CORTEX-INSPIRED DEVELOPMENTAL LEARNING NETWORKS FOR STEREO VISION

By

Mojtaba Solgi

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science-Doctor of Philosophy

2013

ABSTRACT

CORTEX-INSPIRED DEVELOPMENTAL LEARNING NETWORKS FOR STEREO VISION

By

Mojtaba Solgi

How does the human brain make sense of the 3D world while its visual input, the retinal images, are only two-dimensional? There are multiple depth-cues exploited by the brain to create a 3D model of the world. Despite the importance of this subject both for scientists and engineers, the underlying computational mechanisms of the stereo vision in the human brain is still largely unknown. This thesis is an attempt towards creating a developmental model of the stereo vision in the visual cortex. By developmental we mean that the features of each neuron are developed, instead of hand-crafted, so that the limited resource is optimally used. This approach helps us learn more about the biological stereo vision, and also yields results superior to those of traditional computer vision approaches, e.g., under weak textures. Developmental networks, such as Where-What Networks (WWN), have been shown promising for simultaneous attention and recognition, while handling variations in scale, location and type as well as inter-class variations. Moreover, in a simpler prior setting, they have shown sub-pixel accuracy in disparity detection in challenging natural images. However, the previous work for stereo vision was limited to 20 pixel stripes of shifted images and unable to scale to real world problems. This dissertation presents work on building neuromorphic developmental models for stereo vision, focusing on 1) dynamic synapse retraction and growth as a method of developing more efficient receptive fields 2) training for images that involve complex natural backgrounds 3) integration of depth perception with location and type information. In a setting of 5 object classes, $7 \times 7 = 49$ locations and 11 disparity

levels, the network achieves above 95% recognition rate for object shapes, under one pixel disparity detection error, and under 10 pixel location error. These results are reported using challenging natural and synthetic textures both on background and foreground objects in disjoint testing.

To my parents, for their unlimited love and support throughout my life.
And to my beautiful wife, Ladan, for putting up with me while I worked on this dissertation.

ACKNOWLEDGMENTS

I have to thank my advisor, Professor Juyang (John) Weng for his dedicated and careful supervision of this work during my six years at the Embodied Intelligence Lab. I would like to thank professors George Stockman, Taosheng Liu and Fathi Salem for serving on my committee and providing useful and critical comments on this work. A special thanks goes to Dr. Stockman for being extremely supportive during my years at graduate school, both academically and personally. His great personality is known to everybody who knows him. Thanks for Dr. Taosheng Liu for teaching me about Vision Science, and supervising my research during and after my rotation in his lab in Summer 2010. Dr. Salem's teachings in his Neural Networks class was imperative to my theoretical understanding of the field. Also, I would like to extend many thanks to Mr. Mark McCullen from whom I learned a great deal while working with him as a TA for 8 semesters. I am grateful to the Computer Science graduate director, professor Eric Torng, for his support and consultation during difficult times.

This dissertation is the result of my six years of work at Michigan State University. There are many people to whom I feel grateful. I should thank my labmates Mathew Luciw, Paul Cornwell, Zhengping Ji, Arash Ashari, Benjamin Syzek, Kajal Miyan, Nikita Wagle, Zejia Zhang, Max Leason, Stephen Peslaski and Charles Bardel for being there for discussions on my research and beyond. I truly enjoyed and valued the company of many friends at MSU (too many to name). Specially, I should mention my friend and roommate for 4.5 years, Dr. Seyyed Rouhollah Jafari Tafti, for being a reliable presence all these years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Background

1.1 Physiology of binocular vision

This chapter presents the fundamentals of neurological knowledge required for understanding the biological binocular vision systems regarding disparity encoding and detection. At the end of the chapter, related works on disparity models are presented. Most material on biological visual systems is adapted from Kandel 2000 [49] and Ramtohul 2006 [88], and those about LCA and MILN are largely adapted from Weng & Luciw 2009 [114].

The human visual system is one of the most remarkable biological systems in nature, formed and improved by millions of years of evolution. About the half of the human cerebral cortex is involved with vision, which indicates the computational complexity of the task. Neural pathways starting from the retina and continuing to V1 and the higher cortical areas form a complicated system that interprets the visible light projected on the retina to build a three dimensional representation of the world. In this chapter we provide background information about the human visual system and the neural mechanisms involved during the development and operation of visual capabilities.

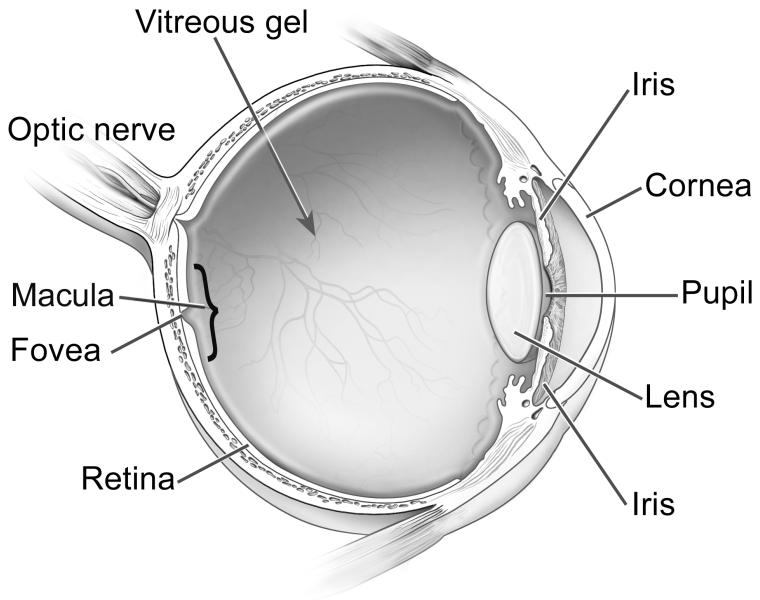


Figure 1.1: Anatomy of the human eye (reprinted from [73])

1.1.1 Eye

When visible light reaches the eye, it first gets refracted by the cornea. After passing through the cornea, it reaches the pupil. To control the amount of light entering the eye, the pupils size is regulated by the dilation and constriction of the iris muscles. Then the light goes through the lens, which focuses it onto the retina by proper adjustment of its shape.

1.1.2 Visual Pathway

The early visual processing involves the retina, the lateral geniculate nucleus of thalamus (LGN), and the primary visual cortex (V1). The visual signals then go through the higher visual areas, which include V2, V3, V4 and V5/MT. After initial processing in the retina, output from each eye goes to the left and right LGNs, at the base of either side of the brain.

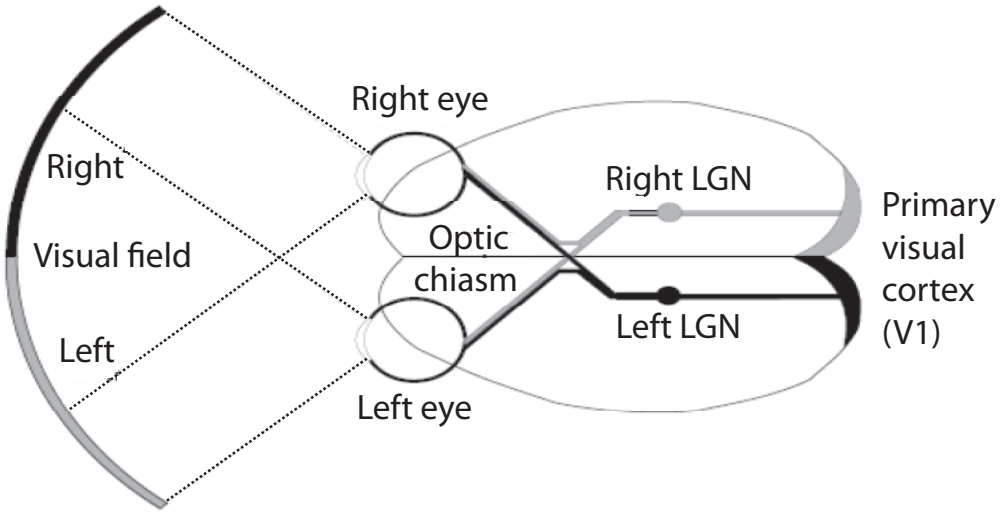


Figure 1.2: Visual pathway in human (reprinted from [68])

LGN in turn does some processing on the signals and projects to the V1 of the same side of the brain. The optic nerves, going to opposite sides of the brain, cross at a region called the *optic chiasm*. V1 then feeds its output to higher visual cortices where further processing takes place. Fig. 1.2 presents a schematic overview of the visual pathway.

1.1.3 Retina

The retina is placed on the back surface of the eye ball. There is an array of special purpose cells on the retina, such as photoreceptors, that are responsible for converting the incident light into neural signals.

There are two types of light receptors on the retina: 1) rods that are responsible for vision in dim light 2) cones that are responsible for vision in bright light. The total number of rods is more than cones, however there are no rod cells in the center of retina. The central part of the retina is called the *fovea* which is the center of fixation. The density of the cone cells is high in the fovea, which enables this area to detect the fine details of retinal images.

For the first time, Stephen Kuffler recorded the responses of retinal ganglion cells to rays

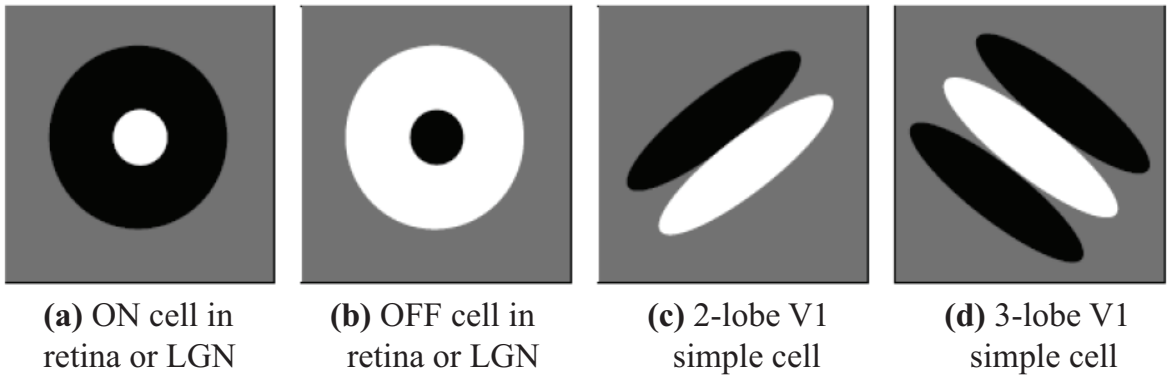


Figure 1.3: Samples of the receptive fields shapes in human V1 (reprinted from [68])

of light in a cat in 1953(Hubel, 1995). He discovered that it is possible to influence the firing rate of a retinal ganglion cell by projecting a ray of light to a specific spot on retina. This spot is called the *receptive field* (RF) of the cell. Below is a definition of receptive field from Livine & Shefner 1991:

"Area in which stimulation leads to a response of a particular sensory neuron"

In other words, for any neuron involved in the visual pathway, the receptive field is a part of the visual stimuli that influences the firing rate of the specific neuron. Fig. 1.3 shows a few examples of the shape of receptive fields in the visual pathway.

1.1.4 LGN

The LGN acts like a relay that gets signals from the retina and projects to the primary visual cortex (V1). It consists of neurons similar to retinal ganglion cells, however the role of these cells is not clear yet. The arrangement of the LGN neurons is *retinotopic*, meaning that the adjacent neurons have gradually changing, overlapping receptive fields. This phenomena is also called *topographic representation*. It is believed that the LGN cells perform edge detection on the input signals they receive from the retina.

1.1.5 Primary Visual Cortex

Located at the back side of the brain, the primary visual cortex is the first cortical area in the visual pathways. Similar to LGN, V1 neurons are reinotopic too. V1 is the lowest level of the visual system hierarchy in which there are *binocular* neurons. These neurons are identified by their ability to respond strongly to stimuli from either eye. These neurons also exhibit preference to specific features of the visual stimuli such as spatial frequency, orientation and direction of motion. It has been observed that some neurons in V1 show preference for particular disparities in binocular stimuli - stimuli with a certain disparity causes potential discharge in the neuron. V1 surface consists of columnar architecture where neurons in each column have more or less similar feature preference. In the columnar structure, feature preference changes smoothly across the cortex, meaning that nearby columns exhibit similar and overlapping feature preference while columns far from each other respond differently to the same stimuli. Overall, there is a smoothly varying map for each feature in which preferences repeat at regular intervals in any direction. Examples of such topographic maps include orientation maps, and disparity maps which are the subject of study in this thesis.

1.1.6 Disparity

It is known that the perception of depth arises from many different visual cues (Qian 1997 [85]) such as occlusion, relative size, motion parallax, perspective, shading, blur, and relative motion (DeAngelis 2000 [19], Gonzalez & Perez 1998 [36]). The cues mentioned were monocular. There are also binocular cues because of the stereo property of the human vision. Binocular disparity is one of the strongest binocular cues for the perception of depth. The existence of disparity is because the two eyes are laterally separated. The terms *stereo*

vision, *binocular vision* and *stereospsis* are interchangeably used for the three-dimensional vision based on binocular disparity.

1.1.7 Geometry of Binocular Vision

Fig. 1.4 illustrates the geometry of the stereo vision. Suppose that the eyes are focused(fixated) at the point Q. The images of the fixation point falls on the fovea, Q_L and Q_R on the left and right eyes, respectively. These two points are called *corresponding* points on the retina, since they both get the reflection of the same area of the visual field (fixation point in this example). The filled circle S is closer to the eyes and its image reflects on different spots on the two retinas, which are called *non-corresponding* points. This lack of correspondence is referred to as disparity. The relative depth of the point S, distance z from the fixation point, can be easily calculated given the retinal disparity $\delta = r - l$, and the interocular distance (the distance between the two eyes), I. Since this kind of disparity is caused by the location of the objects on the horizontal plane, it is known as *horizontal disparity*.

It can be proven that all the points that are at the same disparity as the fixation point lie on a semi-sphere in the three-dimensional space. This semi-sphere is referred to as the *horopter*. Points on the horopter, inside and outside of the horopter have zero, negative and positive disparities, respectively. The projection of the horopter on the horizontal plane crossing the eyes (at the eyes level) is the *Vieth-Muller* circle.

It is known that another type of disparity, called *vertical disparity*, plays some role in the perception of depth, however, it has not been studied as intensively as horizontal disparity. The vertical disparity occurs when an object is considerably closer to one eye than the other. According to Bishop 1989 [8], such vertical disparities occur when objects are located

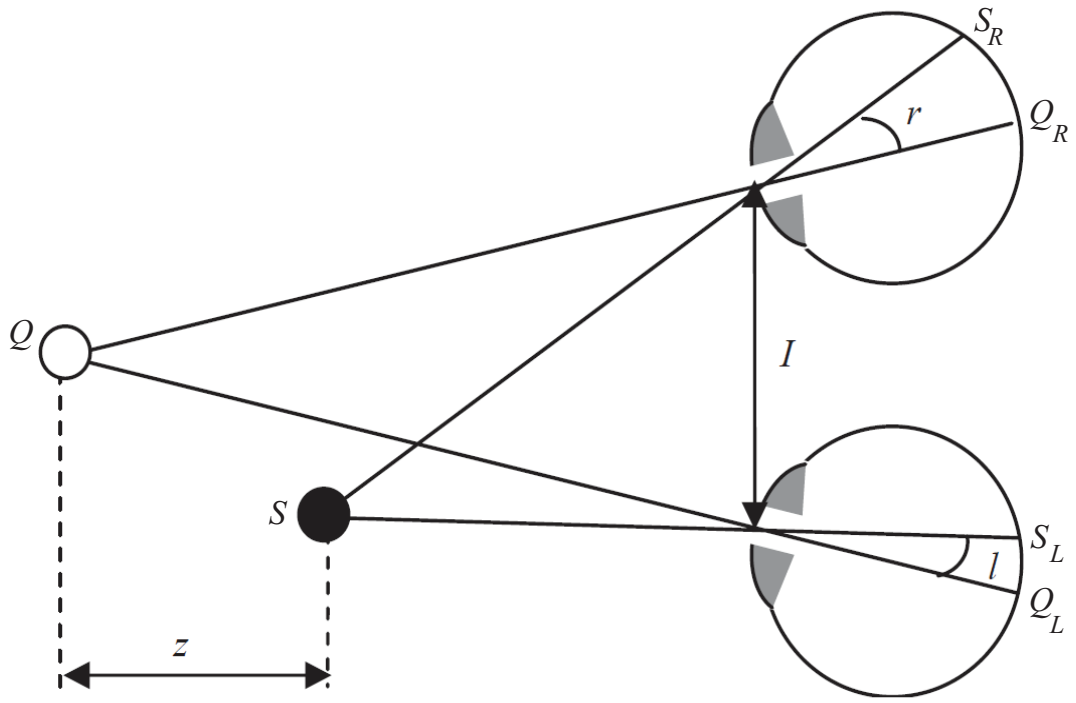


Figure 1.4: The geometry of stereospsis (reprinted from [89])

relatively close to eyes and are above or below the horizontal visual plane, but do not reside on the median plane, the vertical plane that divides the human body into left and right halves. Fig. 1.6 simply illustrates vertical disparity. Point P is above the visual plane and to the right of the median plane, which makes it closer to the right eye. It can be seen that the relation $\beta_2 > \beta_1$ holds between two angles β_1 and β_2. The vertical disparity, denoted by v, is the difference between these two angles, $v = \beta_2 - \beta_1$ [8].

1.1.8 Encoding of Binocular Disparity

There are several ways that binocular disparities can be described. One can encode disparity as the retinal positions of visual features (such as edges) corresponding to the same spots in the visual field, or formulate the images as a set of sine waves using Fourier analysis, and encode disparity as the phase difference between the sine waves at the same retinal position.

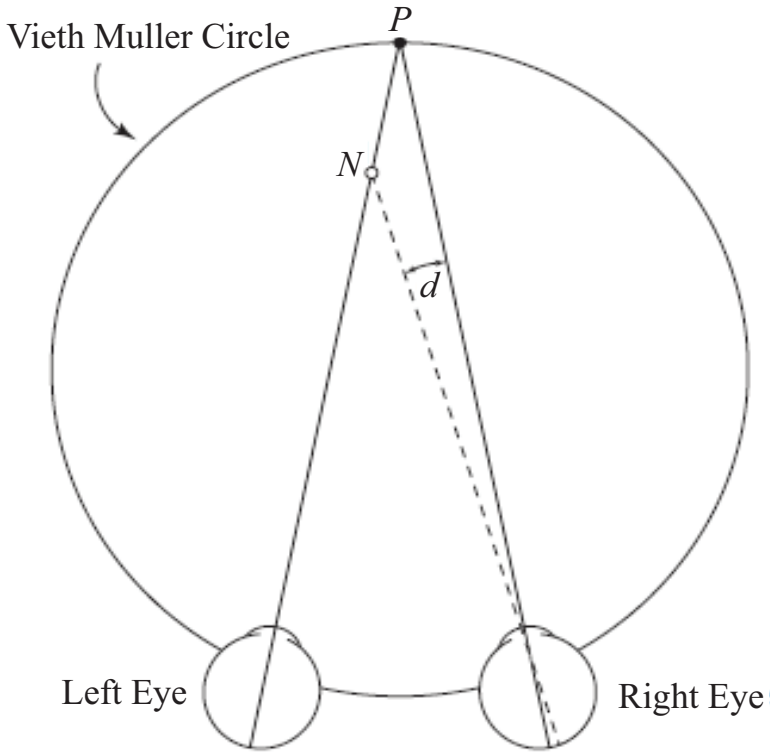


Figure 1.5: Horizontal Disparity and the Vieth-Muller circle(reprinted from [19])

The former is referred to as *position disparity* and the latter is *phase disparity*. There is evidence supporting the existence of the both of disparities in biological visual systems [16]. These two possibilities are illustrated in Fig. 1.7.

1.2 Existing Work in Computational Modeling of Binocular Vision

Perhaps the first remarkable study of the neural mechanisms underlying binocular vision dates back to the 1960's by Barlow et. al. [9]. They discovered that neurons in the cat striate cortex respond selectively to the objects with different binocular depth. In 1997 Poggio and Fischer [33] did a similar experiment with an awake macaque monkey that confirmed the previous evidence by Barlow et. al. [9]. Since the visual system of these animals to a great

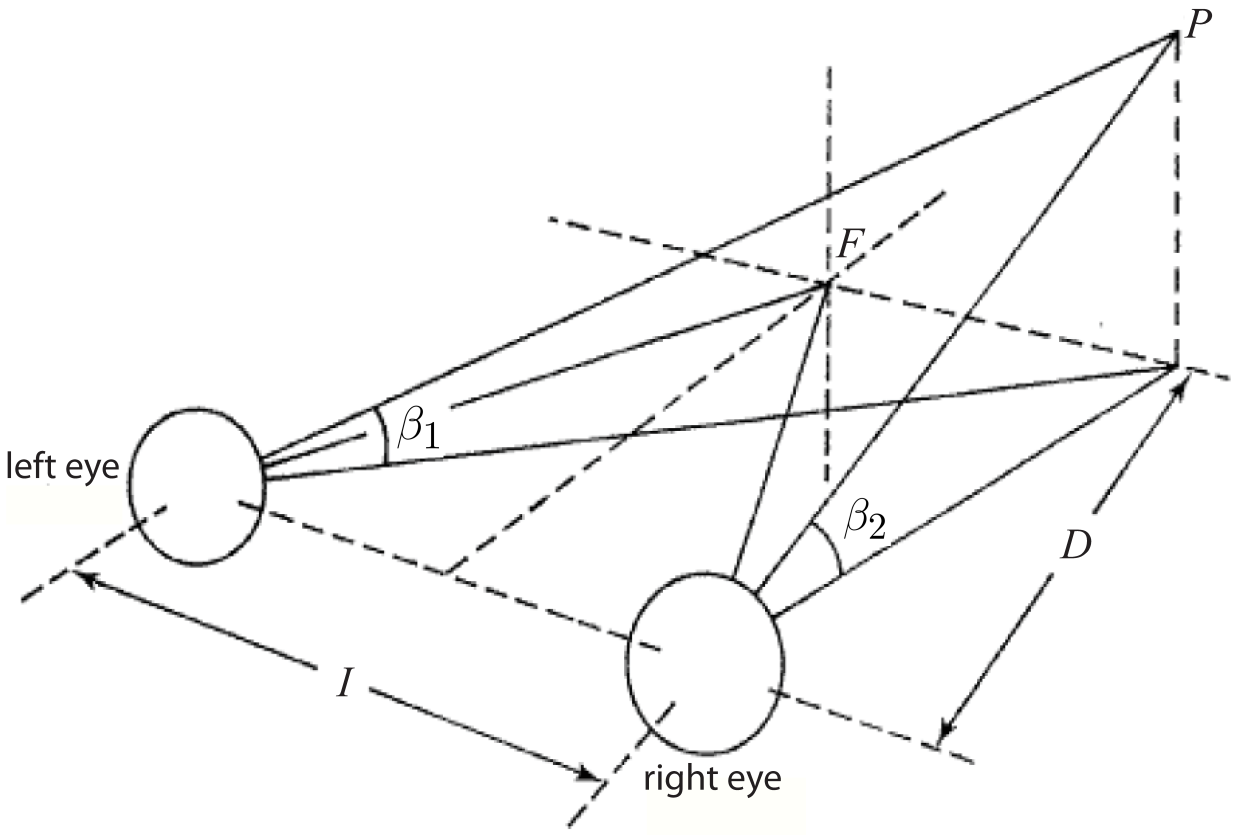


Figure 1.6: Vertical Disparity (reprinted from [8])

extent resembles that of human, researchers believe that there are disparity-selective neurons in the human visual cortex as well. Poggio & Fischer [33] used solid bars as visual stimuli to identify and categorize the disparity selective neurons. Table 1.1 contains the naming they used to categorize the cell types.

