Kapitel entwickelten System als auch den Erweiterungen in Kapitel 5 zugrunde.

Das vollständige System

Die kleinste funktionelle Einheit des Objekterkennungssystems ist ein Modell der kortikalen Minikolumne. Eine Minikolumne, im Folgenden einfach *Einheit* genannt, repräsentiert durch die kollektive Dynamik ihrer ca. 100 Neuronen ein bestimmtes (visuelles) Merkmal an einer bestimmten Position. Mehrere Einheiten sind zu einer Makrokolumne, oder einfach *Kolumne*, zusammengefasst; jede Kolumne repräsentiert alle relevanten Merkmale an einem Ort. Wir beschreiben die Dynamik von Kolumnen durch eine Erweiterung der Evolutionsgleichung von Manfred Eigen mit einem zusätzlichen Kompetitionsparameter ν. Ist dieser 0, so haben alle Einheiten der Kolumne im Gleichgewichtszustand eine Aktivität, die proportional zu ihrem jeweiligen Input ist. Diese Einstellung wird verwendet für Kolumnen, die visuelle Merkmale darstellen sollen (*Merkmalkolumnen*). Höhere Werte von ν sorgen für eine mehr oder weniger starke Konkurrenz zwischen den Einheiten; für $\nu = 1$ bleibt am Ende des dynamischen Prozesses nur die Einheit mit dem größten Input aktiv. Kolumnen, die dynamische Links repräsentieren, werden durch ein kontinuierliches Ansteigen von ν zur Entscheidung für eine Untereinheit gezwungen (*Entscheidungskolumnen*).

Das vollständige System besteht aus drei Schichten. In der *Inputschicht* wird das Eingabebild durch ein rechteckiges Gitter von Merkmalkolumnen dargestellt. Als visuelle Merkmale werden die Amplituden der (komplexwertigen) Gabortransformation verwendet. Die darauf folgende *Assemblyschicht* hat eine Topologie in Form eines "Gesichtsgraphen", in dem jeder Knoten einer semantischen Landmarke wie "linkes Auge", "Nase", "Mundwinkel" usw. entspricht. Die Assemblyschicht besteht aus einer *Input Assembly*, die Signale aus der Inputschicht empfängt, und einer *Gallery Assembly*, die eine Überlagerung aller aktiven Gesichter der weiter unten beschriebenen Galerieschicht erhält. Zusätzlich gibt es in der Assemblyschicht dynamische Links, die wie oben beschrieben Korrespondenzen zwischen Inputschicht und Gallery Assembly finden und selbst den Informationsfluss von Inputschicht in die Input Assembly entsprechend dieser Korrespondenzen steuern. Schließlich folgt die *Galerieschicht*, die ebenfalls in einer Gesichtsgraphentopologie alle gespeicherten Gesichter enthält. Die Entscheidungskolumnen dieser Schicht repräsentieren jeweils eine bestimmte Landmarke, und jede Einheit einer Kolumne steht mit ihrer Aktivität für die jeweilige Landmarke in einem bestimmten Gesicht. Die tatsächlichen visuellen Merkmale eines solchen Gesichtsausschnitts sind in den reziproken Verbindungen zwischen Assembly- und Galerieschicht gespeichert. Die Galerieschicht erhält Input aus der Input Assembly, wodurch diejenigen Gesichter besonders aktiviert werden, die dem in der Input Assembly enthaltenen Bild am ähnlichsten sind. Gleichzeitig fließen die Gesichtsmuster der

Galerieschicht in die Gallery Assembly, wo sie sich zu einem gewichteten Durchschnittsgesicht aller Galeriebilder überlagern.

Im Laufe des Erkennungsprozesses finden die dynamischen Links die Korrespondenzen zwischen Eingabebild und dem Durchschnittsgesicht der Gallery Assembly, und können somit ein korrekt positioniertes Eingabebild an die Input Assembly schicken. Dieses wiederum aktiviert vornehmlich diejenigen Gesichter der Galerie, die dem Eingabebild am ähnlichsten sind, wodurch sich das Durchschnittsgesicht in der Gallery Assembly in eine Kopie des erkannten Gesichts verwandelt.