Disparity selective cell type	**Placement of stimuli**
Tuned-excitatory	Stimuli at zero disparity
Tuned-inhibitory	Stimuli at all disparities except those near zero disparity
Near	Stimuli at negative disparity
Far	Stimuli at positive disparity

Table 1.1: Four basic types of disparity selective neurons.

Julesz 1971 [47] invented random dot stereogram (RDS), which was a great contribution

to the field. A random dot stereogram consists of two images filled with dots randomly black or white, where the two images are identical except a patch of one image that is horizontally shifted in the other (Fig. 1.8).

When a human subject fixates eyes on a plane farther or closer to the plane on which RDS lies, due to the binocular fusion in the cortex, the shifted region jumps out (seems to be at a different depth from the rest of the image). Experiments based on RDS contributed to strengthen the theory of 4 categories of disparity selective neurons [36]. Later experiments revealed the existence of two additional categories, named *tuned near* and *tuned far* [82]. Fig. 1.9 depicts the 6 categories identified by Poggio et. al. 1988 [82].

Despite neurophysiological data and thrilling discoveries in binocular vision, a computational model was missing until 1990 when Ohzawa et. al. [20] published their outstanding article in Science journal. They introduced a model called the *disparity energy model*. Later some results from physiological studies did not match the predictions made by energy model. Read et. al. 2002 [76] proposed a modified version of the original energy model. In the following sections, we present an overview of the two different versions of the important work of the energy model.

1.2.1 Energy Model

Ohzawa-DeAngelis-Freeman (ODF) 1990 [20] studied the details of binocular disparity encoding and detection in the brain, and tried to devise a computational model compatible with the biological studies of binocular vision. They argued that at least two more points need to be taken into account before one can devise a plausible model of the binocular vision.

1. Complex cells must have much finer receptive fields compared to what was reported

by Nikara et. al. [71]

2. Disparity sensitivity must be irrelevant to the position of the stimulus within the receptive field.

Considering the limitations of the previous works and inspired by their own predictions, Ohzawa et. al. presented the *Energy Model* for disparity selective neurons. Fig. 1.10 schematically shows their model. There are 4 binocular Simple Cells (denoted by S) each receiving input from both eyes. The receptive field profile of the simple cells is depicted in small boxes. The output of the simple cells then goes through a half-wave rectification followed by a squaring function. A complex cell (denoted by Cx in Fig. 1.10) then adds up the output of the 4 subunits $S1$, $S2$, $S3$ and $S4$ to generate the final output of the network.

Read et. al. [76] completed the previous energy model by Ohzawa et. al. [20]. They added monocular simple cells to the model that performs a half-wave rectification on the inputs from each eye before feeding them to the binocular simple cells. The authors claimed that the modification in the Energy Model results in the neurons exhibiting behavior close to real neuronal behavior when the input is anti-correlated binocular stimuli. Fig. 1.11 shows the modified Energy Model.

1.2.2 Wiemer et. al. 2000

Wiemer et. al. [44] used SOM as their model to exhibit self-organization for disparity preference. Their work was intriguing as for the first time it demonstrated the development of modeled binocular neurons. They took stereo images form three-dimensional scenes, and then built a binocular representation of each pair of stereo images by attaching corresponding stripes from the left and right images. They then selectively chose patches from the binocular

representation to create their input to the network. An example of this pre-processing is shown in Fig. 1.12.

After self-organization they obtained *disparity maps* that exhibited some of the characteristics observed in the visual cortex. Fig. 1.13 shows one exmaple of the maps they reported.

1.2.3 Works based on LLISOM

Laterally **I**nterconnected **S**ynergetically **S**elf-**O**rganizing **M**aps by Mikkulainen et. al. [68] is a computational model of the self-organizing visual cortex that has been extensively studied over the past years. It emphasized the role of the lateral connections in such self-organization. Mikkulainen et. al. [68] point out three important findings based on their models:

1. Self-organization is driven by bottom-up input to shape the cortical structure
2. Internally generated input (caused by genetic characteristics of the organism) also plays an important role in Self-organization of the visual cortex.
3. Perceptual grouping is accomplished by interaction between bottom-up and lateral connections.

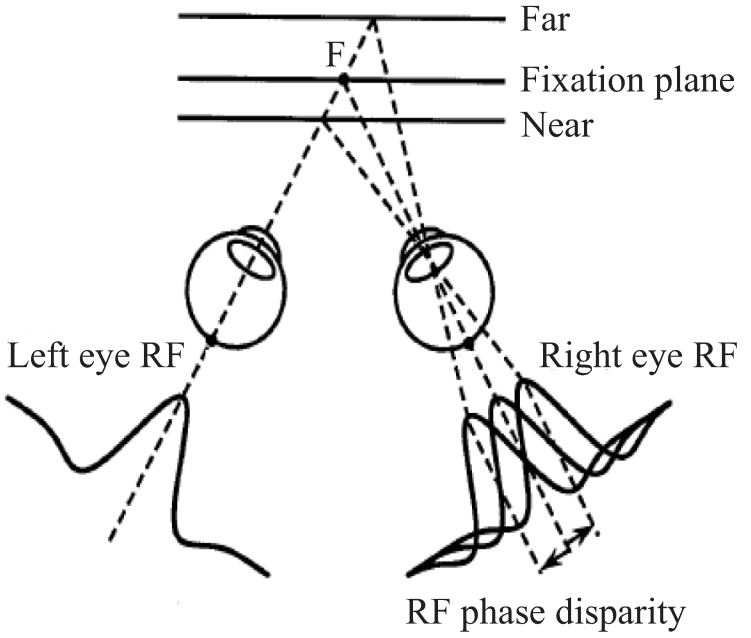
Although LLISOM was an important work that shed some light on the self-organization in the visual cortex, they failed to model an important part of the signals received at the visual cortex, namely *top-down* connections, and the role of this top-down connections in perception and recognition.

Fig. 1.14 shows an overall structure of the LLISOM. It consists of retina, LGN-ON and LGN-OFF sheets, and V1 sheet. Unlike SOM, in LLISOM each neuron is locally connected

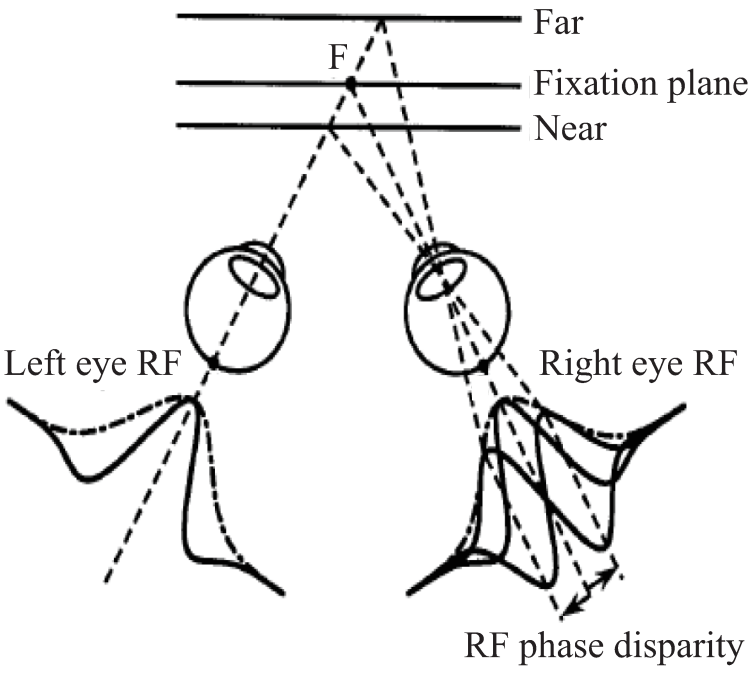
to a number of neurons in its lower-level sheet. Also, neurons are laterally connected to their neighbors. The strength of the connection between neurons is adapted during learning based on Hebbian learning rule. The process of learning connection weights is called *self-organization*. Thanks to lateral connections, LLISOM gains finer self-organized maps than SOM.

Fig. 1.15 presents an example of the self-organizing maps using LLISOM.

Ramtohul 2006 [88] studied the self-organization of disparity using LLISOM. He extended the basic architecture of LLISOM to handle two eyes, and the new architecture *two eye model* for disparity selectivity. Fig. 1.16 shows a schematic diagram of his model. He then provided the network with patches of natural images as input to investigate the emergence of disparity maps. The network successfully developed topographic disparity maps as a result of input-driven self-organization using LLISOM. However, this work did not provide any performance measurement report, since the motor/action layer was absent in the model. Fig. 1.17 shows an example of the topographic disparity maps reported by Ramtohul 2006 [88].



(a) Position Difference Model



(b) Phase Difference Model

Figure 1.7: Two models of disparity encoding (reprinted from [5])

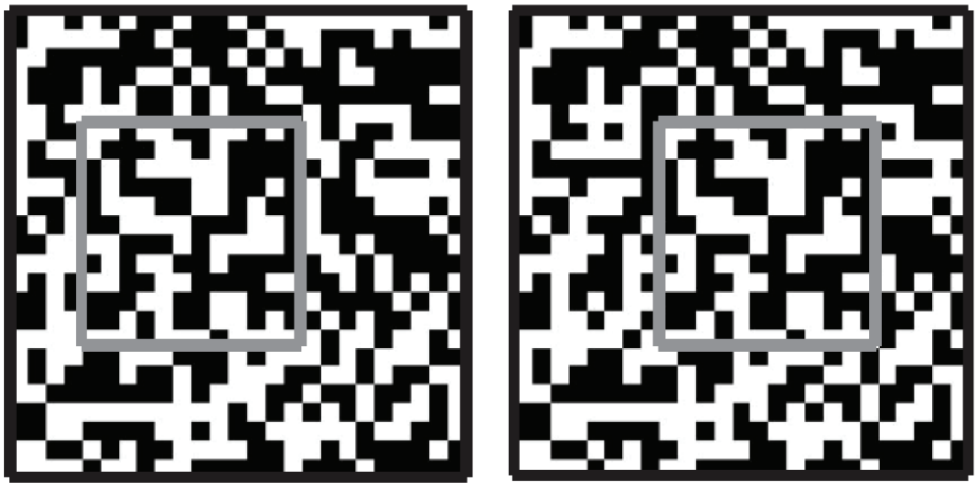


Figure 1.8: An example of random dot stereogram (reprinted from [89])

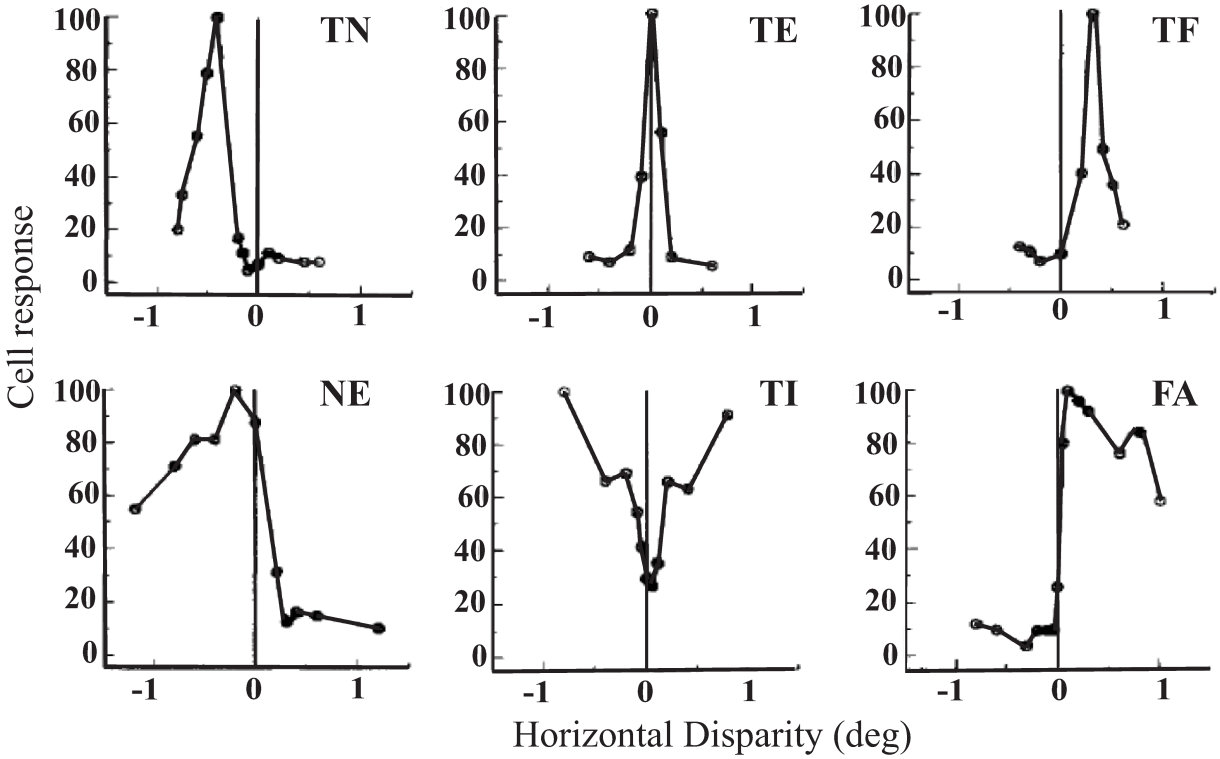


Figure 1.9: Disparity tuning curves for the 6 categories of disparity selective neurons. TN: tuned near, TE: tuned excitatory, TF: tuned far, NE: near, TI: tuned inhibitory, FA: far (reprinted from [36])

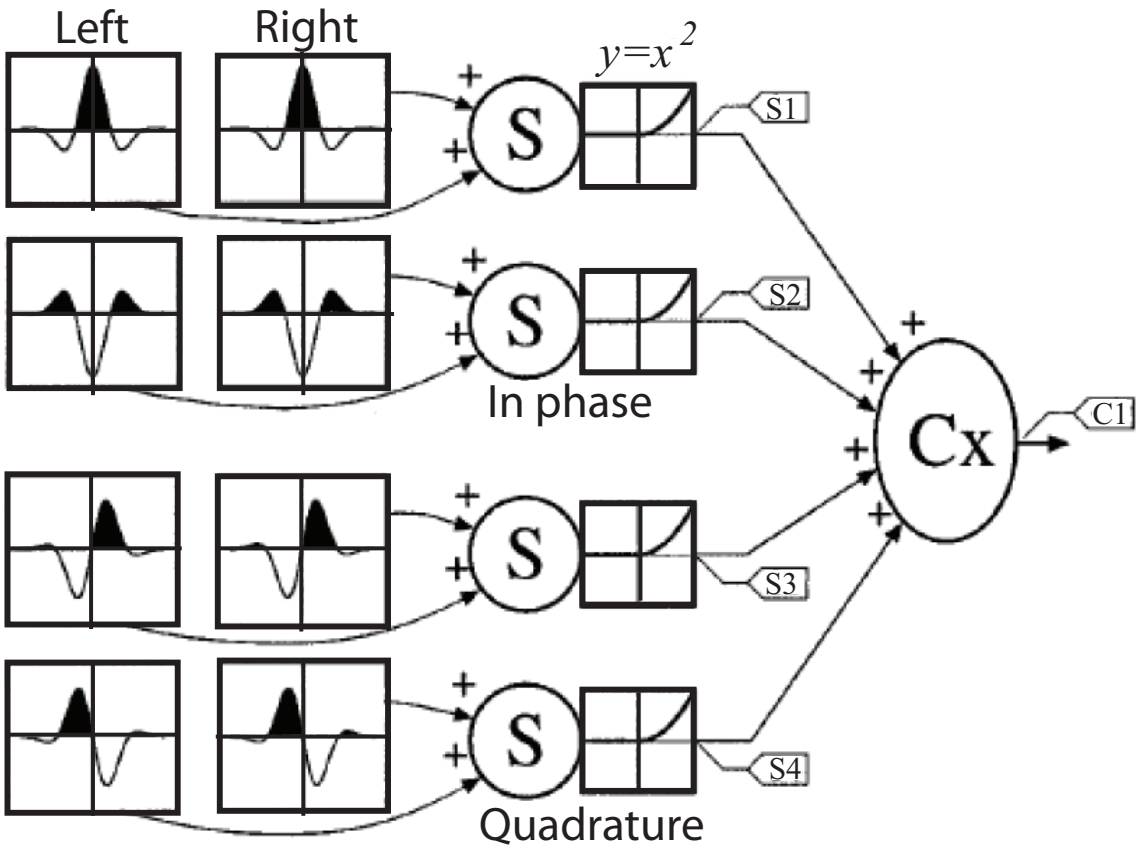


Figure 1.10: Energy Model by Ohzawa et. al. [20] (reprinted from [20])

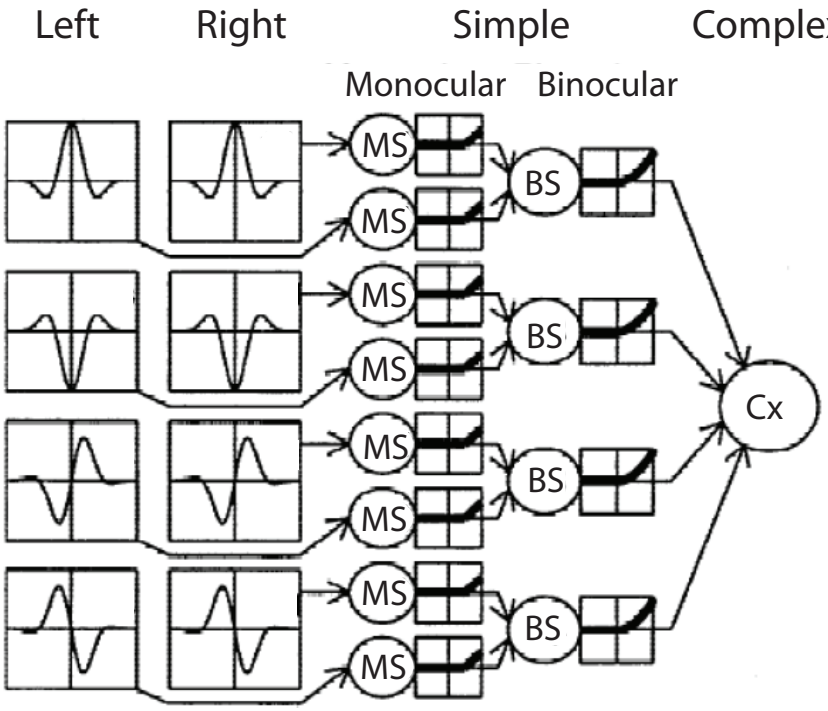


Figure 1.11: Modified Energy Model by Read et. al. [76] (reprinted from [76])

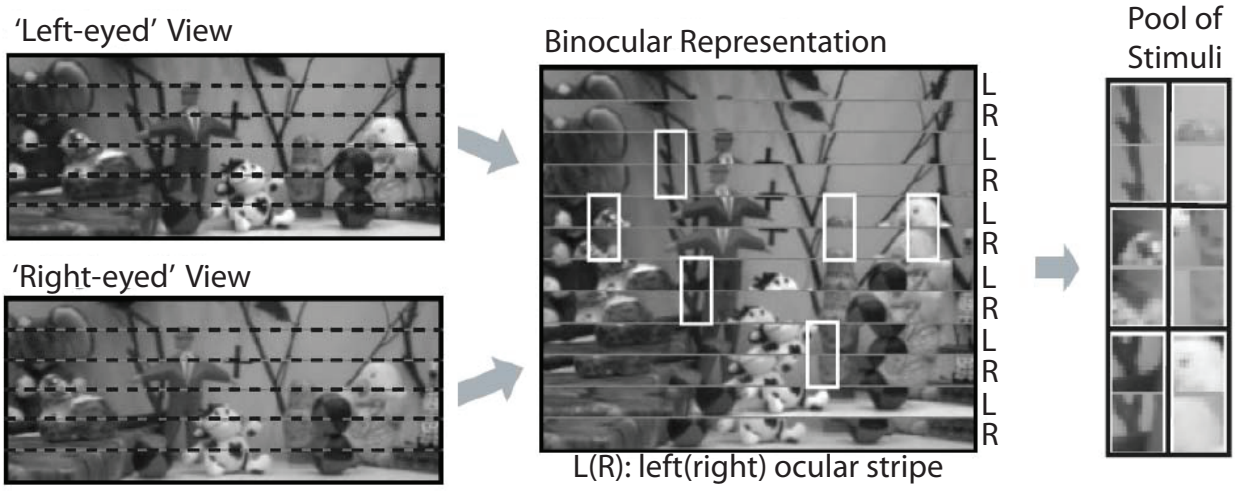


Figure 1.12: Pre-processing to create a pool of stimuli by Wimer et. al. [44] (reprinted from [44])

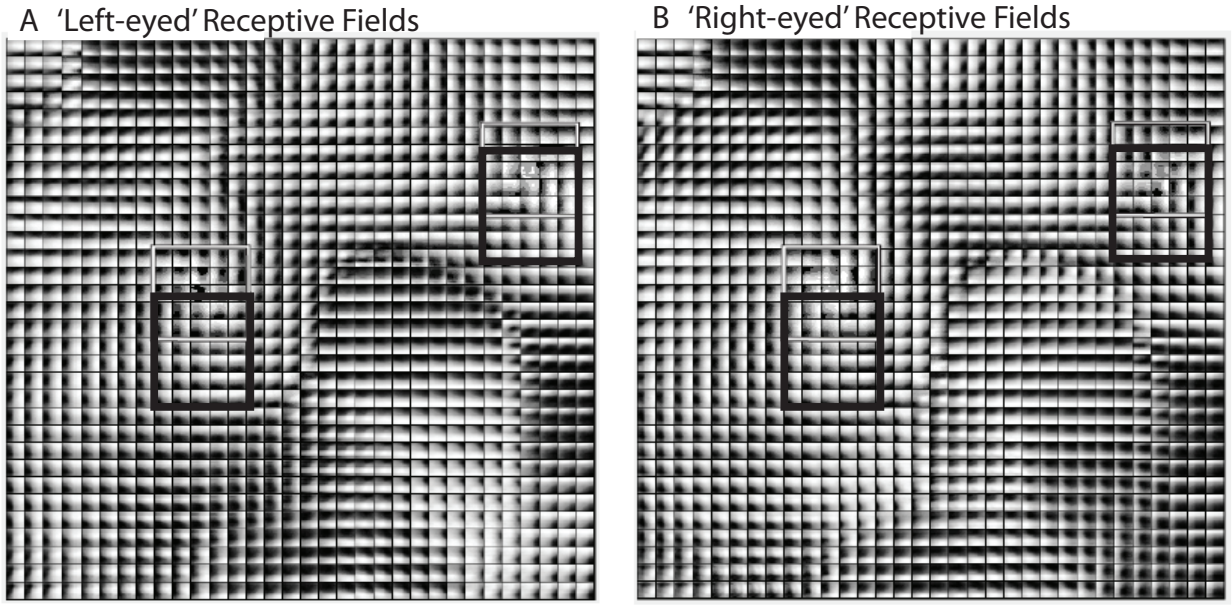


Figure 1.13: Self-organized maps of left and right eye receptive fields (reprinted from [44])