Das Verfahren ist inhärent positionsinvariant und robust gegenüber Verfälschungen, wie sie sich durch geänderte Gesichtsausdrücke, Verdeckungen oder Alterung ergeben. Es stellt sich heraus, dass das System auf Benchmarkdaten konkurrenzfähige Erkennungsraten zeigt, obwohl es mit Blick auf neuronale Plausibilität und nicht Erkennungsleistung entwickelt wurde. Weiterhin können im System ohne Veränderungen Aufmerksamkeitsphänomene realisiert werden, wie etwa räumliche Aufmerksamkeit oder Suche nach bestimmten Gesichtern.

3 Switchyards—Routingstrukturen im Gehirn

Im vorigen Kapitel wurde eine direkte “all-to-all”-Konnektivität angenommen, um Informationen von der Inputschicht zur Input Assembly zu routen. Diese Annahme ist allerdings unrealistisch, da die Anzahl der notwendigen Verbindungen quadratisch mit der Anzahl der Einheiten in den Schichten wächst und damit für realistische Größen der zu verbindenden Schichten schnell zu groß würde. Auch passiert die tatsächliche Verarbeitung visueller Information im Gehirn in mehreren hintereinander geschalteten Arealen. Es ist daher sinnvoll, zu mehrschichtigen Verbindungsstrukturen überzugehen, die eine volle Konnektivität zwischen Eingangs- und Ausgangsschicht über mehrere Zwischenschichten und dafür mit vergleichsweise wenig Verbindungen zwischen den einzelnen Schichten erreichen. Dieses “devide and conquer”-Prinzip wird in der Technik vielfach und auch in einigen neurowissenschaftlichen Modellen angewandt. Gefehlt hatte bisher allerdings eine konsequente Suche nach Verbindungsstrukturen, die die Gesamtmenge an benötigten neuronalen Ressourcen, also sowohl Verbindungen als auch merkmalrepräsentierende Einheiten der Zwischenschichten, minimieren.

Diese Optimierung wird hier durchgeführt und führt zu Architekturen wie den in Abb. 3.1 gezeigten, die wir *Switchyards* nennen. Quantitativ zeichnen sich die minimalen Strukturen durch einen ganz bestimmten Fanout (Anzahl der Verbindungen, die von einem Knoten ausgehen) aus, der je nach Randbedingungen im Bereich 3..9 liegt. Außerdem ist die Anzahl der Schichten eines Switchyards proportional zum Logarithmus der Anzahl von Einheiten in den zu verbindenden Schichten, ebenfalls mit einem klar definierten Vorfaktor.

Es folgen Erweiterungen der Untersuchungen auf den Fall, dass mehrere Merkmale simultan durch eine Verbindung geroutet werden können, was in der Optimierung das Gewicht der merkmalrepräsentierenden Einheiten erhöht. Außerdem werden sich zum Output hin verjüngende Switchyards untersucht, was eher der Realität im Gehirn entsprechen dürfte. In beiden Fällen bleiben die qualitativen Ergebnisse—logarithmische Abhängigkeit der Zahl der Schichten von der Schichtgröße und klar definierter Fanout und Vorfaktor für die Schichtanzahl—erhalten, allerdings ändern sich die tatsächlichen Größen.

Bei der anschließenden Interpretation der Ergebnisse werden die Unterschiede zu Sortiernetzwerken diskutiert: Switchyards sind billiger als Sortiernetzwerke, können aber nicht wie diese beliebige Permutationen der gerouteten Daten durchführen, sondern nur eine Untermenge, die allerdings gut zur Verarbeitung visueller Daten geeignet ist. Weiterhin wird die Plausibilität von dynamischem Routing und speziell mehrschichtigem Routing aufgezeigt durch die Diskussion von neuronalen Mechanismen zur Signalmultiplikation und von Koordinatentransformationen, die im Gehirn nachweislich ablaufen. Schließlich wird gezeigt, dass Switchyards mit den qualitativen und quantitativen Gegebenheiten im Primatenhirn vereinbar sind, soweit diese bekannt sind.

4 Ontogenese von Switchyards

Wenn man argumentiert, dass die recht komplizierten Strukturen, die im vorigen Kapitel vorgeschlagen wurden, tatsächlich im Gehirn realisiert sind, so sollte man erklären können, wie sich solche Verbindungsmuster ontogenetisch (d.h. vor der Geburt und ohne äußere Stimuli) entwickeln können. Da man nicht davon ausgehen kann, dass die Regeln für die Bildung dieser Strukturen explizit im Genom gespeichert sind, muss ein solcher Wachstumsprozess selbstorganisiert ablaufen. Ein Modell hierfür wird in diesem Kapitel vorgestellt.

Im Gehirn gibt es zwei Gruppen von Mechanismen, die das Wachstum von Nervenverbindungen beeinflussen. Einerseits können Axone (von einem Neuron ausgehende Nervenfasern) Gradienten chemischer Substanzen wahrnehmen und ihr Wachstum danach ausrichten. Zum Anderen können Synapsen zwischen Neuronen wachsen oder schrumpfen je nachdem, wie korreliert die Feueraktivitäten beider Zellen sind (vgl. die Hebb-Regel oder *spike timing dependent plasticity*). Durch zahlreiche Experimente ist belegt, dass sowohl das Fehlen chemischer als auch aktivitätsinduzierter Mechanismen während des Wachstums zu fehlerhaften Verbindungsstrukturen führen kann. Das hier vorgestellte Wachstumsmodell basiert auf chemischen Markern. Am Ende des Kapitels wird jedoch gezeigt, wie ein funktionell äquivalenter Mechanismus auf der Basis von Aktivitätskorrelationen formuliert werden kann.

Es wird angenommen, dass in jedem Knoten der Eingangsschicht ein eigener Markertyp produziert wird. Im Laufe des Wachstumsprozesses diffundieren diese Marker durch die gerade entstandenen Links nach oben und leiten so auch in den folgenden Schichten das Verbindungswachstum ein. Dem Prozess liegen drei Wirkmechanismen zugrunde. Ein *Topologieterm* sorgt dafür, dass in erster Linie gerade Links wachsen. D.h. in Abwesenheit anderer Einflüsse wachsen zwischen zwei Schichten zuerst Verbindungen zwischen Knoten mit gleicher Position in der jeweiligen Schicht, wodurch sozusagen die Koordinatensysteme beider Schichten aufeinander ausgerichtet werden. Daneben beinhaltet der Topologieterm einen Kooperationsterm zwischen benachbarten parallelen Links, was den Prozess robuster macht (s.u.). Ein *Normalisierungsterm* versucht, die Gesamtstärke der Verbindungen, die von einem Knoten ausgehen, auf einen bestimmten Zielwert zu bringen. Ist die Zahl der Verbindungen kleiner als der Zielwert, so unterstützt der Normalisierungsterm weiteres Wachstum von diesem Knoten aus, ansonsten wird Wachstum unterdrückt, oder bestehende Verbindungen schrumpfen. Ein *Ähnlichkeitsterm* schließlich verbietet Verbindungen zwischen Knoten, die zu viele gleiche chemische Marker enthalten (dabei werden im Endknoten des Links nur die Marker berücksichtigt, die nicht vom Ausgangsknoten, sondern von einer dritten Quelle herrühren). Das führt zu den für Switchyards typischen gespreizten Verbindungen auf den höheren Routingstufen, die dafür sorgen, dass von jedem Knoten der Eingangsschicht jeder beliebige Knoten der Ausgangsschicht erreicht werden kann. Das Endergebnis des ontogenetischen Prozesses ist fast identisch mit den in Kapitel 3 analytisch hergeleiteten Architekturen. Allerdings hat der selbstorganisierte Prozess die "wraparound"-Verbindungen vom Ende einer Schicht zum gegenüberliegenden Ende der nächsten durch nach innen verschobene Verbindungen ersetzt, was immer noch volle Konnektivität gewährleistet, aber biologisch plausibler ist.