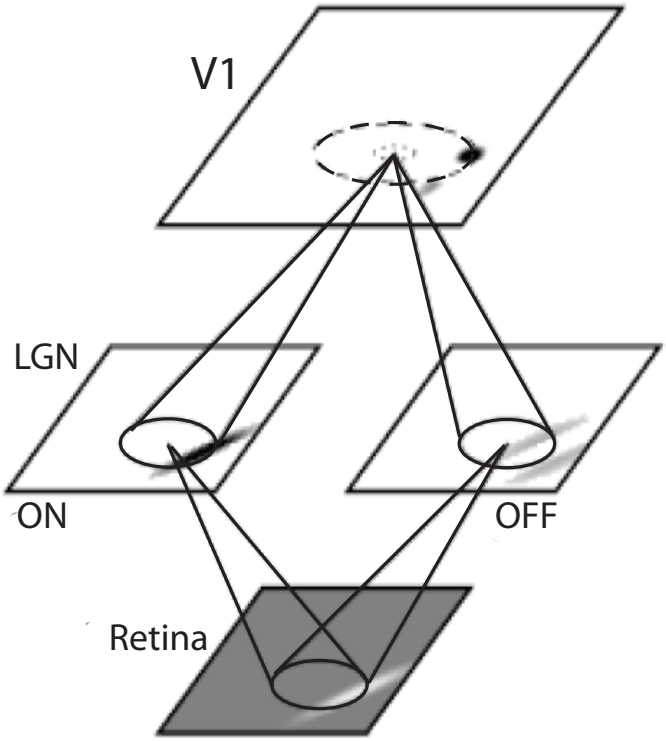


Figure 1.14: Schematic of the architecture for basic LLISOM (reprinted from [68])

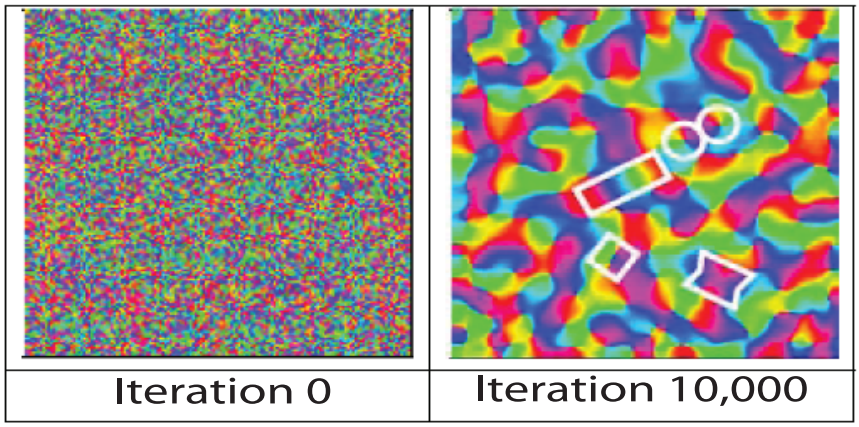


Figure 1.15: Self-organized orientation map in LLISOM (reprinted from [68]). For interpretation of the references to color in this and all other figures, the reader is referred to the electronic version of this dissertation.

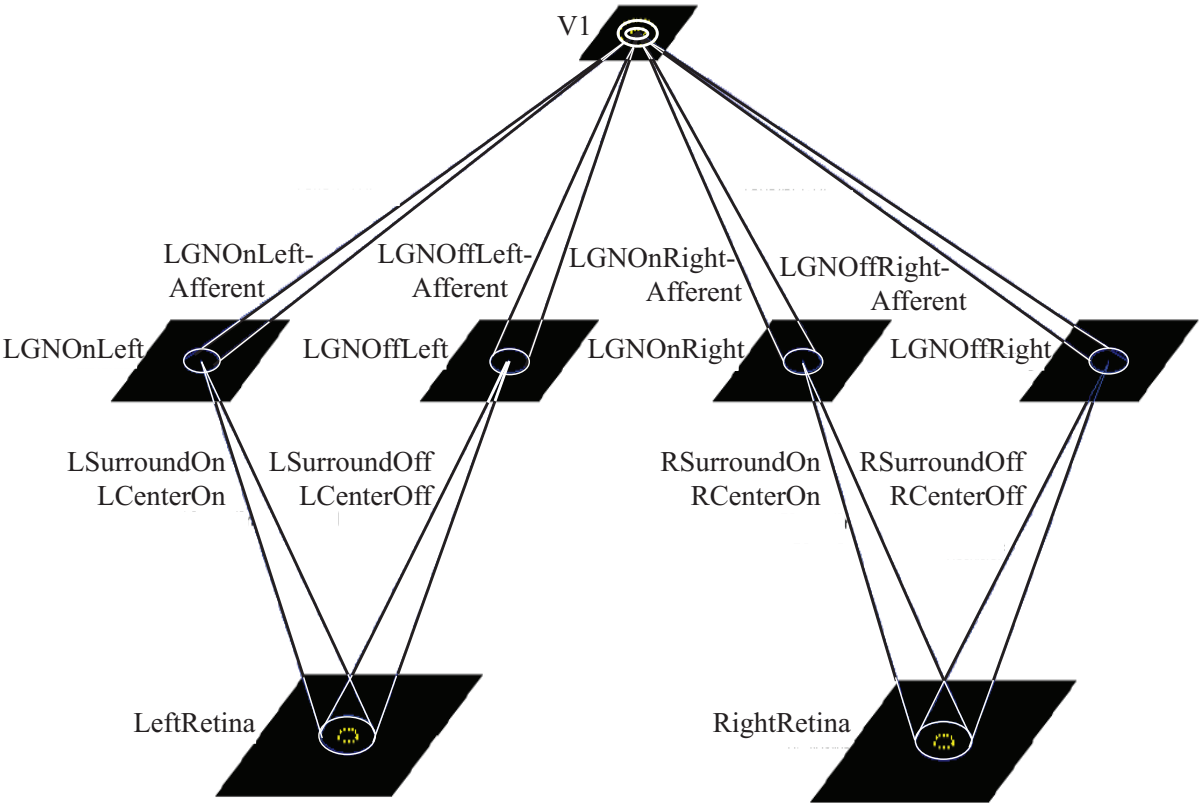


Figure 1.16: Two eye model for self organization of disparity maps in LLISOM (reprinted from [88])

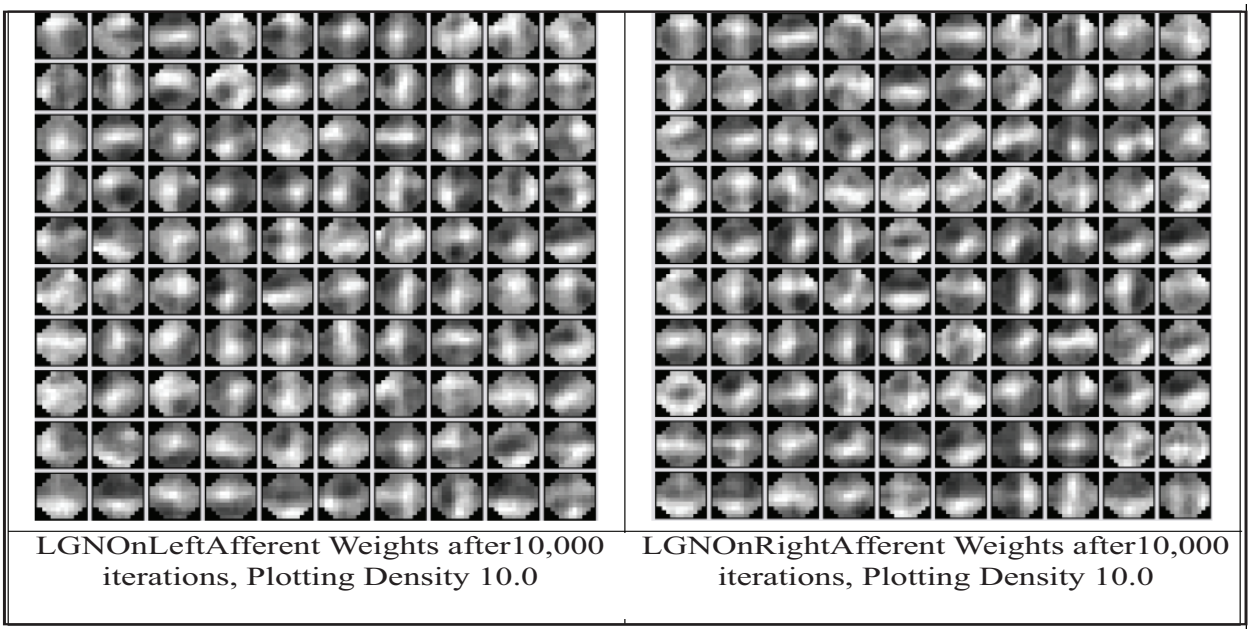


Figure 1.17: Topographic disparity maps generated by LLISOM (reprinted from [88])

Chapter 2

Transfer of Learning in Where-What Networks

The material in this section are adapted from [97]. Please refer to the originial paper for details.

2.1 Introduction to Perceptual Learning

Perceptual Learning (PL) is the long-lasting improvement in perception followed by repeated practice with a stimulus. The fact that low-level sensory perception is still highly plastic in adult humans sheds lights on the underlying mechanisms of learning and plasticity. The subject of PL has long attracted researchers interested in behavioral [34, 24], modeling [35, 103, 95] and physiological [94, 54] implications of perceptual learning.

Conventional paradigms of perceptual learning studies have established the specificity (as opposed to transfer) of PL to the trained stimulus: orientation, direction of motion, eye of presentation, and retinal location (for a review of different tasks see [29, 84, 75, 7, 87, 25, 4, 28, 120, 45]). For example, in a well-known study, [94] observed that the slope of the tuning curve of orientation sensitive neurons in V1 increased only at the trained location. Furthermore, the change was retinotopic and orientation specific. [50] reported that in a

texture discrimination task, PL effects were retinotopically specific, strongly monocular and orientation specific.

In recent years there has been accumulating experimental evidence that has challenged the specificity during perceptual learning, i.e., specificity is not an inherent property of perceptual learning, but rather a function of experimental paradigms (e.g., [118, 121, 122, 78]). As illustrated in Fig. 2.1, there seems to be a general pattern in many of the studies that showed transfer in PL: training the perceptual task in one condition, accompanied by exposure to a second condition results in transfer of learning effects to the second condition. The model of transfer presented in this article is inspired by this general pattern, although we will show that the observed improved performance in transfer condition is a result of gated self-organization mechanisms rather than literal transfer of the information learned for one condition to a novel condition.

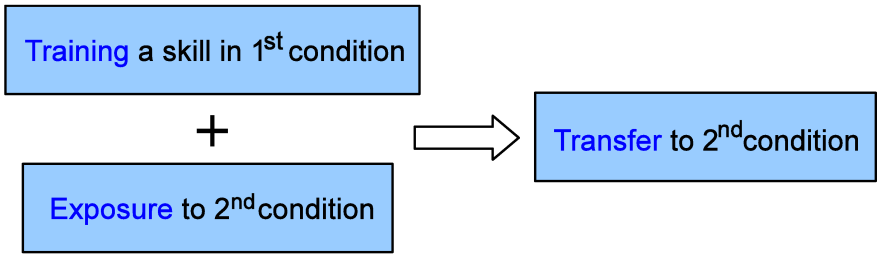


Figure 2.1: General pattern observed in transfer studies. Regardless of the order, a training and an exposure step seem to be common prior to transfer.

Previous models of perceptual learning attribute the improvement in stimulus discrimination to neural modification in either low-level feature representation areas, such as V1, or the connection patterns from the low-level to high-level areas. From a computational point of view, models that predict specificity of training effects are not very difficult to come by. Therefore, not surprisingly, nearly all of the computational models of perceptual learning predict specificity but not transfer.

The first group of models (we call them low-level based, or lower models) are inspired by the retinotopic nature of the lower visual areas, e.g., [2, 123, 102]. These models predict specificity—not transfer—of training effects since stimulus reaches only the parts of the V1 that retinotopically correspond to the specific trained features and locations in the visual field.

The second group of perceptual learning models (we call them reweighting based, or higher models), unlike the first group, assume that discrimination takes place in higher stages (e.g., post V1) of visual processing (e.g., [23, 62, 84]), and perceptual experience improves the readouts from sensory cortex by modifying (reweighting) the connections from low-level representation areas to high-level decision making areas [79, 59]. Since practice with visual stimuli at a certain location and feature reaches only certain connections from low to high-level areas, these models also predict specificity of perceptual learning across locations and features.

How then the neural circuits manage to generalize (transfer) the learning effects to untrained locations and features? As stated above, existing computational models fail to explain this. A rule-based learning model by [121] attempted this important question by assuming that a set of location-invariant or feature-invariant heuristics (i.e., rules) can be learned during perceptual practice, given appropriate experimental settings. This theory lacks neuromorphic level detail, and is not implemented and verified by computer simulation.

We propose a model of perceptual learning, based on the brain-inspired computational framework proposed by [108]. The general assumption of the model is that the brain consists of a cross-connected network of neurons in which most of the modules and their connectivity pattern emerges from neural activities. These assumptions were based on neuroanatomical observations that there are extensive two-way connections between brain areas, and develop-

mental neurobiological studies showing that the brain develops its network in an individual's life time (see, e.g., [26, 49]).

Before providing the details of the model in the next section, we highlight several key aspects of the model that are relevant to PL. In terms of architecture, the model is distinct from existing models by attributing the training effects to not only the improved connections from the sensory to higher cortical areas (e.g., motor areas) but also the improved representations in the sensory cortex due to *neuronal recruitment*. Moreover, in order for transfer to occur, a critical role is assumed for descending (top-down) connections, from motor areas that represent concepts down to adaptively selected internal feature neurons.

In terms of algorithm, we present a rather unconventional and counter-intuitive mechanism for transfer in PL, namely *gated self-organization*. A prevalent assumption in the PL research community seems to be that transfer of learning is caused by the re-use of the representations learned for trained conditions during testing for untrained conditions. Our model, however, does not assume any representational overlap between training and transfer conditions. It assumes a base performance level for the PL task, which simulates the condition where human subjects can always perform at a high level on an easy task without extensive training. The discrimination power existing in this base performance level is improved via gated self-organization as a result of "exposure" effects accumulated during the prolonged training and testing sessions. These mechanisms occur during off-task processes when the model is not actively engaged in a PL task, resulting in performance improvement as significant as those for the trained conditions. In essence, the training sessions merely prime the neuronal circuits corresponding to the untrained conditions to utilize the information already stored in the network (even before the PL training sessions) and bootstrap their performance to the trained level via self-organization.

The model is tested on a simulated Vernier discrimination task. It predicts specificity of training effects under conventional experimental settings, as well as transfer of feature discrimination improvement across retinal locations when the subject is exposed to another stimulus at the transfer location ("double training" per [118]). Although the results presented here are only for the Vernier discrimination task and transfer across locations, the general model presents a detailed network-level explanation of how transfer can happen regardless of task, feature, or location, because the network's developmental mechanisms are independent of stimuli (e.g., Vernier) and outputs of the network (e.g., type, orientation, location, etc.). In other words, since our model is a developmental network in which the internal representations are developed from experience, as opposed to being fixed, pre-designed feature detectors such as Gabor filters, the presented results should in principle generalize to other types of stimuli and experimental settings.

2.2 Model

2.2.1 The overall architecture – Introduction to WWN

Where-What Networks [46] are a visuomotor version of the brain-inspired model outlined in [108], modeling the dorsal (where) stream and the ventral (what) stream of visual and behavioral processing. A major advance from the existing rich studies of the two streams is to attribute the major causality of the "where" and "what" representations to the higher concept areas in the frontal cortex, since motor signals participate in the formation of representations along each stream through top-down connections. That is, each feature neuron represents, not only a bottom-up feature vector $\mathbf{x}$ in the bottom-up source, but instead a joint feature $(\mathbf{x}, \mathbf{z})$ consisting of both bottom-up feature vector $\mathbf{x}$ from receptors and top-down

feature vector **z** from effectors. In order for such a neuron to win the lateral competition and subseqauently fire, its internal representation must match well with both the top-down part of its input signal, **z**, and the bottom-up part of its input signal, **x**.

Where-What Networks (WWN) have been successfully trained to perform a number of tasks such as visual attention and recognition from complex backgrounds [65], stereo vision without explicit feature matching to generate disparity outputs [99] and early language acquisition and language-based generalization [70]. Fig. 2.2 shows a schematic of the version of the network used in this study to model PL as part of an integrated sensorimotor system. The network is developmental in the sense of [111], i.e., none of the internal feature sensitive neurons are pre-designed by the programmer, but rather they are developed (learned) via agent's interactions with the natural stimuli.

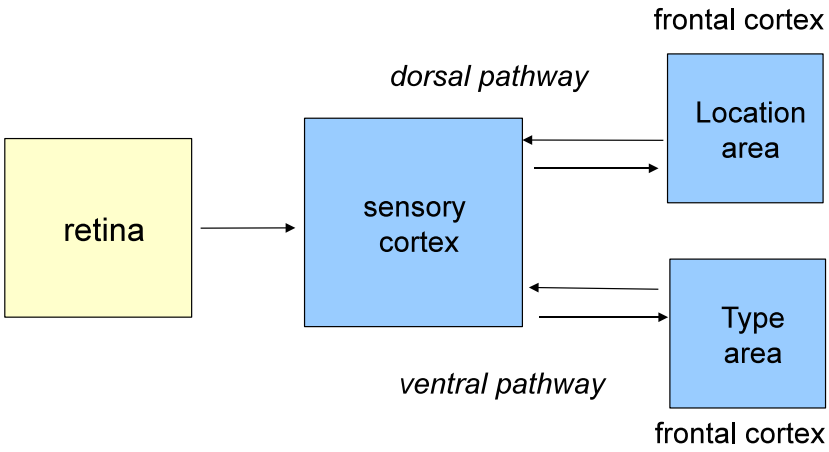


Figure 2.2: A schematic of the Where-What Networks (WWN). It consists of a sensory cortex which is connected to the What area in the ventral pathway and to the Where area in the in the dorsal pathway.

In order for internal network structures to emerge through such interactions, the initial structure of the network does not impose much restrictions. As illustrated in Fig. 2.2, the network consists of one area of neurons modeling the early sensory areas LGN/V1/V2.

The signals then diverge into two pathways; dorsal (or "where") pathway, and ventral (or "what") pathway. The two pathways are bi-directionally connected to the location area and the type area in the frontal cortex, respectively. Unlike the sensory cortex, we assume that the outputs from the location area and the type area can be observed and supervised by teachers (e.g., via the motor areas in the frontal cortex).

The Lobe Component Analysis (LCA) [114] is used as an algorithm for neural learning in a cortical area in WWNs. It uses the Hebbian mechanism to enable each neuron to learn based on the pre-synaptic and post-synaptic activities that are locally available to each synapse. In other words, the learning and operation of WWN do not require a central controller.

In the following subsection, the learning algorithm and signal processing operations in the network are laid out. It is assumed that the network has the overall structure shown in Fig. 2.2. Namely, the internal sensory cortex consists of a 2-dimensional array of cortical columns, laid out in a grid fashion, where each column receives bottom-up input from a local patch on the retina (input image), and has bidirectional connections with all of the neural columns in the concept area. Although the concept areas in the brain have a similar 6-laminar structure, we implemented only a single-layer structure for the concept areas, since there is no top-down input to the concept areas in this simplified model of the brain.

2.2.2 The learning algorithm

The learning algorithm in WWN is inspired by the 6-layer structure of the laminar cortex [12]. The internal area of the network (see Fig. 2.3) consists of a 2D grid of columns of neurons. As shown in Fig. 2.3C, each column has three functional layers (Layers 2, 3 and 4, shown enclosed in dotted rectangles in the figure), as well as three assistant layers (Layers 5a,

5b and 6, not shown for simplicity of illustration). No functional role is assumed for Layer 1, hence not included in the model. We speculate that the computational advantage of the laminar structure of the neocortex is that each area can process its incoming bottom-up and top-down signals separately before combining them. The bottom-up signals first reach Layer 4, where they are *pre-screened* via lateral interaction in the layer assisted by Layer 6. Similarly, the top-down signals are first captured and pre-screened by the lateral interactions in Layer 2, assisted by Layer 5a. The result of these two separate parallel operations is then integrated in Layer 3, processed via the lateral interactions assisted by Layer 5b, and then projected to the next higher level (concept areas in this case). Hebbian learning rule is used for updating the bottom-up weights of Layer 4 and the top-down weights of Layer 2, while all the other connection weights are one-to-one and fixed. Below is a step-by-step algorithmic description of the operations. For simplicity of notations, the time factor, t, is not shown in the equations.