Das Verfahren ist relativ robust gegenüber Störungen in den Anfangsbedingungen. Detaillierte Untersuchungen zeigen, dass auch anfängliche Verbindungsstärken, die mit bis zu 20% additivem Rauschen behaftet sind, meist immer noch zu perfekten Konnektivitäten führen. Allerdings ist diese Robustheit in erster Linie dem orientierten Kooperationsterm des Topologieterms geschuldet. Wird dieser durch nicht orientierte Kooperation oder durch gar keine Kooperation ersetzt, so fällt die Robustheit deutlich und statistisch signifikant ab.

Weiterhin wird die Entstehung dreidimensionaler Switchyards untersucht. Es stellt sich heraus, dass der beschriebene Mechanismus im Prinzip auch solche Strukturen hervorbringen kann. Allerdings ist im dreidimensionalen Fall aufgrund der Rotationssymmetrie des Mechanismus eine spontane Symmetriebrechung notwendig, was den Prozess deutlich störanfälliger macht.

5 Mustererkennung mit Switchyards

In den vorangegangenen Kapiteln wurde zum Einen ein Mechanismus zum Finden von Korrespondenzen zwischen zwei direkt verbundenen neuronalen Aktivitätsmustern eingeführt, aus dessen Basis dann ein vollständiges Gesichtserkennungssystem entwickelt wurde. Zum Anderen wurden Switchyards untersucht, die eine vollständige Konnektivität zwischen neuronalen Schichten über mehrere Zwischenschichten herstellen, und damit im Prinzip eine kostengünstigere und neuronal plausiblere Verbindungsstruktur zum Informationsrouting darstellen. In diesem Kapitel werden beide Ideen zusammengeführt, um das Korrespondenzfinden zwischen Mustern und schließlich Objekterkennung über mehrstufige Routingstrukturen hinweg zu realisieren.

Wenn man versucht, den in Kapitel 2 beschriebenen Korrespondenzfindungsprozess auf Muster zu erweitern, die nur indirekt über einen mehrstufigen Switchyard verbunden sind, so wird man mit dem Problem konfrontiert, dass anfangs an keiner der Routingstufen auf beiden Seiten sinnvolle Muster anliegen, zwischen denen Korrespondenzen ermittelt werden könnten. Für "typische" visuelle Muster (s.u.) ist Korrespondenzfindung über einen Switchyard dennoch möglich, wie im Folgenden beschrieben wird. Die merkmalrepräsentierenden Zwischenschichten des Switchyards müssen dafür doppelt ausgelegt werden, mit je einer getrennten Schicht für "bottom-up"-Informationsfluss vom Eingang zum Ausgang des Switchyards (*Input Assembly*) und für von dort zurückfließende "top-down"-Information (*Gallery Assembly*). Die Links jeder Routingstufe werden nun—neben Kooperation mit Nachbarlinks—durch die Ähnlichkeit der jeweils unten anliegenden Input Assembly und der oben anliegenden Gallery Assembly aktiviert. Man geht davon aus, dass zu Beginn des dynamischen Prozesses alle Links des Switchyards leicht aktiv sind. Dadurch kann das am Eingang anliegende Muster bis zur obersten Input Assembly fließen, wird dabei allerdings durch die unspezifisch offenen Links “verschmiert”, d.h. tiefpassgefiltert. Zwischen dem verschmierten Eingangsbild und dem am Ausgang anliegenden Vergleichsmuster können dennoch grobe Korrespondenzen gefunden werden, die gerade der Genauigkeit entsprechen, die die gespreizten, weitreichenden Verbindungen der obersten Routingstufe darstellen können. Durch diese Verbindungen kann das Zielmuster nun in die ungefähr korrekte Position der nächsttieferen Gallery Assembly gerouted werden, wo es mit einem weniger verschmierten Eingangsbild auf feinere Korrespondenzen abgeglichen wird. Dies setzt sich fort bis zur untersten Routingstufe, wo ein fast korrekt positioniertes Vergleichsbild direkt mit dem Eingangsbild auf feinste Verschiebungen hin verglichen werden kann. Dieses Prinzip wird exemplarisch an eindimensionalen Zufallsmustern vorgeführt. Weiterhin wird das hier entwickelte Verfahren mit dem “Blob focussing” auf Bruno Olshausens *Shifter Circuits* verglichen. Beide Ansätze führen zu qualitativ ähnlichen Ergebnissen, allerdings fehlt bei Olshausen eine plausible Realisierung des notwendigen “top-down”-Informationsflusses.

Im Anschluss wird das hier vorgestellte Verfahren zu einem Muster*erkennungs*system ausgebaut. Dafür wird das Zielmuster am Ausgang des Switchyards durch eine weitere, oberste Doppelschicht aus Input Assembly und Gallery Assembly ersetzt, an die sich eine Galerieschicht wie in Kapitel 2 anschließt. Die in der Galerieschicht gespeicherten Muster überlagern sich anfangs in der obersten Gallery Assembly zu einem Durchschnittsmuster, das nun—wie vorher das Zielmuster—mit dem Eingangsmuster abgeglichen wird. Gleichzeitig aktiviert die oberste Input Assembly vor allem die zu ihr passenden Muster in der Galerie, was schließlich zur Dominanz eines “erkannten” Musters in der Galerie führt. Dadurch verändert sich der Inhalt in der Gallery Assembly von einem Durchschnittsmuster hin zu dem erkannten Muster, was wiederum die Korrespondenzfindung im darunter liegenden Switchyard verbessert.

Voraussetzung für Korrespondenzfinden und Erkennung mit mehrstufigen Routingstrukturen ist, dass die zu verarbeitenden visuellen Muster signifikante tieffrequente Information enthalten. Nur dann können nämlich erfolgreich Korrespondenzen zwischen dem tiefpassgefilterten Eingangsbild und dem Vergleichsbild gefunden werden, und nur dann führt die Überlagerung aller Galeriebilder zu einem Durchschnittsmuster, das noch brauchbare Struktur enthält. Ein Beispiel, für das diese Voraussetzungen nicht erfüllt sind, sind sogenannte “random dot stereograms”, Stereogramme aus Zufallspunkten, zu deren dreidimensionaler Wahrnehmung die Verschmelzung der enthaltenen überlagerten Punktwolken notwendig ist. Dies erfordert Korrespondenzfindung zwischen Mustern ohne brauchbare tieffrequente Information, was die hier vorgestellten mehrschichtigen Systeme überfordert.

6 Diskussion

Am Anfang dieser Arbeit wurden als Voraussetzungen für das Abgleichen räumlicher Muster im Gehirn erstens die Existenz geeigneter Verbindungsstrukturen und zweitens ein genereller Mechanismus zur Korrespondenzfindung genannt. Mit neuronal plausiblem Korrespondenzfinden und darauf aufbauend einem Objekterkennungssystem beschäftigt sich Kapitel 2, der analytischen Herleitung und möglichen selbstorganisierten Entstehung mehrschichtiger Verbindungsstrukturen haben wir uns in den Kapiteln 3 und 4 gewidmet. Kapitel 5 führt diese Ideen zusammen.

Im Zuge der Arbeit haben sich drei Hauptargumente herauskristallisiert, warum Musterabgleich über mehrere Zwischenstufen hinweg ein zentrales Prinzip der Hirnfunktion sein sollte. Zum Einen sind Objekterkennungssysteme, die auf diesem Prinzip aufbauen, von sich aus generativ. Notwendigerweise fließt reichhaltige visuelle Information von "höheren" Bereichen zurück und generiert in den eingangsseitigen Arealen explizite Repräsentationen der Entscheidungen und Wahrnehmungen des Systems. Weiterhin können hintereinander geschaltete Routingstufen, die jeweils nur einfache visuelle Transformationen ausführen, zusammen einen sehr großen Raum möglicher Abbildungen abdecken. Dieser kombinatorische Code ist deutlich sparsamer, als wenn sämtliche Transformationen von einer einzigen Schicht umgesetzt werden müssten. Drittens stellen mehrschichtige, korrespondenzbasierte Erkennungssysteme eine Art Synthese zwischen dem merkmalbasierten und dem korrespondenzbasierten Ansatz dar. Der erste, passive Durchlauf des Eingangsbildes ähnelt dem Erkennungsprozess in merkmalbasierten Systemen, während das darauf folgende sukzessive Abgleichen der einzelnen Stufen korrespondenzbasiert ist. Insofern könnten solche Systeme prinzipiell die Schnelligkeit merkmalbasierter Methoden mit der Genauigkeit korrespondenzbasierter Ansätze vereinen.