2.2.3 Pre-screening of bottom-up signals in Layer 4

For each i'th neuron, n_i, in Layer 4, the bottom-up weight vector of the neuron, $\mathbf{w}_{b,i}^{(L4)}$, and the bottom-up input to the neuron, $\mathbf{b}_i^{(L4)}$, are normalized and then multiplied. Dot product is used to multiply the two vectors, as it measures the cosine of the angle between the vectors–a measure of similarity and match between two vectors.

$$\hat{z}_i^{(L4)} = \frac{\mathbf{b}_i^{(L4)}}{\|\mathbf{b}_i^{(L4)}\|} \cdot \frac{\mathbf{w}_{b,i}^{(L4)}}{\|\mathbf{w}_{b,i}^{(L4)}\|} \tag{2.1}$$

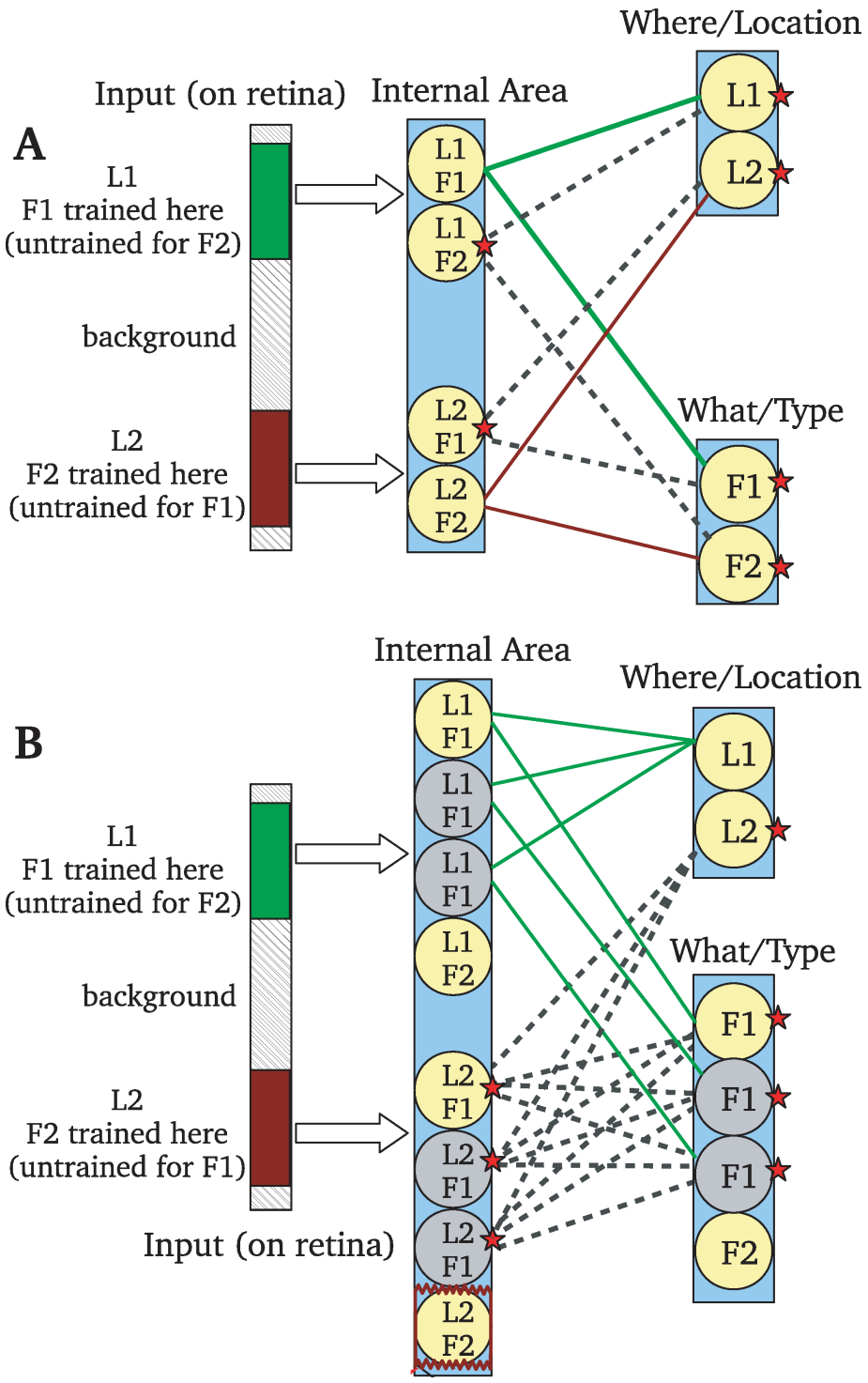


Figure 2.3: How training and exposure accompanied by off-task processes can cause the learning effects to transfer. **(A)** Transfer across locations in Where-What Networks. See the text for explanation. **(B)** Recruitment of more neurons in the sensory and concept areas. Many connections are not shown for the sake of visual simplicity. See text for details.

We call $\hat{z}_i^{(L4)}$ the initial or *pre-response* of the i'th neuron before lateral interactions in the layer. The lateral interactions, which yield the response of the neuron, consist of lateral inhibition and lateral excitation. In the current version of the model, there are no *explicit* lateral connections which makes the algorithms more computationally efficient by avoiding oscillations necessary to stabilize lateral signals while getting essentially the same effects. Lateral inhibition is roughly modeled by the top-k winner rule. i.e., the $k \geq 1$ neurons with the highest pre-response inhibit all the other neurons with lower pre-response from firing—by setting their response values to zero. This process simulates the lateral competition process and was proposed by [32] and [74], among others, who used the term k-winner-takes-all (kWTA). The pre-response of these top-k winners are then multiplied by a linearly declining function of neuron's rank:

$$\hat{z}_i^{(L4)} \leftarrow \frac{k - r_i}{k} \hat{z}_i^{(L4)} \tag{2.2}$$

where $\leftarrow$ denotes the assignment of the value, and $0 \leq r_i < k$ is the rank of the neuron with respect to its pre-response value (the neuron with the highest pre-response has a rank of 0, 2nd most active neuron get the rank of 1, etc.). Each neuron competes with a number of other neurons for its rank, in its local neighborhood in the 2D grid of neurons of the layer. A parameter called *competition window size*, ω, determines the local competitors of the neuron. A competition windows of size $\omega = 5$, centered on the neuron, is used for the reported results. The modulation in Equation 2.2 simulates lateral inhibition among the top-k winners.

2.2.3.1 Pre-screening of top-down signals in Layer 2

The exact same algorithm of pre-screening described above for Layer 4 runs in Layer 2 too. The only difference is that Layer 2 receives top-down signals from a higher area instead of bottom-up input from a lower area.

2.2.3.2 Integration and projection to higher areas in Layer 3

In each cortical column (See Fig. 2.3C), the neuron in Layer 3, n_i, receives the response value of the neuron in Layer 4, b_i, and the neuron in Layer 2, e_i, and sets its pre-response value to be the average of the two values:

$$\hat{z}_i^{(L3)} = \frac{1}{2}(b_i^{(L3)} + e_i^{(L3)}) \tag{2.3}$$

The pre-response value of the Layer 3 neuron, $z_i^{(L3)}$, is then updated after lateral interactions with other neurons in Layer 3, following the exact same algorithm for lateral inhibition described for Layer 4 neurons. For simplicity of terminology, we choose to equate the pre-response and response of Layer 3 with the pre-response and response of the whole column.

To model lateral excitation in the internal area, neuronal columns in the immediate vicinity of each of the k winner columns are also allowed to fire and update their connection weights. In the current implementation, only 8 columns in the 3×3 neighborhood (in the 2D sheet of neuronal columns) are excited. The responses level of the excited columns are set to the response level of the their neighboring winner column, multiplied by an exponentially declining function of their distance (in the 2D grid of columns) to the winner columns:

$$z_i^{(L3)} \leftarrow e^{\frac{-d^2}{2}} z_{winner}^{(L3)} \tag{2.4}$$

where the distance $d = 1$ for immediate neighbors of the winner columns, and $d = \sqrt{2}$ for the diagonal neighbors in the 3×3 neighborhood of the columns. The output of the neurons in Layer 3 are projected to the next higher area (concept areas in the experiments of this article).

2.2.3.3 Hebbian learning in the winning cortical columns

If a cortical column of neurons wins in the multi-step lateral competitions described above and projects signals to higher areas, i.e., if the Layer 3 neuron in the column has a non-zero response value, the adaptable weights of Layer 2 and Layer 4 neurons in the column will be updated using the following Hebbian learning rule:

$$\mathbf{w}_{b,i}^{(L4)} \leftarrow \beta_1 \mathbf{w}_{b,i}^{(L4)} + \beta_2 z_i^{(L4)} \mathbf{b}_i^{(L4)} \tag{2.5}$$

where β_1 and β_2 determine retention and learning rate of the neuron, respectively:

$$\beta_1 = \frac{m_i - 1 - \mu(m_i)}{m_i}, \beta_2 = \frac{1 + \mu(m_i)}{m_i}, \tag{2.6}$$

with $\beta_1 + \beta_2 \equiv 1$. In the equation above, m_i is the column's maturity level (or age) which is initialized to one, i.e., $m_i = 1$ in the beginning, and increments by one, i.e., $m_i \leftarrow m_i + 1$, every time the column wins. The maturity level parameter, m_i, is used to simulate the amount of neural plasticity or "learning rate" in the model. Similar to the brain, the model's plasticity decreases as the maturity level or "age" increases. This is compatible with human development; neural plasticity decreases as people get older.

μ is a monotonically increasing function of m_i that prevents the learning rate β_2 from

converging to zero as m_i increases.

$$\mu(m_i) = \begin{cases} 0, & \text{if } m_i < t_1 \\ c(m_i - t_1)/(t_2 - t_1), & \text{if } t_1 < m_i < t_2 \\ c + (t - t_2)/r, & \text{if } m_i > t_2 \end{cases} \tag{2.7}$$

For the results reported here, we used the typical value $t_1 = 10$, $t_2 = 10^3$, $c = 2$ and $r = 10^4$. See Appendix A for detailed description of these parameters.

Equation 2.5 is an implementation of the Hebbian learning rule. The second term in the right-hand-side of the equation, which implements the learning effect of the current cycle, consists of response of the pre-synaptic firing rate vector, $\mathbf{b}_i^{(L4)}$, multiplied by post-synaptic firing rate, $z_i^{(L4)}$. This insures that a connection weight is strengthened only if the pre- and post-synaptic neurons are firing together, hence, the Hebbian rule.

The same Hebbian learning rule updates the top-down weights of neurons in Layer 2:

$$\mathbf{w}_{e,i}^{(L2)} \leftarrow \beta_1 \mathbf{w}_{e,i}^{(L2)} + \beta_2 z_i^{(L2)} \mathbf{e}_i^{(L2)} \tag{2.8}$$

The neurons in the Where and What concept areas use the same Hebbian learning above for updating their weight vectors. They also utilize the same dot-product rule and lateral interactions for computing their response values. During the times when the firing of a concept neuron is imposed, however, e.g., during supervised training or off-task processes, the response value of each neuron in the concept areas is set to either zero (not firing) or one (firing).

2.2.4 The off-task processes, triggered by exposure

Off-task processes in WWN are the neural interactions during the times when the network is not attending to any stimuli or task. In contrast with most neural network models, WWN runs the off-task processes to simulate the internal neural activities of the brain, even when sensory input is absent or not attended. The off-task processes are run all the time when the network is not in the training mode, e.g., during perceptual learning. As explained in detail below, these processes may or may not alter the network connections, depending on the recent experience of the network.

During the off-task processes, the cortical columns in the internal area operate using the exact same algorithm described in Section 2.2.3.3 while the bottom-up input is irrelevant to the trained and transfer tasks (random pixel background images were used in the current implementation). Similarly, the neurons in the Where and What concept areas operate using the same algorithms. Whether or not a concept neuron fires during off-task processes is a function of the amount of recent exposure of the network to the concept (location or feature) that the neuron is representing.

The probability of a concept neuron firing during off-task processes, given no other neuron is firing in the same concept area, is modeled as a monotonically increasing function, a logistic sigmoid function, of the amount of recent exposure to the corresponding concept. i.e.,

$$p(z_i = 1 | \exists j \neq i,\ z_j = 1) = 0 \tag{2.9}$$

$$p(z_i = 1 | \forall j \neq i,\ z_j = 0) = \frac{2}{1 + e^{-\gamma_i}} - 1 \tag{2.10}$$

where $\gamma_i \geq 0$ measures the amount of recent exposure to the concept that neuron n_i repre-

sents. To simulate lateral inhibition in the concept area, the conditional part of the probabilities in Equations 2.9 and 2.10 models lateral inhibition in the concept areas—it insures that a concept neurons does not fire if there is already another neuron firing in the same area.

The amount of exposure to the i'th concept, γ_i, is formulated as the accumulation of the effect of exposure across trials of experiment. The effect of each trial, in turn, is a decreasing function of the time passed since the trial. i.e.,

$$\gamma_i = \alpha \sum_j e^{-c\Delta t_j} \tag{2.11}$$

where α is a small positive normalization factor, c is a positive constant, and $\Delta t_j = t - t_j$ is the difference between the time of j'th exposure, t_j, and the time of off-task processes, t. This insures that a trial of exposure has the highest effect factor, if it is immediately followed by the off-task processes ($\Delta t_j = 0$), and the effect exponentially decreases in time. In the simulations implemented in this paper, the time t of trials are assumed to have the granularity of one hour. Therefore, the trials of experiments in the same day all have the same time stamp t and the following day gets a time stamp of $t + 24$, etc.

We chose the parameters so that $p(z_i = 1 | \forall j \neq i,\ z_j = 0) \simeq 1$ during off-task processes, if extensive recent (in the scale of several days) exposure has occurred for the i'th combination of location and orientation, and $p(z_i = 1) \simeq 0$ otherwise. See Appendix A for the parameter values used for the reported results.

After the firing of neurons in the concept area are determined from the mechanisms above, the response of the internal area is first computed using the algorithms in Section 2.2.2. Next, the same probabilistic mechanisms above determine whether or not a winner

neuron will fire, depending on its amount of recent exposure. This is an approximate way to simulate the phenomenon that an active brain region tends to keep active for a short while. This could be caused by diffused neurotransmitters such as norepinephrine released from active neurons which indicate a kind of novelty or recency (e.g., [119]).

Using the gated self-organization terminology, the role of exposure, modeled by Equations 2.5, 2.8 and 2.11 above, is to open the gates for the concept neurons corresponding to the exposed conditions to fire in later off-task processes. The nonlinear accumulation of exposure effects in time (Eq. 2.11) can be considered as the "gatekeeper" for self-organization mechanisms during the off-task processes.

2.2.5 How off-task signals and neural recruitment result in transfer

Here, we first explain the idea of transfer in a general form, only its essence while skipping details for simplicity. We then add more details by describing the idea in the context of Where-What Networks for transfer across locations and later adding the neural recruitment explanation.

2.2.5.1 Transfer via "off-task processes"

Expanding on the general pattern discussed in Introduction section —that training in one condition accompanied by exposure to another condition results in transfer to the second condition— we introduce our theory of transfer in three phases:

- **The training phase** performs stimulus-specific and concept-specific learning, e.g., $L1F1$ (green) and $L2F2$ (red) in Fig. 2.3A. This phase establishes associations between

presented stimuli and trained concepts via the internal layer of the network, using the neural mechanisms laid out in Section 2.2.2.

- **The testing phase** was meant for measuring a baseline performance, corresponding to the transfer skill. But, this phase also provides a weak but important memory for the later off-task periods to associate two concepts, e.g., $L1$ with $F2$ and $L2$ with $F1$ shown in Fig. 2.3A. In the general case, the exposure can be active or passive engagement with stimuli. Section 2.2.4 describes how the effect of exposure is systematically implemented in the model during off-task processes.

- **The off-task processes** are assumed to take place while the human subject is not required to perform the training or testing tasks, such as during taking a brief pause. Inside the network, the off-task processes are passes of signals in cortical pathways when no direct sensory signal is present (or attended). The off-task processes cause the concept neurons (e.g., $L1$ and $F2$) and internal (feature) columns of neurons (e.g., $L1F2$) to not only reweight their existing connections, but also grow new ascending and descending connections which did not previously exist, causing new circuits to be developed. These new circuits along with recruitment of new neurons (See Section 2.2.6.2) represent the newly learned network functions that cause performance improvement in the transfer condition.

What we refer to as "off-task processes" is not necessarily processes that could happen when the subject is "off" training sessions. Rather, they are any kind of top-down driven neuronal processes which could occur when the task at hand does not fully occupy the subject's brain. Such processes are mostly unconscious, and could take place during any task pause or even during a trial if the task is not very attentionally demanding.

2.2.5.2 Example: transfer across locations

Here we use an example of transfer of learning across retinal locations to explain how the WWN mechanisms enable the above general theory to be realized. Consider Fig. 2.3A which depicts a WWN. In the following example, we denote the input area (stimulus) as X, the internal area as Y and the concept areas as Z. The concept area consists of two sub-areas Where or Location denoted by Z_L and What or Feature denoted by Z_F. The connections in our model are denoted by the following concise notation:

$$X \rightleftharpoons Y \rightleftharpoons [Z_L, Z_F]$$

where the sign $\rightleftharpoons$ denotes a connection in each of the two directions. Y has bidirectional connections with both areas in $[Z_L, Z_F]$.

Here we present a step-by-step description of the connection changes during the experimental procedure.

2.2.5.3 The training phase

The area X is presented with stimulus $X(L1F1)$, denoting that the feature $F1$ is presented at retinal location $L1$. The subject is taught with two concepts in the area Z, Where area with concept $L1$ denoted by $Z_L(L1)$ and What area with concept $F1$ denoted by $Z_F(F1)$. While top-k firing neuronal columns (see Section 2.2.2) fire in the response pattern $Y(L1F1)$, they connect with the firing neurons in Where and What bidirectionally, due to the Hebbian learning mechanisms. The resulting network connections are denoted as

$$X(L1F1) \rightleftharpoons Y(L1F1) \rightleftharpoons [Z_L(L1), Z_F(F1)] \tag{2.12}$$

and shown by the green links in Fig. 2.3A.

Similarly, in the second part of the training phase, the area X is presented with stimulus $X(L2F2)$, denoting that the feature $F2$ is presented at retinal location $L2$. The resulting network connections are denoted as

$$X(L2F2) \rightleftharpoons Y(L2F2) \rightleftharpoons [Z_L(L2), Z_F(F2)] \tag{2.13}$$

and shown by the red links in Fig. 2.3A.

2.2.5.4 The testing phase

This phase was meant for measuring the baseline performance before the above intensive training sessions, after the intensive training sessions at $L1F1$ and again after intensive training sessions at $L2F2$ for measuring improvements in performance after each step. However, this phase also results in weak but important connections and priming for the off-task processes as discussed below. For example, the area X is presented with feature $F1$ at location $L2$, i.e., $X(L2F1)$, resulting in the following network connections:

$$X(L2F1) \rightleftharpoons Y(L2F1) \rightleftharpoons [Z_L(L2), Z_F(F1)] \tag{2.14}$$

Similarly, the feature $F2$ is also presented at location $L1$, i.e., $X(L1F2)$, resulting in the following weak network connections:

$$X(L1F2) \rightleftharpoons Y(L1F2) \rightleftharpoons [Z_L(L1), Z_F(F2)]. \tag{2.15}$$

2.2.6 The off-task processes

During the off-task processes, the concept neurons which were primed (i.e., fired repeatedly) in training and testing phases spontaneously fire. This spontaneous firing in the absence of relevant stimuli is justified by an accumulated recency effect, formulated in Section 2.2.4. In particular, during the off-task processes around the $L1F1$ sessions (temporal recency), the model "thinks" about $L1$ often which means that $Z_L(L1)$ fires often, which excites $Y(L1F2)$ to fire as a recall using the Z_L-to-Y link in Eq. 2.15, and vice versa. This process is denoted as:

$$Z_L(L1) \rightleftharpoons Y(L1F2). \tag{2.16}$$

Likewise, the model thinks about $F1$ often which means that $Z_F(F1)$ fires often, which excites $Y(L2F1)$ to fire as a recall using the Z_F-to-Y link in Eq. 2.14, and vice versa:

$$Z_F(F1) \rightleftharpoons Y(L2F1). \tag{2.17}$$

Similarly, during the off-task processes around the $L2F2$ sessions, we have:

$$Z_L(L2) \rightleftharpoons Y(L2F1) \tag{2.18}$$

from Eq. 2.14 and

$$Z_F(F2) \rightleftharpoons Y(L1F2) \tag{2.19}$$

from Eq. 2.15. Consequently, the Hebbian mechanism during off-task processes strengthens all the two-way dashed links in Fig. 2.3A. In particular, the neural changes denoted by 2.16 and 2.19 above result in transfer to the untrained condition $L1F2$. Similarly, the

neural changes denoted by 2.17 and 2.18 result in transfer to the untrained condition $L2F1$. Therefore, a double transfer effect takes place.

Such firing of active neurons in the above four expressions not only reweights and strengthens the connections between the corresponding co-firing Y neurons and Z neurons, but also recruit more Y and Z neurons, for improving the representation of the $L1F2$ and $L2F1$ combinations. The newly recruited neuronal columns for condition $L2F1$, for example, are depicted by the two gray circles labeled $L2F1$ in Fig. 2.3B.

2.2.6.1 Example: activation patterns during off-task processes

Here we present a concrete example of activation patterns and neural changes during one iteration of the off-task processes where $F1$ and $L2$ happen to be spontaneously activated. Fig. 2.4 illustrates the example below, while showing only 5×5 neuronal columns in the internal area ($50 \times 50 \times 2$ in the actual implementation), for the sake of explanation.

1. Each neuron in Layer 4 receives bottom-up input from its bottom-up receptive field of 10×10 pixels. During the off-task processes each input pixel value is a random number between 0 and 100 (maximum pixel intensity is 255). Each Layer 4 neuron then computes its "pre-response" according to Eq. 2.1. Then, neurons in Layer 4 with highest pre-response values "win" in lateral competition, and their activation level is scaled depending on their rank, according to Eq. 2.2. All the other neurons in Layer 4 "lose" and their activation level is set to zero.

2. Each neuron in Layer 2 receives a joint top-down input from Where and What concept areas. In this example, top-down input from one neuron corresponding to an instance of concept $F1$, e.g. offset -4 horizontal bars, is one, and input from all the other

What neurons is zero. The same applies to the top-down input from Where neurons; input from $L2$ is one and from $L1$ is zero. The neurons in Layer 2 then compute their pre-response and final activation level via lateral inhibition simulated by ranking and scaling, similar to the operations explained for Layer 4 above.

3. After computations in Layer 4 and Layer 2, each Layer 3 neuron receives two types of input signals; the activation level of the Layer 4 neuron in the same column and the activation of the Layer 2 neuron in the same column. Each Layer 3 neuron then takes the average of these two numbers according to Eq. 3.7, as its pre-response value. This value is also considered as the pre-response for the column that the Layer 3 neuron belongs to.

4. Then each neuronal column computes the rank of its pre-response value among a neighborhood of $5 \times 5 \times 2$ columns. The losing columns are suppressed (activations set to zero) and the winning columns get to scale their pre-response depending on their rank (the same formula as in Eq. 2.2). The winning columns then laterally excite their immediate 3×3 neighboring columns to fire as well. The active columns (winners and their neighbors) then get to update the bottom-up connection weights to their Layer 4 neuron according to Eq. 2.5 and the top-down connection weights to their Layer 2 neuron according to Eq. 2.8.

Improved performance for the condition $L2F1$ (transfer to $L2F1$) is due to the newly modified connections of the winning columns and their neighbors (enclosed in dashed, red square in Fig. 2.4 on the response level of Layer 3 neurons). In our terminology, these columns were "recruited" for the condition $L2F1$ (or recruited more, if they were already recruited), since they develop connections to both concept neurons $L2$ and $F1$. New columns

of neurons are recruited because of lateral excitation. These newly recruited neurons provide necessary representational resources for the untrained, transfer condition to demonstrate improvements as large as the trained conditions. More discussions on this matter will be presented in Section 2.2.6.2. to win and excite its neighbors, both its Layer 4 and Layer 2 neurons must win in the lateral competition (Steps 1 and 2 above).

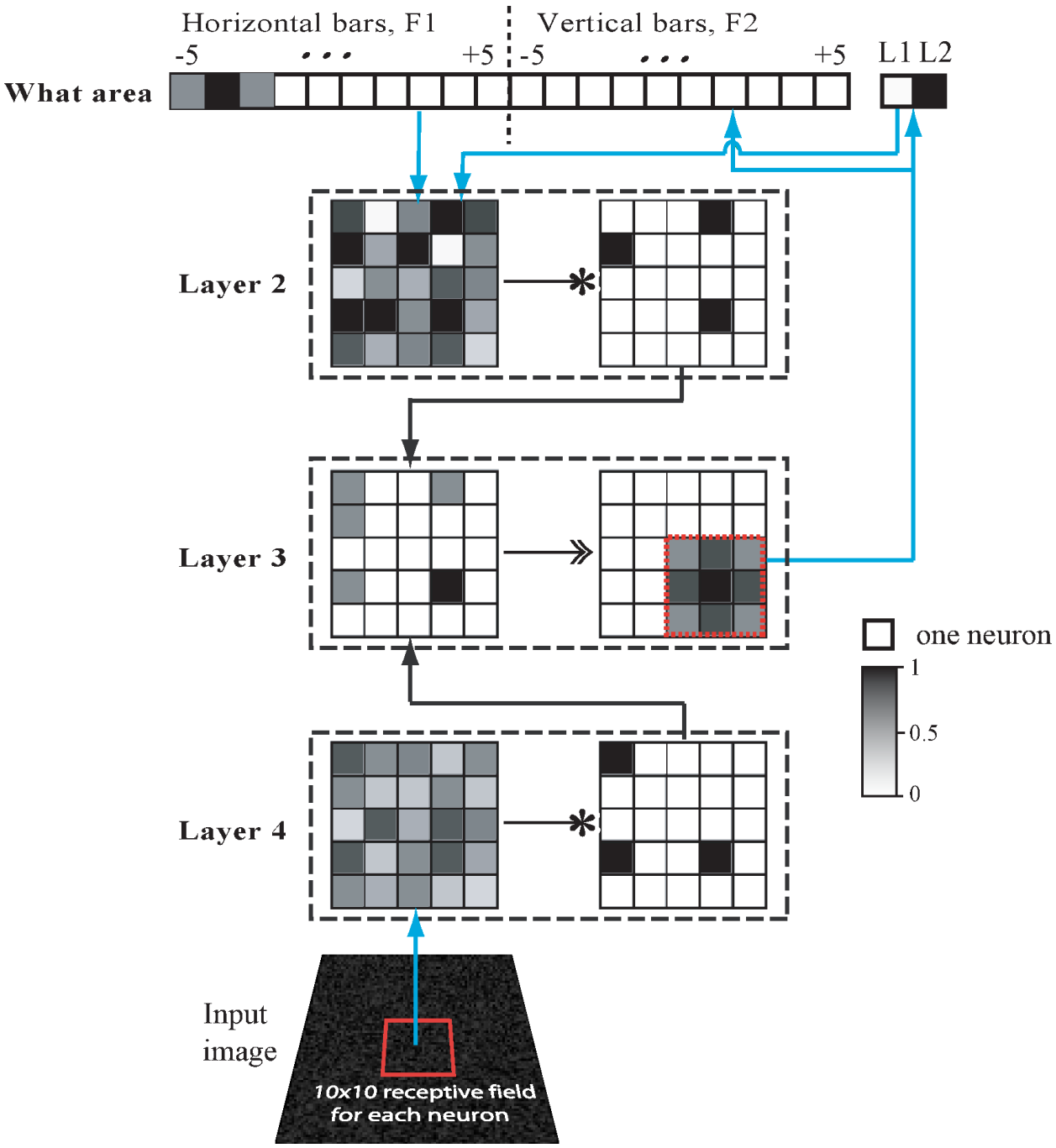


Figure 2.4: An example of activation patterns and neuronal changes during the off-task processes in the network. Only $5 \times 5 = 25$ neuronal columns in the internal area are shown. See Section 2.2.6.1 for a step-by-step description of the neural activation patterns. concept $F1$) and the second neuron in the Where area (corresponding to the concept $L2$) happen to be active.

An experimental example of this type of transfer is [118] where $F1$ is Vernier of vertical orientation and $F2$ is Vernier of horizontal orientation as illustrated in Fig. 2.5. It is important to note that the concepts learned by two concept areas of WWN do not have to be location and Vernier of a certain orientation. At least in principle, a concept area can be taught to represent virtually any concept, such as location, feature type (face, object), scale, color, lighting, and so on.

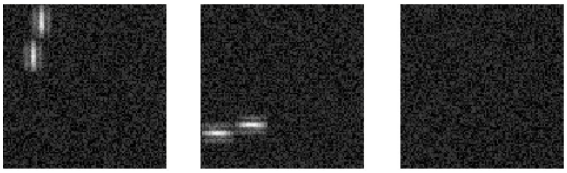


Figure 2.5: Sample images of Vernier input to the model. **(left)** Sample vertical Vernier stimulus at upper left corner ($loc1\_ori1$). **(middle)** Sample horizontal Vernier stimulus at lower left corner ($loc2\_ori2$). **(right)** Background (no input) used as input during network's "off-task" mode.

2.2.6.2 Neural recruitment facilitates fine learning and transfer

The Hebbian learning process among top-winner neuronal columns enables the firing neurons to update their connection weights in a way which makes the neurons more selective for specific inputs (consisting of both bottom-up and top-down components). We say the neuron is "recruited" for that input. The more often a type of inputs (e.g., $L2F1$) is present, the more effective is its recruitment. When an increasing number of neurons are recruited to represent a particular input type, each recruited neuron is more sharply tuned (more sensitive) since more neurons partition the same area of input space. In particular, the lateral excitation in the same area during off-task processes enables multiple winner neuronal columns to fire (instead of a single winner column), which results in recruitment of new representations for the transfer condition, and can lead to transfer effects as large as the direct learning effect.

Fig. 2.3B demonstrates neural recruitment for training the task corresponding to feature $F1$ at location $L1$, and its transfer to the second location $L2$. For the sake of visual simplicity, not all the connections are drawn in the figure. The neuronal columns shown in gray are newly recruited columns, and the two marked $L2F1$ are the columns that are specifically recruited during the off-task processes.

2.3 Simulation

In order to verify the model at the algorithmic level, we implemented WWN to simulate transfer across locations in a Vernier discrimination task, as done by [118]. Similar to the behavioral study by [118], in our simulated experiments the input consisted of two horizontal or vertical Gabor patches presented at upper left or lower left corner of the visual field.

The neuronal resources of the WWN are shown in Fig. 2.2. It has $50 \times 50 \times 2$ (simulated) neuronal columns in the internal sensory area, 20 neurons in the What area and 2 neurons in the Where area. Each of the 20 neurons in the What area are taught for a certain orientation of the Vernier stimulus, vertical or horizontal, and a certain Vernier offset, ranging from -5 to $+5$—excluding zero—pixels (where negative offset indicates left/below, and positive offset indicates right/above). The 2 neurons in the Where area represent the two locations: $loc1$ and $loc2$. There are no excitatory lateral connections between the What neurons corresponding to different tasks, e.g., -5 offset for the vertical Vernier task and -5 degrees for the horizontal Vernier task. There are only *implicit* inhibitory lateral connections between them. i.e., if one is active, it inhibits the other one from being active. Each sensory neuronal column had a bottom-up receptive field of 10×10 pixels, and was fully connected to all the concept neurons, which were in turn fully connected to the neuronal columns in the sensory

cortex. The Vernier input to the network was two Gabor patches with wave length $\lambda = 10$ pixels and the standard deviation of the Gaussian envelope $\sigma = 4$ pixels. The offset of the two Gabors (the amount of misalignments) could be any integer value in range $[-5, +5]$, where the sign of offset (positive or negative) specifies left vs. right (or equivalently, above vs. below), the magnitude of offset is the amount of misalignment in pixels and zero offset denotes perfect alignment. The center of the Vernier stimuli were placed on either $loc1$ at $(r, c) = (20, 20)$ or $loc2$ at $(r, c) = (75, 20)$ in the 100×100 zero intensity image. Then a noise of random intensity in range $[0, 100)$ was added to the image where the maximum intensity was clamped at 255. Fig. 2.5 shows two samples of input to the network.

In the simulation results reported below, we used the method of constant stimuli to train and test our model. This is slightly different from the psychophysical experiments conducted by [118], which used adaptive staircases. This should not affect our results, as our model is based on the Lobe Component Analysis theory [114] in which the effects of training are a function of statistical properties of the stimuli (see Equations 2.5 and 2.8). We do not claim that each trial in our simulation is exactly equivalent to one trial in real human experiments — one neural updating in the model could be many times stronger or weaker than what happens in a human brain. Since the staircase procedure tries to adapt the stimulus difficulty level to the subject's performance, it simply is a more efficient way for training which reduces the number of trials, while the number of "useful" trials (those that contribute to learning) is statistically consistent with the method of constant stimuli with noise. In separate simulations, we have verified that the staircase procedure and method of constant stimuli produced similar results on our network.

The current implementation of the network assumes that teaching signals are available with great precision (a teacher knows the exact offset of the Vernier bars from -5 to $+5$). In

most PL experiments (including the ones in [118]), however, participants are given feedback only on the correctness of their response and not the precise features of the stimuli. We used this standard training regime out of convenience. In fact, we have shown in other works (e.g., [99] in a supervised learning setting and [77] in a reinforcement learning setting) that a probabilistic teaching signal (where, for example, a stimuli is taught to be offset -3 with 0.25 probability, offset -4 with 0.5 probability and -5 with 0.25 probability) also works for our model (indeed sometimes more effectively). Such a training regime is consistent with the intuition that in actual psychophysical experiments, a given offset (e.g., -4) may be mistaken by the participants with the ones that are similar (e.g., -3 and -5), but probably not with the ones that are very different (e.g., -1). Our model is indeed quite tolerant to these small mistakes. Had we used this more realistic training regime, we would also obtain the same basic results, as the particulars of the training regime would not affect the outcome of the training phase and off-task processes.

Below is a step-by-step description of the simulation, in chronological order:

2.3.1 Early development

Due to the developmental design of WWNs, the internal feature representations are not pre-designed, but rather need to emerge while exploiting the sensory input. Therefore, a $50 \times 50 \times 2$ array of naive (with random connection weights) simulated cortical columns were presented with natural images in an unsupervised mode. To develop stable representations 10^4 natural image patches (randomly selected from larger images) of size 100×100 were used. Fig. 2.6 shows that the neuronal columns in this stage develop features that resemble oriented edges and blobs.

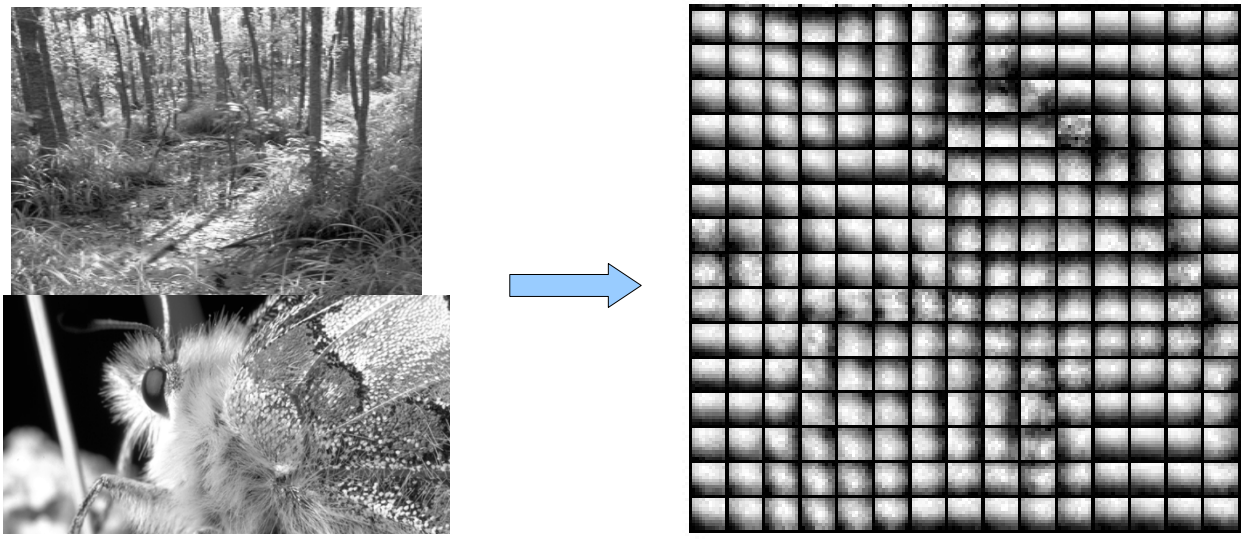


Figure 2.6: **(left)** Sample natural scene images used in early development step of the simulation. **(right)** Bottom-up weight vectors (receptive field profile) of 15×15 sensory neurons developed after exposure to natural images.

2.3.2 Coarse training

Adult human subjects understand the concept of left vs. right and above vs. below. Therefore, even naive subjects can do the Vernier discrimination task when the offset is large. In order to enable the WWN develop this capability, we trained it for only easy (or "coarse") discrimination at offsets $-5, -4, +4, +5$, ten times for each offset at both locations and both orientations. As a result, the model developed the capability to successfully discriminate the Vernier input at both locations and orientations, if their offsets were large enough. The dashed curve in Fig. 2.7 shows the performance of the WWN after this step.

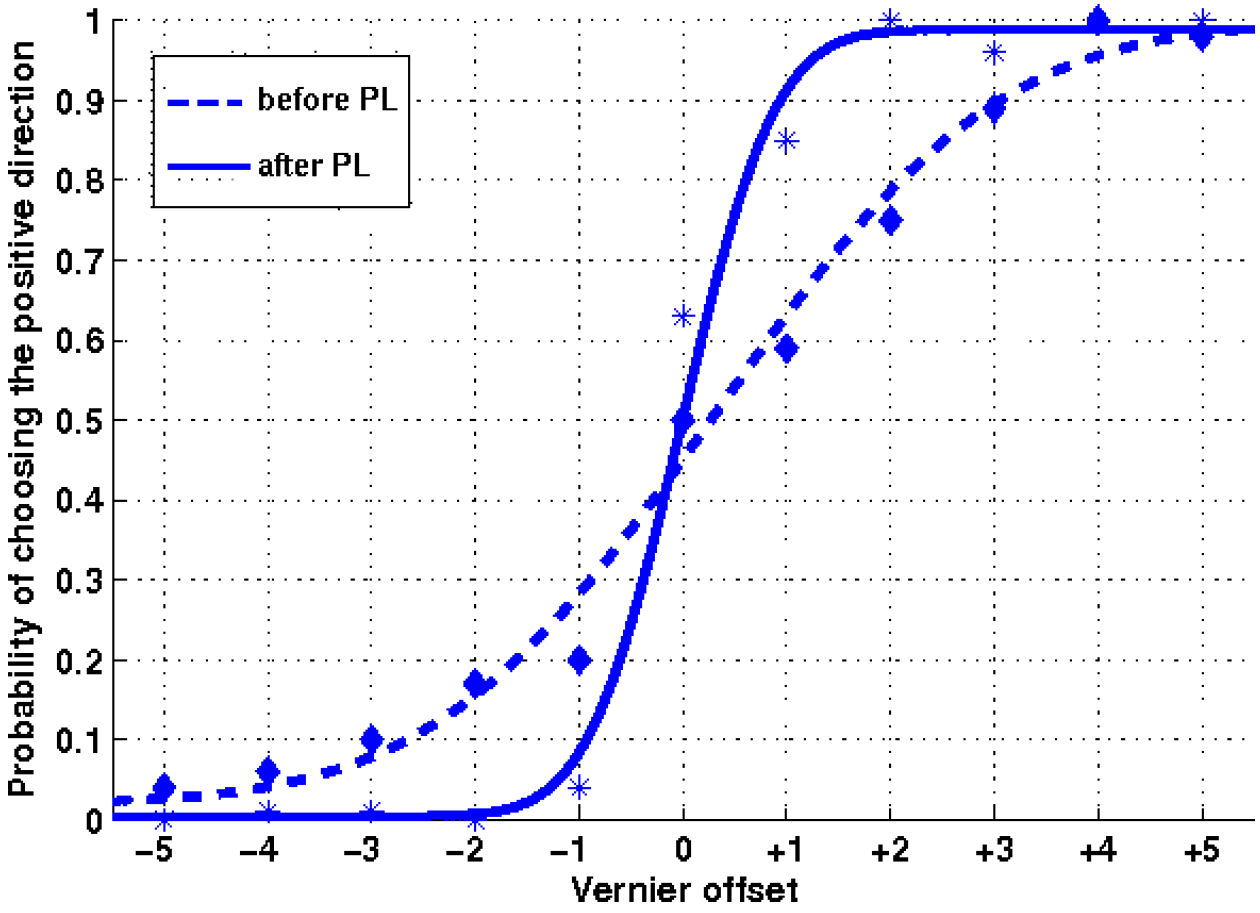


Figure 2.7: Psychometric function for the network's performance before and after perceptual learning.

2.3.3 Perceptual learning at $loc1\_ori1$ and $loc2\_ori2$

Each of the offsets in range $[-5, +5]$, excluding 0, were trained 100 times at $loc1\_ori1$ for 5 consecutive sessions. Then the same training procedure was simulated for Vernier stimuli at $loc2\_ori2$. The training regime followed the double-training procedure employed in [118].

2.3.4 Off-task processes and transfer

In the last step of the simulation, new and improved connections between feature neurons at $loc1$ and concept neurons for $ori2$ were formed during off-task processes, due to spontaneous

firing of these neurons and the Hebbian rule. Therefore, the skill originally learned for $loc2\_ori2$ was transferred to $loc1\_ori2$. In a similar scenario, the skill learned for $loc1\_ori1$ was transferred to $loc2\_ori1$. Here we explain the details of the off-task processes in our simulation.

To simulate the off-task processes in each iteration of our simulations, one of the 20 neurons in the What area, say neuron z_1, and one of the 2 neurons in the Where area, say neuron z_2, were selected to fire (their output value was set to 1) and the remaining concept neurons were imposed not to fire (their output value was set to 0). The selection of firing neurons during the off-task processes was based on the exposure-dependent probabilities in Equations 2.9 and 2.10 explained in Section 2.2.4. Input to the network was a 100×100 noise background (see Fig. 2.5, right). The random noisy input guaranteed nondeterministic network behavior, while providing unbiased excitation to sensory neurons, i.e., the background input did not influence the discrimination behavior of the network, since all the sensory neuronal columns received equal bottom-up excitation on average.

Top-down connections from the concept areas to the sensory area, however, selectively excited a subset of sensory neurons. The sensory neuronal columns excited by both active What and active Where neurons were more likely to win in lateral competition (see Section 2.2.2). Let us denote the set of winner neuronal columns in the sensory area by Y_1. Due to the Hebbian nature of our learning rule (Equations 2.5 and 2.8), repetition of this process caused the weight of connections between Y_1 and z_1 and connections between Y_1 and z_2 to increase and eventually converge to similar weight vectors for both concept neurons.

Similar connections from Y_1 in the sensory area and z_1 and z_2 in concept areas helped to increase the likelihood for z_1 and z_2 to fire together, since they receive similar bottom-up input. In other words, although z_1 and z_2 are not directly connected, and therefore, cannot

excite each other, they become indirectly connected via Y_1 neuronal columns in the sensory area, and after completion of these off-task processes they more frequently fire together. If one of the concept neurons has been trained on a particular stimulus prior to off-task processes, say z_1 was trained on Feature 1, then its more frequent simultaneous firing with z_2 after the off-task processes stage is behaviorally interpreted as transfer of training effects to z_2, say Location 2. The processes explained above was repeated until complete transfer was achieved.

Moreover, short-range lateral excitation in sensory area caused the sensory neuronal columns close to Y_1 in neuronal plane, to also fire and get connected to the concept neurons z_1 and z_2 during the off-task processes. This results in extended representation (allocation of more neuronal resources) for the concepts encoded by z_1 and z_2. The newly allocated representations are slight variations of the old representations which result in more inclusive and discriminative coverage of the stimulus space by sensory neurons, and hence improved performance when a similar stimulus is presented. Our simulations showed that allocation of more resources was necessary in order for complete transfer to happen. It is worth to mention that the learning algorithm of the network was not intervened by the programmer in order for neural recruitment to happen. It was rather a mere consequence of the same Hebbian (reweighting) rule (see Equations 2.5 and 2.8) during training and off-task processes.

2.4 Results

2.4.1 Basic perceptual learning effect

In general, the model exhibited a graded response to different Vernier offsets, with near-perfect performance at large Vernier offsets and near-chance performance as the offset ap-

proached zero (Fig. 2.7), similar to human observers. We fitted the performance data with a Weibull function using psignifit [116] and plotted the resulting psychometric functions in Fig. 2.7. After the initial coarse training, the slope of the psychometric function was relatively shallow (dashed curve); after perceptual learning (fine discrimination), the psychometric function became steeper (solid curve), indicating improved discriminability. We defined threshold as the difference in offset between 0.25 and 0.75 response probability.

2.4.2 Specificity and transfer of perceptual learning

The pre-testing threshold (right after the coarse training step) for all the four combinations of location and orientation were similar, at slightly less than 2.2 pixels (first four points in Fig. 2.8A). In the first training phase ($loc1\_ori1$), the threshold decreased consistently across sessions with smaller decreases in later sessions. At the end of this training phase, threshold in the other three conditions were measured and were found to be close to their pre-testing counterparts (first three points in Fig. 2.8B). This result shows that the specificity of learning in our model, i.e., training in $loc1\_ori1$ does not transfer to untrained location or orientation, as we expected.