Natürlich ist die vorliegende Dissertation hierbei nur ein erster Schritt. In Zukunft wäre es neben dem weiteren Ausbau der in Kapitel 5 nur an "Spielzeugbeispielen" veranschaulichten Prinzipien wichtig, das hier vorgestellte System auf die Erkennung aus mehr als einer Kategorie zu erweitern. Des Weiteren sollte auch die Galeriedomäne hierarchisch aufgebaut werden, ähnlich, wie dies auf Eingangsseite mit Einführung der Switchyards geschehen ist.

Lebenslauf

Persönliche Daten

Adresse	Philipp Wolfrum, Birkholzweg 19, 60433 Frankfurt E-Mail: wolfrum@fias.uni-frankfurt.de
Geburt	am 08. 04. 1978 in Heilbronn
Familienstand	ledig

Schulbesuch

9/1988–6/1997	Gymnasium bei St. Michael in Schwäbisch Hall Abitur, Note: 1,0

Wehrdienst

9/1997–6/1998	Klarinettist beim Heeresmusikkorps in Ulm/Donau

Studium

01/99	Aufnahme in die Studienstiftung des deutschen Volkes
10/1998–7/2004	Universität Stuttgart Technische Kybernetik Diplomarbeit über nichtlineare Systemidentifikation bei Prof. Dr. Allgöwer Abschluss: Diplomingenieur, Note: 1,1
9/2002–5/2003	Fulbrightstudium an der Boston University, USA Cognitive and Neural Systems am Lab von Prof. Stephen Grossberg Abschluss: Master of Arts, Grade Point Average: 3.96

Promotion

8/2004–7/2005	Ruhr-Universität Bochum Wissenschaftlicher Mitarbeiter am Institut für Neuroinformatik
Seit 8/2005	Frankfurt Institute for Advanced Studies, Universität Frankfurt Gruppe Prof. Dr. Christoph von der Malsburg
12/2006	Mündliche Prüfung in Informatik zwecks Zulassung zur Promotion

Publikationen (peer-reviewed)

Philipp Wolfrum, Alejandro Vargas, Martha Gallivan, and Frank Allgöwer: 2005, Complexity reduction of a thin film deposition model using a trajectory based nonlinear model reduction technique. In *Proc. American Control Conference*, Vol. 4, pp. 2566–2571.

Philipp Wolfrum and Christoph von der Malsburg: 2007, A marker-based model for the ontogenesis of routing circuits. In *Artificial Neural Networks – ICANN 2007*, volume 4669 of *LNCS*, Springer, pp. 1–8.

Yasuomi Sato, Christian Wolff, Philipp Wolfrum, and Christoph von der Malsburg: 2007, Dynamic Link Matching between Feature Columns for Different Scale and Orientation. In *Proc. ICONIP 2007*, volume 4984 of *LNCS*, Springer, pp. 385–394

Philipp Wolfrum and Christoph von der Malsburg: 2007, What is the optimal architecture for visual information routing? *Neural Computation*, **19**:3293–3309.

Philipp Wolfrum, Jörg Lücke, and Christoph von der Malsburg: 2008, Invariant face recognition in a network of cortical columns. In *Proc. International Conference on Computer Vision Theory and Applications*, Vol.2 , pp. 39-45.

Philipp Wolfrum, Christian Wolff, Jörg Lücke, and Christoph von der Malsburg: A recurrent dynamic model for correspondence-based face recognition, *Journal of Vision*. Accepted.

Vorträge auf Einladung

07/2006	Riken Brain Science Institute, Wako-shi, Japan, Gruppe von Shun-ichi Amari. *Object Recognition with Networks of Cortical Columns.*
08/2007	Redwood Center for Theoretical Neuroscience, Berkeley, USA, Gruppe von Bruno Olshausen. *Switch Yards in the Brain*

Reviewertätigkeit

American Control Conference (ACC)

IEEE Conference on Decision and Control (CDC)