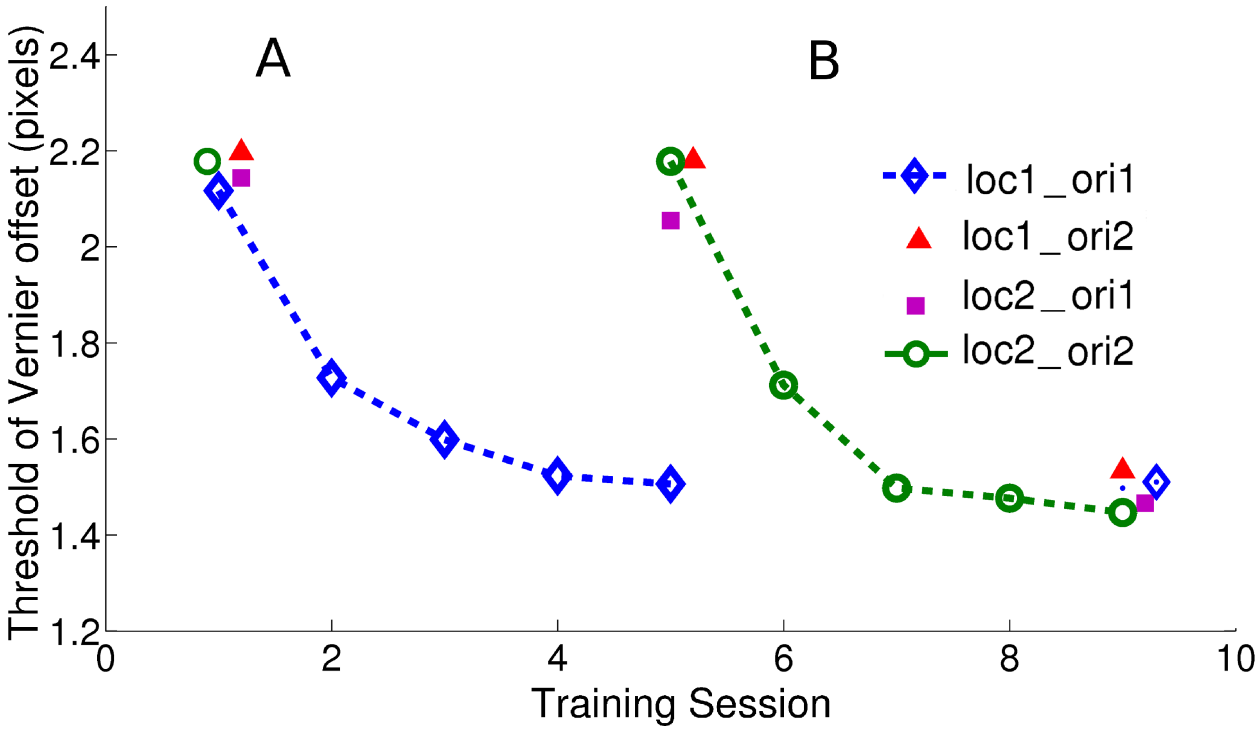


Figure 2.8: Performance of the WWN model - perceptual learning and transfer effects. **(A)** All the four combinations of orientation and location were first pre-tested to measure their threshold, and then in Phase 1, *loc*1_*ori*1 condition. The blue curve shows the decreases in threshold for the trained condition. **(B)** Testing for the three untrained conditions shows no change in their corresponding thresholds at the end of *loc*1_*ori*1 (no transfer). Threshold decreases for *loc*2_*ori*2 as a result of training (green curve). At the end of the 9th training session, threshold for the two untrained conditions *loc*1_*ori*2 and *loc*2_*ori*1 drops to the same level as the trained conditions. **(C)** Percentage of improvement in discrimination after training and transfer. It plots the same data as in (A) and (B). Hollow and filled bars show relative improvement as a result of training and transfer, respectively. See Figure 3C and 3D in [118] for comparison.

The training of *loc*2_*ori*2 then followed and resulted in a similar decrease in the threshold over five sessions. After this second training session, the off-task processes were run to simulate what typically happens with a human subject. Finally, the threshold for all conditions were measured again. Importantly, the threshold for untrained conditions *loc*1_*ori*2 and *loc*2_*ori*1 were at the same level as the trained conditions (the last four points in 2.8B), demonstrating effective transfers as we expected. We calculated the percentage of improve-

ment in performance, defined as $100 \times \frac{T_b - Ta}{T_b}$ in percentage, where T_b is the threshold before the phase is started and T_a is the threshold after the phase is completed (Fig. 2.8C and 2.8D). Specificity was evident after the first training phase (Fig. 2.8C), whereas nearly complete transfer occurred after the second training phase (Fig. 2.8D). Thus, the model showed transfer of the perceptual learning effect across retinal locations in a Vernier task, capturing the basic pattern of results in the original behavioral study of [118].

2.4.3 Reweighting versus change in sensory representation

As mentioned in Introduction, a major debate among PL theories is the neuronal locus of changes that result in performance improvement, i.e., change in sensory representation versus reweighting of connections from sensory to higher level areas. Since all the neurons in our model have plastic weights which change according to the same LCA updating rule, performance improvement is expected to be attributed to change in both the sensory area and higher concept areas. To quantify this change, we define the following metric:

$$d_{i,j} = (w_{i,j}^{preL} - w_{i,j}^{postT})^2 \tag{2.20}$$

where $d_{i,j}$ is the amount of change in the connection weight from neuron n_i to neuron n_j (or a neuron to a sensory pixel), and $w_{i,j}^{preL}$ and $w_{i,j}^{postT}$ are the corresponding weight values after the pre-training (Section 3.2) and post-transfer (Section 3.4) stages, respectively. In the sensory area, only neurons which have overlapping receptive field with the Vernier stimulus were counted in this measurement. When we normalized all the weights to $[0, 1]$ range, the average amount of change for sensory neurons was $d_{sensory} = 0.0098$ while the average value for ascending and descending connections between the sensory area and the concept areas

was $d_{reweighting} = 0.247$. This substantial difference in the amount of change in sensory and higher areas shows that reweighting of sensory readouts is mainly responsible for performance improvement in our model.

2.5 Discussion

In this study, we showed that the WWN learning model of the cerebral cortex is able to demonstrate both specificity and transfer of perceptual learning effects, using representations of sensory and motor signals, which are emergent from the simple rules of Hebbian learning and lateral inhibition and excitation. Similar to previous models of perceptual learning, our work showed performance improvement following extensive practice with the stimulus, where the training effects were specific to the trained feature and location. Our focus on this study, however, was explaining how training effects can transfer (generalize) to untrained but related situations. Although the WWN model was trained and tested only for the Vernier discrimination task, the underlying mechanism for transfer should be applicable to other types of stimuli and tasks, at least in principle.

2.5.1 Top-down and off-task processes and neuronal recruitment in PL

A number of studies have shown the importance of higher brain areas and top-down signals in perceptual learning context. For example, [54] trained monkeys to determine the direction of moving visual stimuli while recording from MT (e.g., representing motion information) and LIP (e.g., representing transformation of motion into action). Their results showed that improved sensitivity to motion signals was correlated with neural responses in LIP, more so

than MT. Hence, at least in learning to discriminate motion direction, performance seems to rely on information from higher areas such as LIP, in addition to information in early sensory areas such as MT. Similarly, experiments by [57] showed that task experience can have a long lasting modulation effect on behavior as well as the response of V1 neurons in a perceptual learning task.

Despite the strong evidence for the important role of top-down and off-task signals in perceptual learning, the prior models are either strictly feed-forward networks (e.g., [84]) or a two-way cascade of internal areas with unsupervised learning, without top-down supervision [40]. Our model, presented in this article, explains transfer of learning effects utilizing top-down and off-task signals as well as the mechanisms of neural recruitment. The model we present here predicts that off-task processes are also essential for generalization in learning, i.e., transfer to novel situations.

Exposure to the untrained situations makes the feature representation neurons corresponding to those situations to fire during off-task time. Co-firing of those representation neurons with the trained concept neurons result in improvement of connections between them, through the Hebbian learning rule. These improved connections as well as recruitment of new neurons to represent the transfer feature, result in transfer of training effects to the untrained situations.

Recruitment of new representations is another important component of our model that helps the transfer of learning. Many previous studies show that neurogenesis is related to the acquisition of new knowledge [72, 52, 51]. Similar to the case of off-task processes, our model predicts that neuronal recruitment is an essential element of transfer/generalization of learning, in addition to being essential for learning itself to take place.

2.5.2 Previous models

A prominent model in perceptual learning literature is the Hebbian channel reweighting model by [23]. [80] expanded this model, and conducted a focused study on the locus of perceptual learning (representation (lower) versus decision (higher) areas). Using a simple model consisting of a layer of fixed feature detectors in the representation area and a linear classifier, Perceptron [93], they suggested that perceptual learning may involve reweighting of connections from lower representation areas to higher decision areas in their feed-forward model with optionally only inhibitory feedbacks. Their relatively simple model used fixed representation of sensory data making their model unable to predict plasticity in stimulus representation in lower visual areas reported by, e.g., [94, 55]. Moreover, their lack of top-down connections from higher areas to representation areas is inconsistent with overwhelming neuroanatomic evidence , as reviewed by, e.g., [27].

Similar to [80], several previous models which have been tested on different perceptual learning tasks (e.g., [104] on motion perception, [124] on bisection, [106] on Vernier hyperacuity) rely on reweighting of connections from sensory representation to concept areas to explain learning effects. In another influential model, Reverse Hierarchy Theory, [3] suggested that lower representations in visual hierarchy are not to be altered unless necessary. Without feedbacks from concept to sensory areas, these feedforward models cannot explain transfers.

Unlike previous feed-forward networks, our model suggested that within a fully developed network (simulating an adult human brain), the lower representations still change, not only because of the exposure to the stimuli, but also due to the overt and covert actions of the subject, projected via top-down connections.

2.6 Conclusion

In summary, the WWN model presented in this article bears analogy to previous models of PL in a number of aspects, including incremental reweighting of connections from sensory areas to concept areas via biologically-plausible Hebbian rule and having the selective reweighting of connections to account for performance improvement after training. However, we present a more extensive brain-anatomy inspired model that goes beyond the previous models in several aspects, including: **(a)** novel approach to viewing transfer as a result of gated self-organization rather than literal transfer of relational information, **(b)** fully developed feature representations emerged from presentation of natural image stimuli to the network as well as top-down signals, as opposed to hand-designed filters (e.g., Gabor filters), **(c)** adaptive and constantly re-weighted connections for neurons which have both top-down and bottom-up components, in contrast with exclusively feed-forward network design, **(d)** modeling both the Where (dorsal) and the What (ventral) visual pathways in an integrated functional system. The model attributes the development of such pathways to top-down connections from the corresponding concept areas in the frontal cortex, going beyond the classical sensory account of the two streams [69], **(e)** the computational model for the 6-layer laminar architecture in the WWN network, **(f)** the proposal of the off-task processes and showing their critical role in transfer, **(g)** the analysis of the dynamic recruitment of more neurons during learning and transfer, and demonstration through the sharpening of the neuronal tuning curves, to account for the improved performance. The last two aspects of the model (off-task processes and neuronal recruitment) were the key new mechanisms in our model that caused transfer of learning effects to untrained conditions.

Chapter 3

Disparity Detection on Natural Images—Shifted Horizontal Image Stripes

The material in this section are adapted from [99]. Please refer to the originial paper for details.

How our brains develop disparity tuned V1 and V2 cells and then integrate binocular disparity into 3-D perception of the visual world is still largely a mystery. Moreover, computational models that take into account the role of the 6-layer architecture of the laminar cortex and temporal aspects of visual stimuli are elusive for stereo. In this paper, we present cortex-inspired computational models that simulate the development of stereo receptive fields, and use developed disparity sensitive neurons to estimate binocular disparity. Not only do the results show that the use of top-down signals in the form of supervision or temporal context greatly improves the performance of the networks, but also results in biologically compatible cortical maps – the representation of disparity selectivity is grouped, and changes gradually along the cortex. To our knowledge, this work is the first neuromorphic, end-to-end model of laminar cortex that integrates temporal context to develop internal representation, and generates accurate motor actions in the challenging problem

of detecting disparity in binocular natural images. The networks reach a sub-pixel average error in regression, and 0.90 success rate in classification, given limited resources.

3.1 Introduction

The past few decades of engineering efforts to solve the problem of stereo vision proves that the computational challenges of binocular disparity are far from trivial. In particular, the correspondence problem is extremely challenging considering difficulties such as featureless areas, occlusion, etc. Further, the existing engineering methods for binocular matching are not only computationally expensive, but also hard to integrate other visual cues to help the perception of depth. It is important to look at the problem from a different angle – How the brain solves the problem of binocular vision? In particular, what are the computational mechanisms that regulate the development of the visual nervous system, and what are the role of gene-regulated cortical architecture and spatiotemporal aspects of such mechanisms?

Although steropsis seems to be a spatial problem, the temporal information appears to help stereopsis due to the physical continuity underlying the physicality of dynamics. Biological agents exploit spatial and temporal continuity of the visual stimuli to enhance their visual perception. In the real world, objects do not come into and disappear from the field of view randomly, but rather, they typically move continuously across the field of view, given their motion is not too fast for the brain to respond. At the pixel level, however, values are very discontinuous as image patches sweep across the field of view. Our model assumes that visual stimuli are largely spatially continuous. Motivated by the cerebral cortex, it utilizes the temporal context in the later cortical areas, including the intermediate areas and motor output area, to guide the development of earlier areas (In Section 3.2.2 Eq. 3.4 the

activation level of the neurons from the *previous time step* is used to supervise $L2$.). These later areas are more *"abstract"* than the pixel level, and thus provide needed information as temporal context. However, how to use such emergent information is a great challenge.

Among the different stages of the explicit matching approaches, the *correspondence problem* is believed to be the most challenging step; i.e. the problem of matching each pixel of one image to a pixel in the other [67]. Solutions to the correspondence problem have been explored using area-, feature-, pixel- and phase-based, as well as Bayesian approaches [22]. While those approaches have obtained limited success in special problems, it is becoming increasingly clear that they are not robust against wide variations in object surface properties and lighting conditions [30].

The network learning approaches in category (3) do not require a match between the left and right elements. Instead, the binocular stimuli with a specific disparity are matched with binocular neurons in the form of neuronal responses. Different neurons have developed different preferred patterns of weights, each pattern indicating the spatial pattern of the left and right receptive fields. Thus, the response of a neuron indicates a degree of match of two receptive fields, left and right. In other words, both texture and binocular disparity are measured by a neuronal response - a great advantage for integration of binocular disparity and spatial pattern recognition.

However, existing networks that have been applied to binocular stimuli are either bottom-up Self-Organizing Maps (SOM) type or error-back propagation type. There has been no biological evidence to support error back-propagation, but the Hebbian type of learning has been supported by the Spike-Time Dependent Plasticity (STDP) [17]. SOM type of networks that use both top-down and bottom-up inputs has not be studied until recently [92, 96, 109, 110]. In this paper we show that top-down connections that carry supervisory disparity

information (e.g. when a monkey reaches an apple) enable neurons to self-organize according to not only bottom-up input, but also supervised disparity information. Consequently, the neurons that are tuned to similar disparities are grouped in nearby areas in the neural plane, forming what is called topographic class maps, a concept first discovered in 2007 [66]. Further, we experimentally show that such a disparity based internal topographic grouping leads to improved disparity classification.

Neurophysiological studies (e.g. [13] and [12]) have shown that the primary visual cortex in macaque monkeys and cats has a laminar structure with a local circuitry similar to our model in Fig. 3.3. However, a computational model that explains how this laminar architecture contributes to classification and regression was unknown. LAMINART [86] presented a schematic model of the 6-layer circuitry, accompanied with simulation results that explained how top-down attentional enhancement in V1 can laterally propagate along a traced curve, and also how contrast-sensitive perceptual grouping is carried out in V1. Weng et. al. 2007 [61] reported performance of the laminar cortical architecture for classification and recognition, and Weng et. al. 2008 [110] reported the performance advantages of the laminar architecture (paired layers) over a uniform neural area. Franz & Triesch 2007 [31] studied the development of disparity tuning in toy objects data using an artificial neural network based on back-propagation and reinforcement learning. They reported a 90% correct recognition rate for 11 classes of disparity. In Solgi & Weng 2008 [98], a multilayer in-place learning network was used to detect binocular disparities that were discretized into classes of 4 pixels intervals from image rows of 20 pixels wide. This classification scheme does not fit well for higher accuracy needs, as a misclassification between disparity class -1 and class 0 is very different from that between a class -1 and class 4. The work presented here also investigates the more challenging problem of regression with sub-pixel precision, in contrast

with the prior scheme of classification in Solgi & Weng 2008 [98].

For the first time, we present a spatio-temporal regression model of the laminar architecture of the cortex for stereo that is able to perform competitively on the difficult task of stereo disparity detection in natural images with sub-pixel precision. The model of the inter-cortical connections we present here was informed by the work of Felleman & Van Essen [26] and that for the intra-cortical connections was informed by the work of Callaway [11] and Wiser & Callaway [117] as well as others.

Luciw & Weng 2008 [64] presented a model for top-down context signals in spatio-temporal object recognition problems. Similar to their work, in this paper the emergent recursive top-down context is provided from the response pattern of the motor cortex at the previous time to the feature detection cortex at the current time. Biologically plausible networks (using Hebbian learning instead of error back-propagation) that use both bottom-up and top-down inputs with engineering-grade performance evaluation have not been studied until recently [110, 98, 61].

It has been known that orientation preference usually changes smoothly along the cortex [10]. Chen et. al. [14] has recently discovered that the same pattern applies to the disparity selectivity maps in monkey $V2$. Our model shows that defining disparity detection as a regression problem (as opposed to classification) helps to form similar patterns in topographic maps; disparity selectivity of neurons changes smoothly along the neural plane.

In summary, the work here is novel in the following aspects: (1) The first laminar model (paired layers in each area) for stereo. (2) The first utilization of temporal signals in a laminar model for stereo (3) The first sub-pixel precision among the network learning models for stereo. Applying the novelties mentioned in (1) and (2) showed surprisingly drastic accuracy differences in performance. (4) The first study of smoothly-changing disparity sensitivity

maps (5) Theoretical analysis that supports and provides insights into such performance differences.

One may question the suitability of supervised learning for Autonomous Mental Development (AMD). However, the AMD literature goes beyond the traditional classification of machine learning types, and divides all the machine learning methods into 8 categories [113]. The learning method used in this work falls in Type 2 of the classification proposed in [113], and therefore, fits the autonomous mental development paradigm.

The extensive research literature in psychology supports the notion of developing visual capabilities via touch and interaction with the environment, also known as associative learning (e.g. [101]). Here is a specific example of supervised learning via touch in disparity detection learning: Assume that a monkey sees a banana and touches it at the same time. The distance that the monkey has extended its hand to touch the banana serves as supervisory signal to guide learning the disparity of the banana in its visual field. In general, any previously categorized (known) stimulus (e.g. length of monkey's hand) can supervise any unknown stimulus (e.g. disparity of the banana), given they are presented at the same time (associative learning).

In a nutshell, the proposed stereoscopic network develops, in the feature detection cortex, a set of binocular features (templates for inner-product matching. See Fig. 4.5). These features are both profile-specific and disparity-specific. The best match from a binocular input means a match for both profile and disparity. The same mechanisms were used to develop the motor cortex neurons; as long as the top-matched neurons in the feature detection cortex and the corresponding motor cortex neurons fire together, they are connected (associated).

In the remainder of this chapter, we first introduce the architecture of the networks in Section II. Section III provides analysis. Next, the implementation and results are presented

in Section IV. Finally, we provide some predictions and concluding remarks in Sections V and VI.

3.2 Network Architecture and Operation

The networks applied in this paper are extensions of the previous models of Multilayer In-place Learning Network (MILN) [110]. To comply with the principles of Autonomous Mental Development (AMD) [111], these networks autonomously develop features of the presented input, and no hand-designed feature detection is needed.

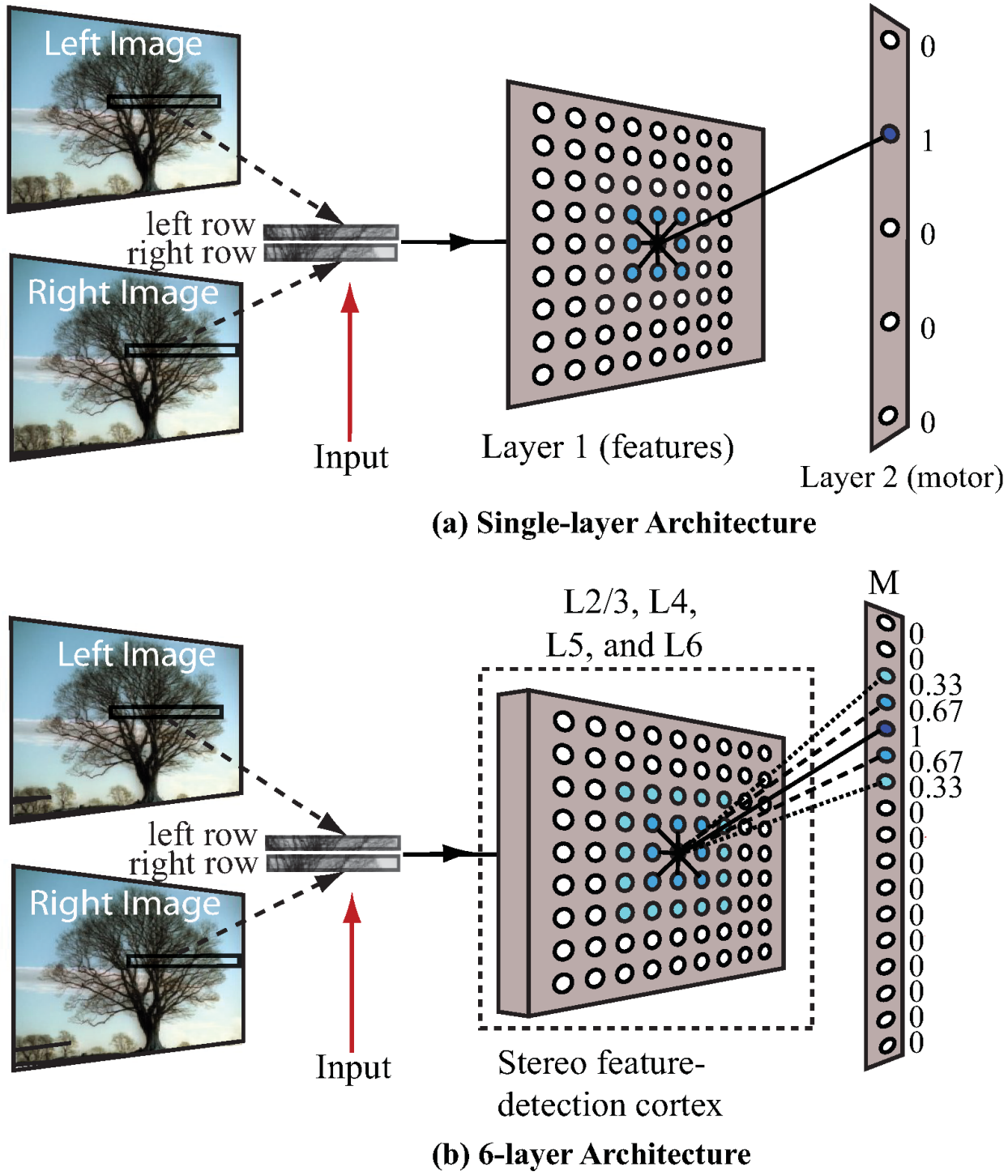


Figure 3.1: **(a)**. The binocular network single-layer architecture for classification. **(b)**. The binocular network 6-layer architecture for regression.

To investigate the effects of supervisory top-down projections, temporal context, and

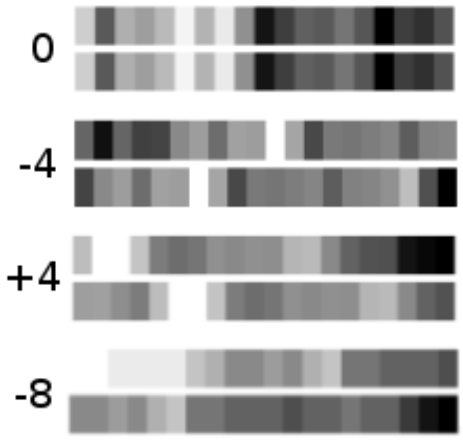


Figure 3.2: Examples of input, which consists of two rows of 20 pixels each. The top row is from the left view and the bottom row is from the right view. The numbers on the left side of the bars exhibit the amount of shift/disparity.

laminar architecture, we study two types of networks: 1) Single-layer architecture for classification and 2) 6-layer architecture for regression. An overall sketch of the networks is illustrated in Fig. 3.1. In this particular study, we deal with networks consisting of a sensory array (marked as *Input* in Fig. 3.1), a stereo feature-detection cortex, which may be a single layer of neurons or have a 6-layer architecture inspired by the laminar architecture of human cortex, and a motor cortex that functions as a regressor or a classifier.

3.2.1 Single-layer Architecture

In the single-layer architecture, the feature-detection cortex simply consists of a grid of neurons that is globally connected to both the motor cortex and input. It performs the following 5 steps to develop binocular receptive fields:

1. Fetching input in Layer 1 and imposing supervision signals (if any) in motor cortex – When the network is being trained, $\mathbf{z}^{(M)}$ is imposed originating from outside (e.g.,

by a teacher). In a classification problem, there are c motor cortex neurons and c possible disparity classes. The true class being viewed is known by the teacher, who communicates this to the system. Through an internal process, the firing rate of the neuron corresponding to the true class is set to one, and all others set to zero.

2. Pre-response – Neuron n_i on the feature-detection cortex computes its pre-competitive response $\hat{z}_i^{(L1)}$ – called *pre-response*, linearly from the bottom-up part and top-down part

$$\begin{aligned} \hat{z}_i^{(L1)}(t) &= (1-\alpha) \cdot \frac{\mathbf{b}^{(L1)}(t) \cdot \mathbf{w}_{b,i}^{(L1)}(t)}{\|\mathbf{b}^{(L1)}(t)\| \, \|\mathbf{w}_{b,i}^{(L1)}(t)\|} \\ &+ \alpha \cdot \frac{\mathbf{z}^{(M)}(t) \cdot \mathbf{w}_{e,i}^{(L1)}(t)}{\|\mathbf{z}^{(M)}(t)\| \|\mathbf{w}_{e,i}^{(L1)}(t)\|} \end{aligned} \tag{3.1}$$

where t denotes time, $\mathbf{w}_{b,i}^{(L1)}(t)$ and $\mathbf{w}_{e,i}^{(L1)}(t)$ are this neuron's bottom-up and top-down weight vectors, respectively, $\mathbf{b}^{(L1)}$ is the bottom-up input vector to Layer 1, and $\mathbf{z}^{(M)}(t)$ is the firing rates of motor cortex neurons (supervised during training, and not active during testing). The relative top-down coefficient α is discussed in detail later. We do not utilize linear or non-linear function g, such as a sigmoid, on firing rate in this paper.

3. Competition via Lateral Inhibition – A neuron's pre-response is used for intra-level competition. k neurons with the highest pre-response win, and the others are inhibited. If $r_i = \text{rank}(\hat{z}_i^{(L1)}(t))$ is the ranking of the pre-response of the i'th neuron (with the highest active neuron ranked as 0), we have $z_i^{(L1)}(t) = s(r_i)\hat{z}_i^{(L1)}(t)$, where

$$\text{s}(r_i) = \begin{cases} \frac{k-r_i}{k} & \text{if } 0 \le r_i < k \\ 0 & \text{if } r_i \ge k \end{cases} \tag{3.2}$$

4. Smoothing via Lateral Excitation – Lateral excitation means that when a neuron fires, the nearby neurons in its local area are more likely to fire. This leads to a smoother representational map. The topographic map can be realized by not only considering a nonzero-responding neuron i as a winner, but also its 3×3 neighbors, which are the neurons with the shortest distances from i (less than two).

5. Hebbian Updating with LCA – After inhibition, the top-winner neuron and its 3×3 neighbors are allowed to fire and update their synapses. We use an updating technique called lobe component analysis [112]. See Appendix A for details.

The motor cortex neurons develop using the same five steps as the above, but there is not top-down input, so Eq. 3.1 does not have a top-down part. The response $\mathbf{z}^{(M)}$ is computed in the same way otherwise, with its own parameter k controlling the number of non-inhibited neurons.

3.2.2 6-layer Cortical Architecture

The architecture of the feature-detection cortex of the 6-layer architecture is sketched in Fig. 3.3. We use no hand-designed feature detector (e.g. Laplacian of Gaussian, Gabor filters, etc.), as it would be against the paradigm of AMD [111]. The five layers in the stereo feature detection cortex are matched in functional-assistant pairs (referred as feedforward-feedback pairs in [12]). and $L3$ are counted as one layer ($L2/3$) for now. Later in the paper, we will hypothesize that they are two functionally-distinct layers. are matched in functional-assistant pairs (referred as feedforward-feedback pairs in [12]). $L6$ assists $L4$ (called assistant layer for $L4$) and $L5$ assists $L2$ and $L3$.

Layer $L4$ is globally connected to the input, meaning that each neuron in $L4$ has a connection to every pixel in the input image. All the two-way connections between $L4$ and

$L6$, and between $L2$, $L3$ and $L5$, and also all the one-way connections from $L4$ to $L3$ are *one-to-one* and consant. In other words, each neuron in one layer is connected to only one neuron in the other layer at the same position in neural plane coordinates, and the weight of the connections is fixed to 1. Finally, neurons in the motor cortex are globally and bidirectionally connected to those in $L2$. There are no connections from $L2$ or $L3$ to $L4$.

The stereo feature-detection cortex takes a pair of stereo rows from the sensory input array. Then it runs the following developmental algorithm.

1. Imposing supervision signals (if any) in motor cortex – During developmental training phase, an external teacher mechanism sets the activation levels of the motor cortex according to the input. If n_i is the neuron representative for the disparity of the currently presented input, then the activation level of n_i and its neighbors are set according to a triangular kernel centered on n_i. The activation level of all the other neurons is set to zero:

$$z_j^{(M)}(t) = \begin{cases} 1 - \frac{d(i,j)}{\kappa} & \text{if } d(i,j) < \kappa \\ 0 & \text{if } d(i,j) \geq \kappa \end{cases} \tag{3.3}$$

where M denotes Motor Cortex, $d(i,j)$ is the distance between neuron n_i and neuron n_j in the neural plane, and κ is the radius of the triangular kernel.

Then the activation level of motor neurons from the previous time step, $z_j^{(M)}(t-1)$, is projected onto $L2$ neurons via top-down connections.

$$\mathbf{e}^{(L2)}(t) = \mathbf{z}^{(M)}(t-1) \tag{3.4}$$

2. Pre-response in $L4$ and $L2$ – Neurons in $L4(L2)$ compute their *pre-response*

(response prior to competition) solely based on their bottom-up(top-down) input. They use the same equation as in Eq. 3.1, except $L4$ only has bottom-up and $L2$ only has top-down.

$$\hat{z}_i^{(L4)}(t) = \frac{\mathbf{b}^{(L4)}(t) \cdot \mathbf{w}_{b,i}^{(L4)}(t)}{\|\mathbf{b}^{(L4)}(t)\| \|\mathbf{w}_{b,i}^{(L4)}(t)\|} \tag{3.5}$$

and

$$\hat{z}_i^{(L2)}(t) = \frac{\mathbf{e}^{(L2)}(t) \cdot \mathbf{w}_{e,i}^{(L2)}(t)}{\|\mathbf{e}^{(L2)}(t)\| \|\mathbf{w}_{e,i}^{(L2)}(t)\|} \tag{3.6}$$

3. $L6$ and $L5$ provide modulatory signals to $L4$, $L2$ and $L3$ – $L6$ and $L5$ receive the firing pattern of $L4$, $L2$ and $L3$, respectively, via their one-to-one connections. Then they send modulatory signals back to their paired layers, which will enable the functional layers to do long-range lateral inhibition in the next step.

Since the LCA algorithm already incorporates the regulatory mechanisms (i.e. lateral inhibition and excitation) in the functional layers ($L2$, $L3$ and $L4$), assistant layers ($L5$ and $L6$) do not have "actual" neurons in our implementation. They are modeled only to explain the important role of $L5$ and $L6$ in the cortical architecture: providing signals to regulate lateral interactions in $L2$, $L3$ and $L4$ [12].

4. Response in $L4$ and $L2$ – Provided by feedback signals from $L6$, the neurons in $L4$ internally compete via lateral inhibition. The mechanism for inhibition is the same as described in Step 3 of single-layer architecture. The same mechanism concurrently happens in $L2$ assisted by $L5$

5. Response in $L3$ – Each neuron, n_i in $L3$ receives its bottom-up input from one-to-one connection with the corresponding neuron in $L4$ (i.e. $b_i^{(L3)}(t) = z_i^{(L4)}(t)$) and its top-down in-

put from one-to-one connection with the corresponding neuron in $L2$ (i.e $e_i^{(L3)}(t) = z_i^{(L2)}(t)$). Then it applies the following formula to merge bottom-up and top-down information and compute its response.

$$z_i^{(L3)}(t) = (1 - \alpha) \cdot b_i^{(L3)}(t) + \alpha \cdot e_i^{(L3)}(t) \tag{3.7}$$

where α is the relative top-down coefficient. We will discuss the effect of this parameter in detail in Section 3.3.2.1.

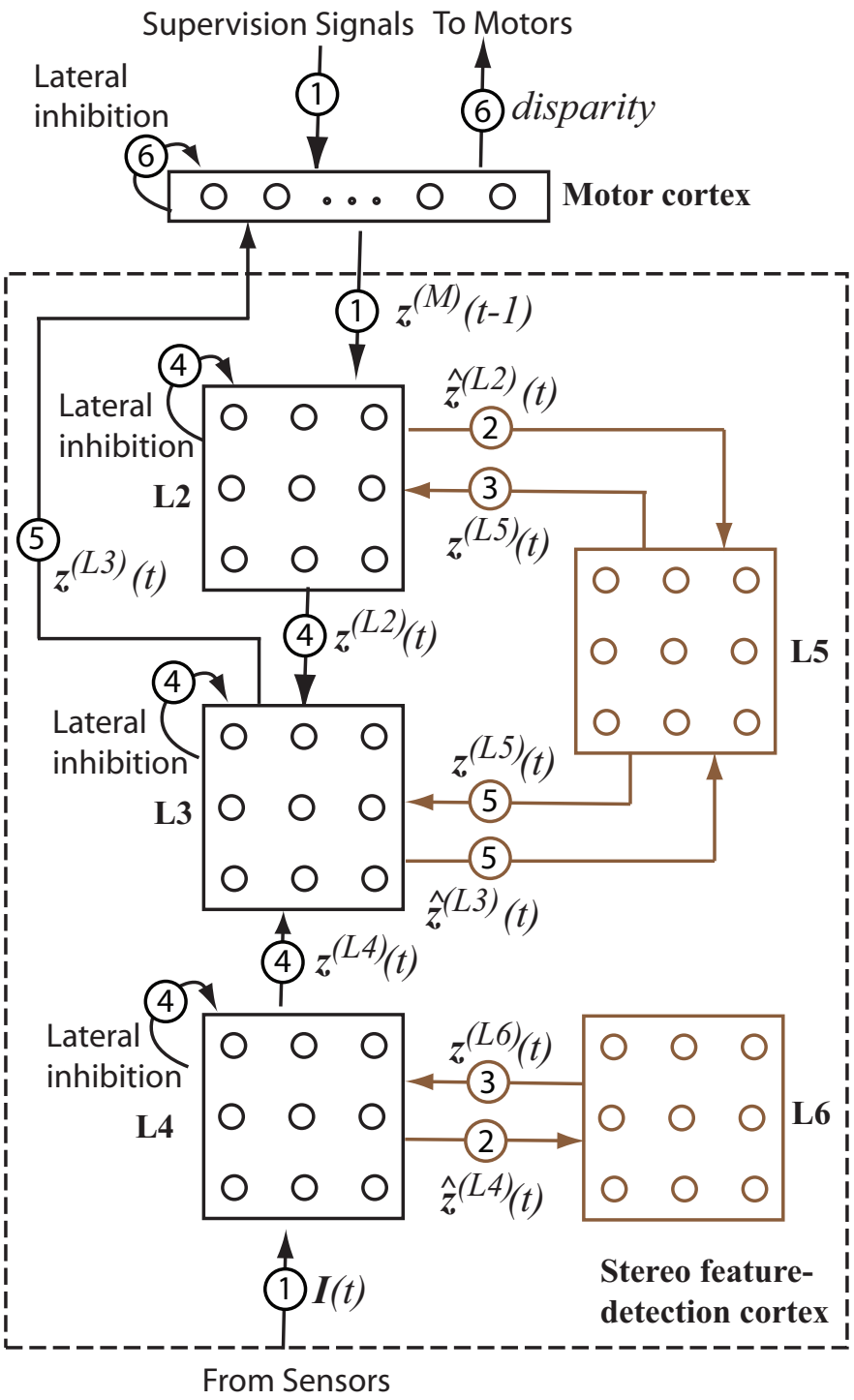


Figure 3.3: Architecture diagram of the 6-layer laminar cortex studied in this paper, which also introduces some notation. The numbers in circles are the steps of the algorithm described in Section 3.2. See the text for notations. Parts depicted in brown (gray in black and white copies) are not implemented in our computer simulation.

6a. Response of motor Neurons in Testing – The activation level of the motor

neurons is not imposed during testing, rather it is computed utilizing the output of feature-detection cortex, and used as context information in the next time step. The neurons take their input from $L3$ (i.e. $\mathbf{b}_i^{(M)}(t) = \mathbf{z}^{(L3)}(t)$). Then, they compute their response using the same equation as in Eq. 3.5, and laterally compete. The response of the winner neurons is scaled using the same algorithm as in Eq. 3.2 (with a different k for the motor layer), and the response of the rest of the neurons will be suppressed to zero. The output of the motor layer is the response weighted average of the disparity of the winner neurons:

$$disparity = \frac{\sum_{n_i \text{ is winner}} d_i \times z_i^{(M)}(t)}{\sum_{n_i \text{ is winner}} z_i^{(M)}(t)} \tag{3.8}$$

where d_i is the disparity level that the winner neuron n_i is representative for.

6b. Hebbian Updating with LCA in Training – The top winner neurons in $L4$ and motor cortex and also their neighbors in neural plane (excited by 3×3 short-range lateral excitatory connections) update their bottom-up connection weights. Lobe component analysis (LCA) [112] is used as the updating rule. See Appendix A for details.

Afterwards, the motor cortex bottom-up weights are directly copied to $L2$ top-down weights. This is another one of the deliberate simplifications we have applied to make this model faster and less computationally expensive at this stage. The LCA theory as well as our experimental results show that neurons can successfully develop top-down and bottom-up weights independently. However, it takes more computation and training time. Our future work models the top-down and bottom-up weights updating independently.

3.3 Experiments and Results

The results of the experiments carried out using the models discussed in the previous sections are presented here. The binocular disparity detection problem was formulated once as a classification problem, and then as a regression problem.

3.3.1 Classification

The input to the network is a pair of left and right rows, each 20 pixels wide. The image-rows were extracted randomly from 13 natural images (available from http://www.cis.hut.fi/projects/ica/imageic The right-view row position is shifted by -8, -4, 0, 4, 8 pixels, respectively, from the left-view row, resulting in 5 disparity classes. Fig. 3.2 shows some sample inputs. There were some image regions where texture is weak, which may cause difficulties in disparity classification, but we did not exclude them. During training the network was randomly fed with samples from different classes of disparity. The developed filters in Layer 2 are shown in Fig. 3.4.

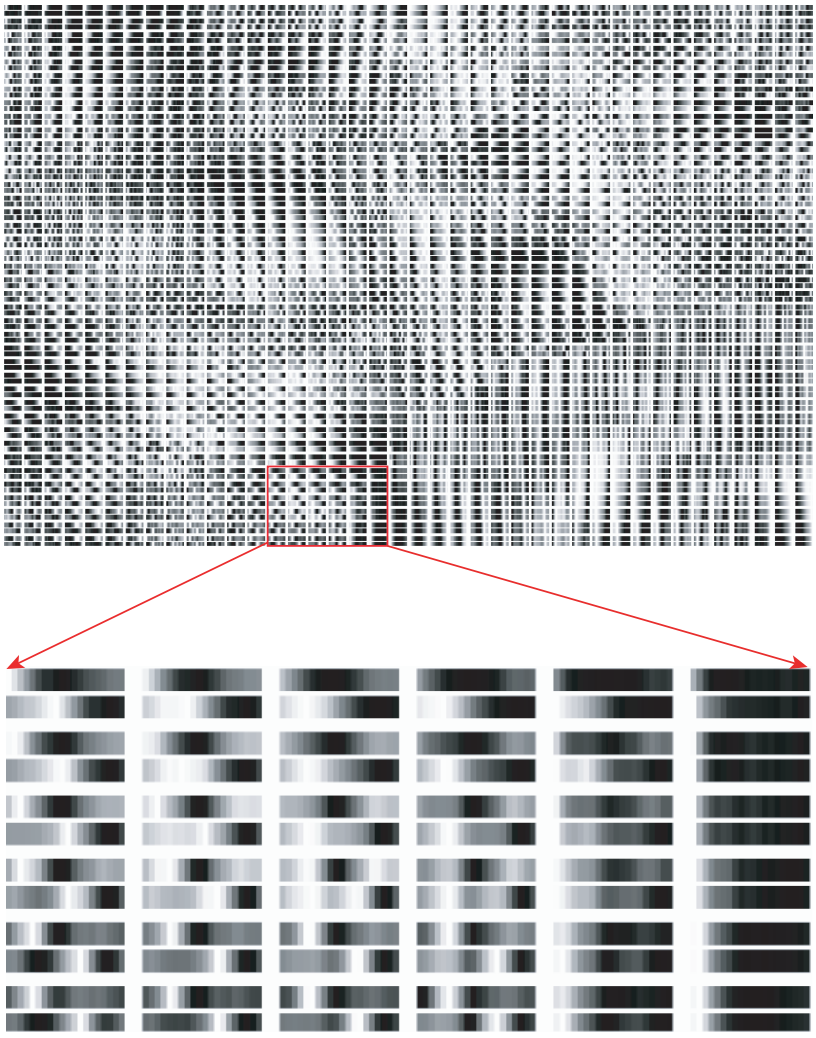


Figure 3.4: Bottom-up weights of 40×40 neurons in feature-detection cortex using top-down connections. Connections of each neurons are depicted in 2 rows of each 20 pixels wide. The top row shows the weight of connections to the left image, and the bottom row shows the weight of connections to the right image.

3.3.1.1 The Effect of Top-Down Projection

As we see in Fig. 3.5, adding top-down projection signals improves the classification rate significantly. It can be seen that when $k = 50$ (k is the number of neurons allowed to fire in each layer) for the top-k updating rule, the correct classification rate is higher early on. This is expected as no feature detector can match the input vector perfectly. With more neurons allowed to fire, each input is projected onto more feature detectors. The population coding

gives richer information about the input, and thus, also the disparity. When more training samples are learned, the top-1 method catches up with the top-50 method.

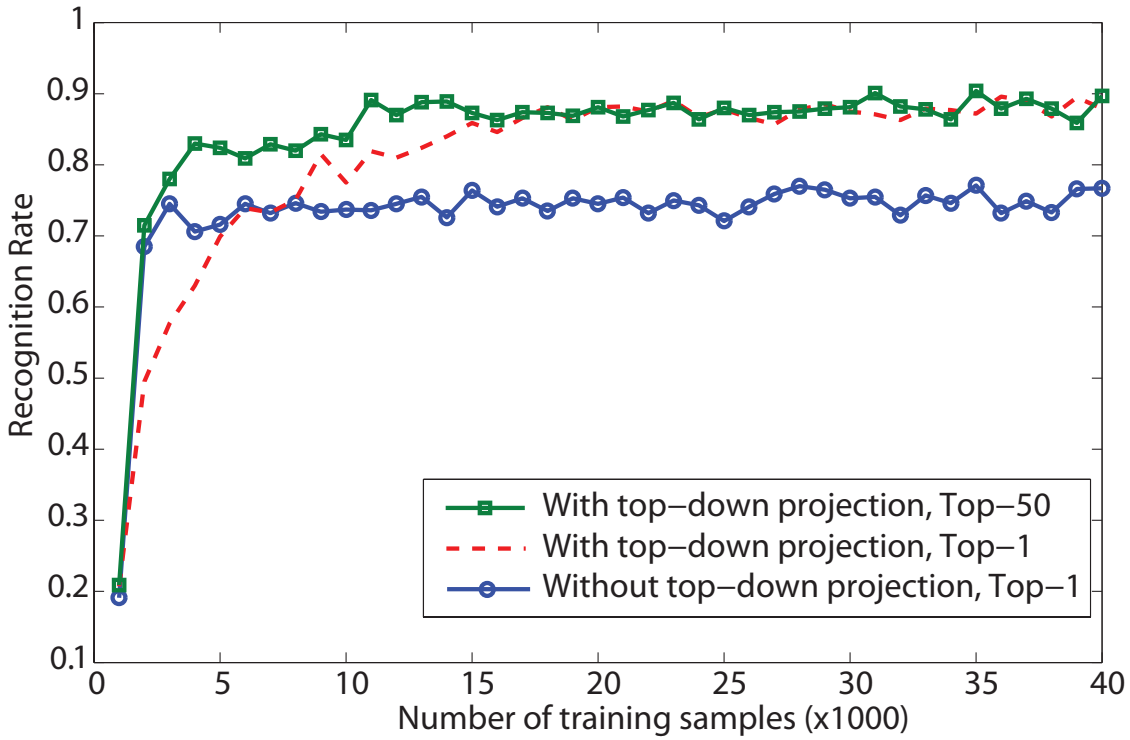


Figure 3.5: The recognition rate versus the number of training samples. The performance of the network was tested with 1000 testing inputs after each block of 1000 training samples.

3.3.1.2 Topographic Class Maps

As we see in Fig. 3.6, supervisory information conveyed by top-down connections resulted in topographically class-partitioned feature detectors in the neuronal space, similar to the network trained for object recognition [66]. Since the input to a neuron in feature-detection layer has two parts, the iconic input $\mathbf{x}_b$ and the abstract (e.g. class) input $\mathbf{x}_t$, the resulting internal representation in feature-detection layer is *iconic-abstract*. It is grossly organized by class regions, but within region it is organized by iconic input information. However, these two types of information are not isolated - they are considered jointly by neuronal self-organization.

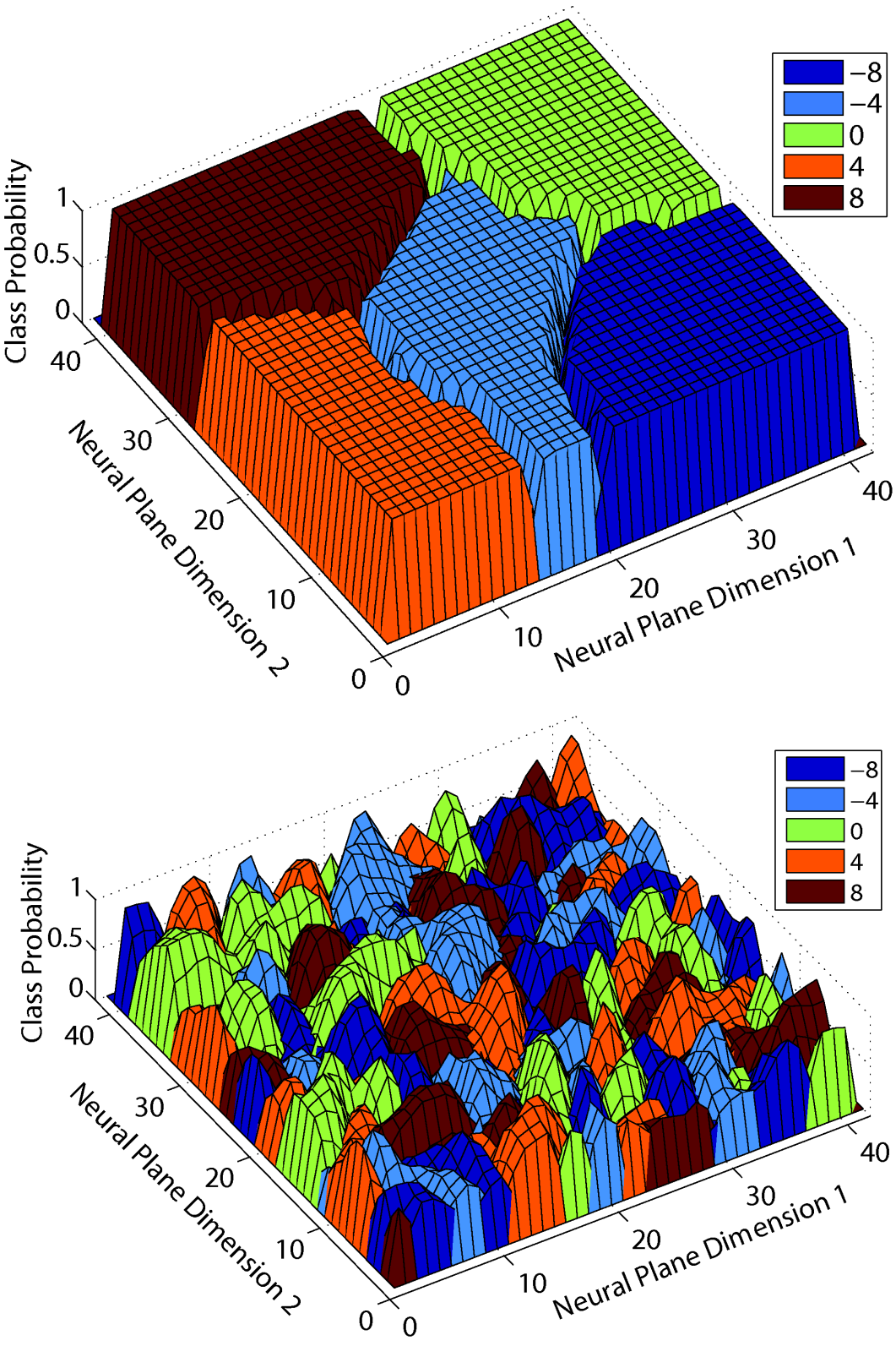


Figure 3.6: The class probability of the 40×40 neurons of the feature-detection cortex. (a) Top-down connections are active ($\alpha = 0.5$) during development. (b) Top-down connections are not active ($\alpha = 0$) during development.

To measure the purity of the neurons responding to different classes of disparity, we computed the entropy of the neurons as follows:

$$H = \sum_{i=1}^{N} -p(n, C_i) \log(p(n, C_i)) \tag{3.9}$$

where N is the number of classes and $p(n, C_i)$ is defined as:

$$p(n, C_i) = \frac{f(n, C_i)}{\sum_{j=0}^{m} f(n, C_j)} \tag{3.10}$$

where n is the neuron, C_i represents class i, and $f(n, C_i)$ is the frequency for the neuron n to respond to the class C_i.

Fig. 3.7 shows that the topographic representation enabled by the top-down projections generalizes better and increases the neurons' purity significantly during training and testing.

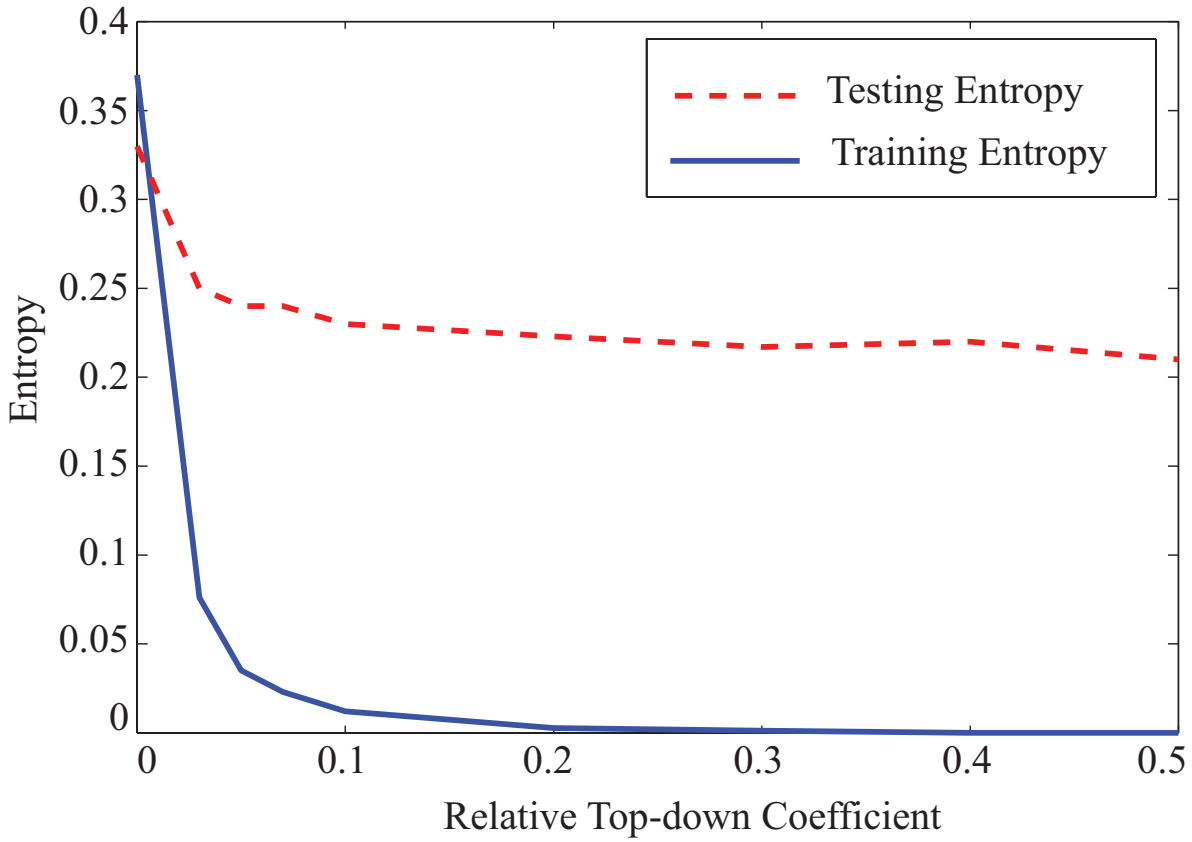


Figure 3.7: The effect of top-down projection on the purity of the neurons and the performance of the network. Increasing α in Eq. 3.1 results in purer neurons and better performance.

3.3.2 Regression

From a set of natural images (available from http://www.cis.hut.fi/projects/ica/imageica/), 7 images were randomly selected, 5 of them were randomly chosen for training and 2 for testing. A pair of rows, each 20 pixels wide, were extracted from slightly different positions in the images. The right-view row was shifted by $-8, -7, -6, \ldots, 0, \ldots, +6, +7, +8$ pixels from the left-view row, resulting in 17 disparity degrees. In each training epoch, for each degree of disparity, 50 spatially continuous samples were taken from each of the 5 training images. Therefore, there was $5 \times 50 \times 17 = 4250$ training samples in each epoch. For testing, 100 spatially continuous samples were taken from each of the 2 testing images (disjoint test),

resulting in $2 \times 100 \times 17 = 3400$ testing samples in each epoch.

We trained networks with 40×40 neurons in each of $L2$, $L3$ and $L4$ layers of the stereo feature-detection cortex. correspondence between input and $L1$ neurons). The k parameter (the number of neurons allowed to fire in each layer) was set to 100 for the stereo feature-detection cortex, and 5 for the motor cortex. We set $\kappa = 5$ in Eq. 3.3 and $\alpha = 0.4$ in Eq. 3.7 for all of the experiments, unless otherwise is stated.

3.3.2.1 The Advantage of Spatio-temporal 6-layer Architecture

Fig. 3.8 shows that applying top-down context signals in single-layer architecture (traditional MILN networks [110]), increases the error rate up to over 5 pixels (we intentionally set the relative top-down coefficient, α, as low as 0.15 in this case, otherwise the error rate would be around chance level). This observation is due the absolute dominance of misleading top-down context signals provided complex input (natural images in this study). On the other hand, context signals reduce the error rate of the network to a sub-pixel level in 6-layer architecture networks. This result shows the important role of assistant layers (i.e. $L5$ and $L6$) in the laminar cortex to modulate the top-down and bottom-up energies received at the cortex before mixing them.

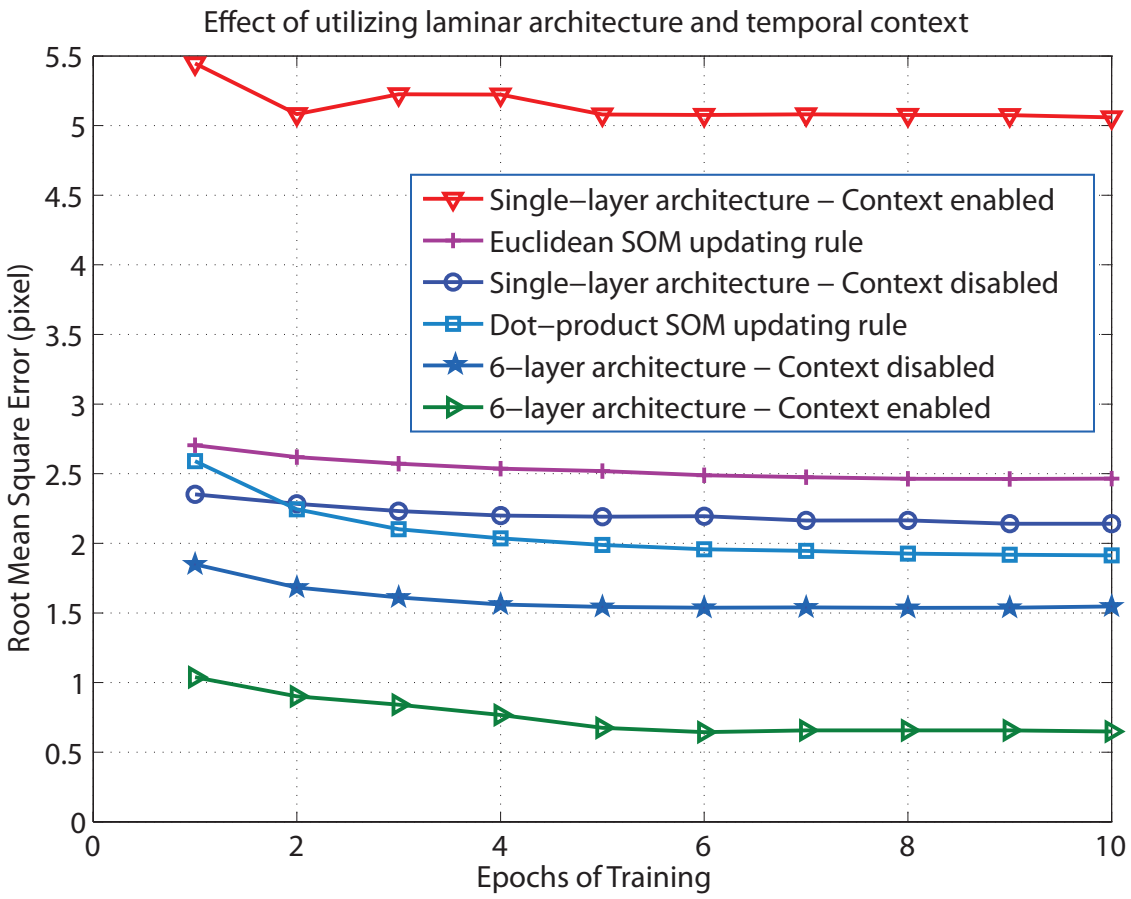


Figure 3.8: How temporal context signals and 6-layer architecture improve the performance.

For comparison, we implemented two versions of Self-Organizing Maps updating rules, Euclidean SOM and dot-product SOM [53]. With the same amount of resources, the 6-layer architecture outperformed both versions of SOM by as much as at least 3 times lower error rate.

In another experiment, we studied the effect of relative top-down coefficient α. Different networks were trained with more than 40 thousand random training samples (as opposed to training with epochs). Fig. 3.9 shows the effect of context parameter, α, in disjoint testing. It can be seen that the root mean square error of disparity detection reaches to around 0.7 pixels when $\alpha = 0.4$. We believe that in natural visual systems, the ratio of contribution of top-down temporal signals (α in our model) is tuned by evolution.

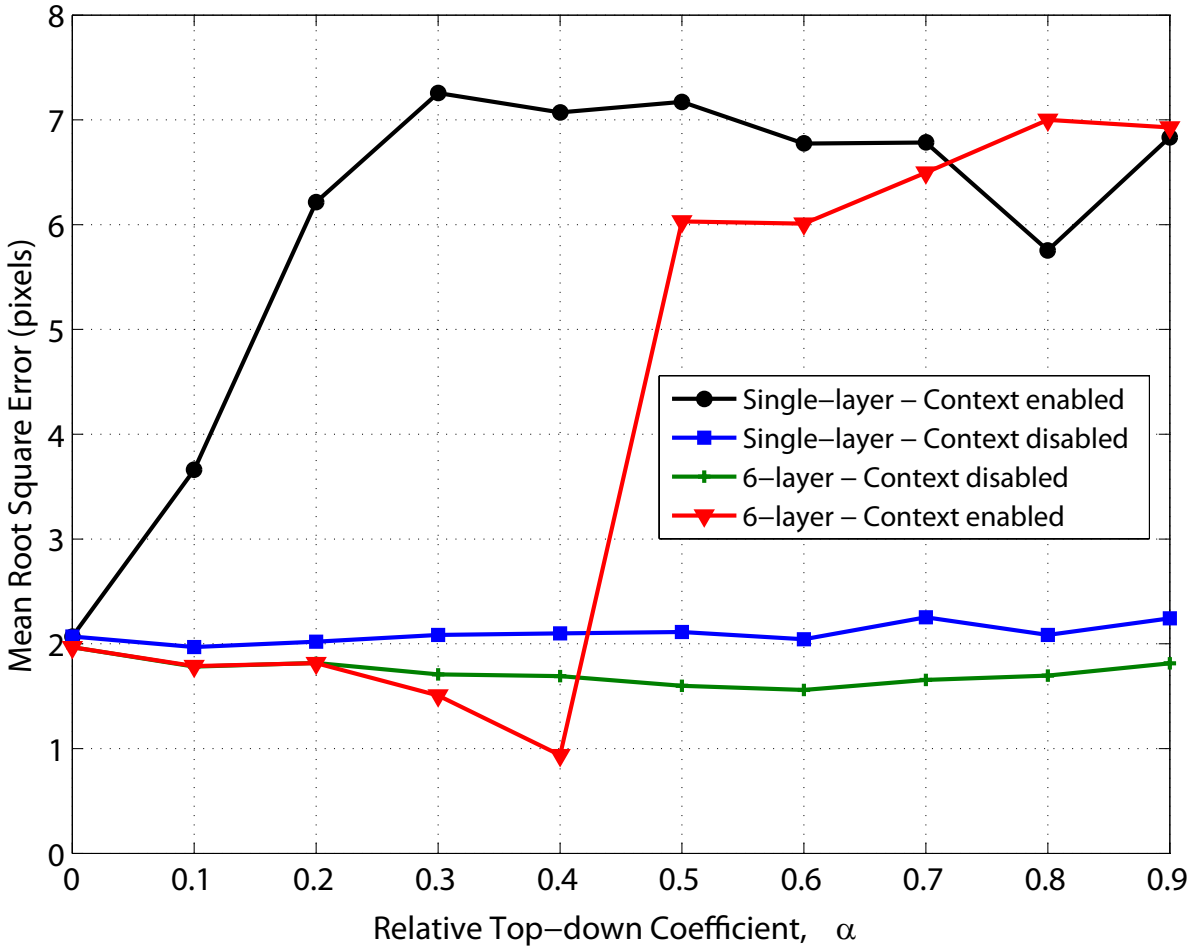


Figure 3.9: The effect of relative top-down coefficient, α, on performance in disjoint recognition test on randomly selected training data.

3.3.2.2 Smoothly Changing Receptive Fields

In two separate experiments, we studied the topographic maps formed in $L3$.

3.3.2.3 Experiment A – $\kappa = 5$

As depicted in Fig. 3.10a, the disparity-probability vectors for neurons tuned to close-by disparities are similar; neurons tuned to close-by disparities are more likely to fire together. Equivalently, a neuron in the stereo feature-detection cortex is not tuned to only one exact disparity, but to a disparity range with a Gaussian-like probability for different

disparities (e.g. neuron n_i could fire for disparities $+1, +2, +3, +4, +5$ with probabilities $0.1, 0.3, 0.7, 0.3, 0.1$, respectively). This fuzziness in neuron's disparity sensitivity is caused by smoothly changing motor initiated top-down signals ($\kappa > 1$ in Eq. 3.3) during training. Fig. 3.10b shows this effect on topographic maps; having $\kappa = 5$ causes the regions sensitive to close-by disparities quite often reside next to each other and change gradually in neural plane (in many areas in Fig. 3.10b the colors change smoothly from dark blue to red).

3.3.2.4 Experiment B – $\kappa = 1$

However, if we define disparity detection as a classification problem, and set $\kappa = 1$ in Eq. 3.3 (only one neuron active in motor layer), then there is no smoothness in the change of the disparity sensitivity of neurons in the neural plane.

These observations are consistent with recent physiological discoveries about the smooth change of stimuli preference in topographic maps in the brain [15] and disparity maps in particular [14, 91].

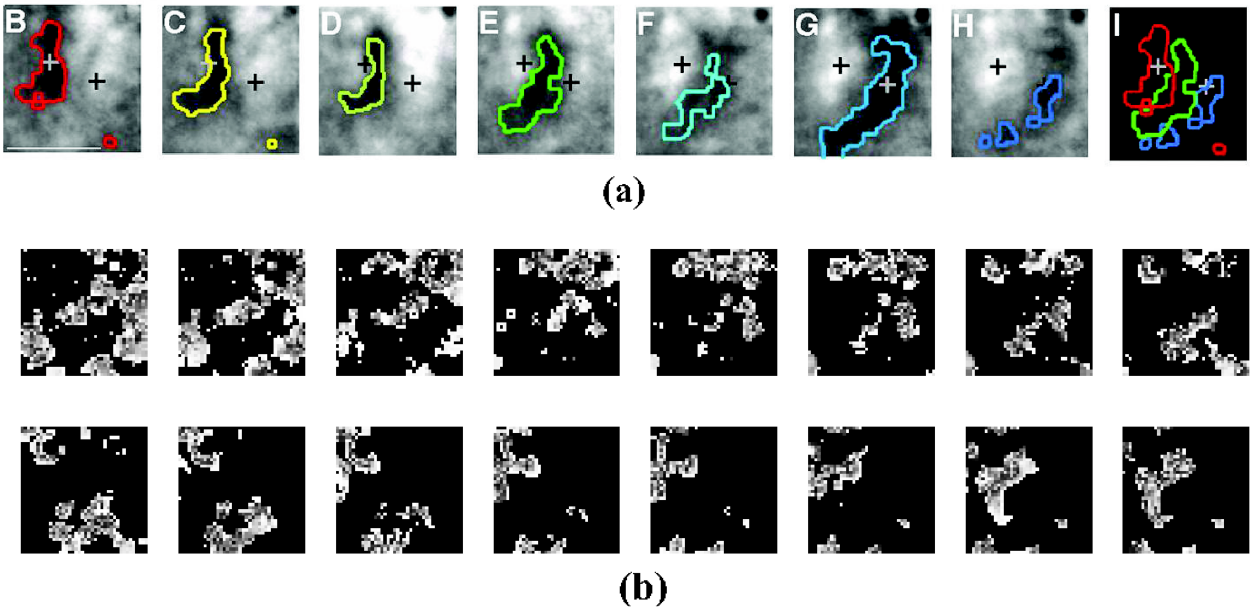


Figure 3.10: **(a)** Map of neurons in V2 of macaque monkeys evoked by stimuli with 7 different disparities. The position of the two crosses are constant through all the images marked as (B)-(H). Adapted from Chen et. al. 2008 [14] **(b)** Disparity-probability vectors of $L3$ neurons for different disparities when $\kappa = 5$. Disparity-probability vector for each disparity is a $40 \times 40 = 1600$ dimensional vector containing the probability of neurons to fire for that particular disparity (black(white): minimum(maximum) probability).

3.4 Discussion

The lack of computational experiments on real-world data in previous works has led to the oversight of the role of sparse coding in neural representation in the models of laminar cortex. Sparse coding of the input is computationally advantageous both for bottom-up and top-down input, specially when the input is complex. Therefore, we hypothesize that the cortical circuits probably have a mechanism to sparsely represent top-down and bottom-up input. Our model suggests that the brain computes a sparse representation of bottom-up and top-down input independently, before it integrates them to decide the output of the cortical region. Thus, we predict that:

Prediction 1: What is known as Layer 2/3 in cortical laminar architecture [1] has two functional roles:

1. Rank and scale the top-down energy received at the cortex (modulated by signals from $L5$) in $L2$

2. Integrate the modulated bottom-up energy received from $L4$ to the modulated top-down energy received from higher cortical areas to determine the output signals of the cortex in $L3$

Neuroscientists have known for a long time that there are sublayers in the laminar cortex [48]. However, the functionality of these sublayers has not been modeled before. This is a step towards understanding the sublayer architecture of the laminar cortex. Our prediction breaks down the functionality of Layer 2/3 ($L2/3$) to two separate tasks. This is different from the previous models (e.g. [11]), as they consider $L2/3$ as one functional layer.

Fig. 3.11 illustrates the result of an experiment in which we compared two models of $L2/3$. In the traditional model of $L2/3$, it is modeled as one functional layer that integrates the sparse coded signals received from $L4$ with the top-down energy. While in our novel model used in this paper, $L2/3$ functions as 2 functional layers, namely $L2$ and $L3$ (see Prediction 1).

[1] Marked as $Level2$, layers 2 through $4B$ in [11] Figure 2.

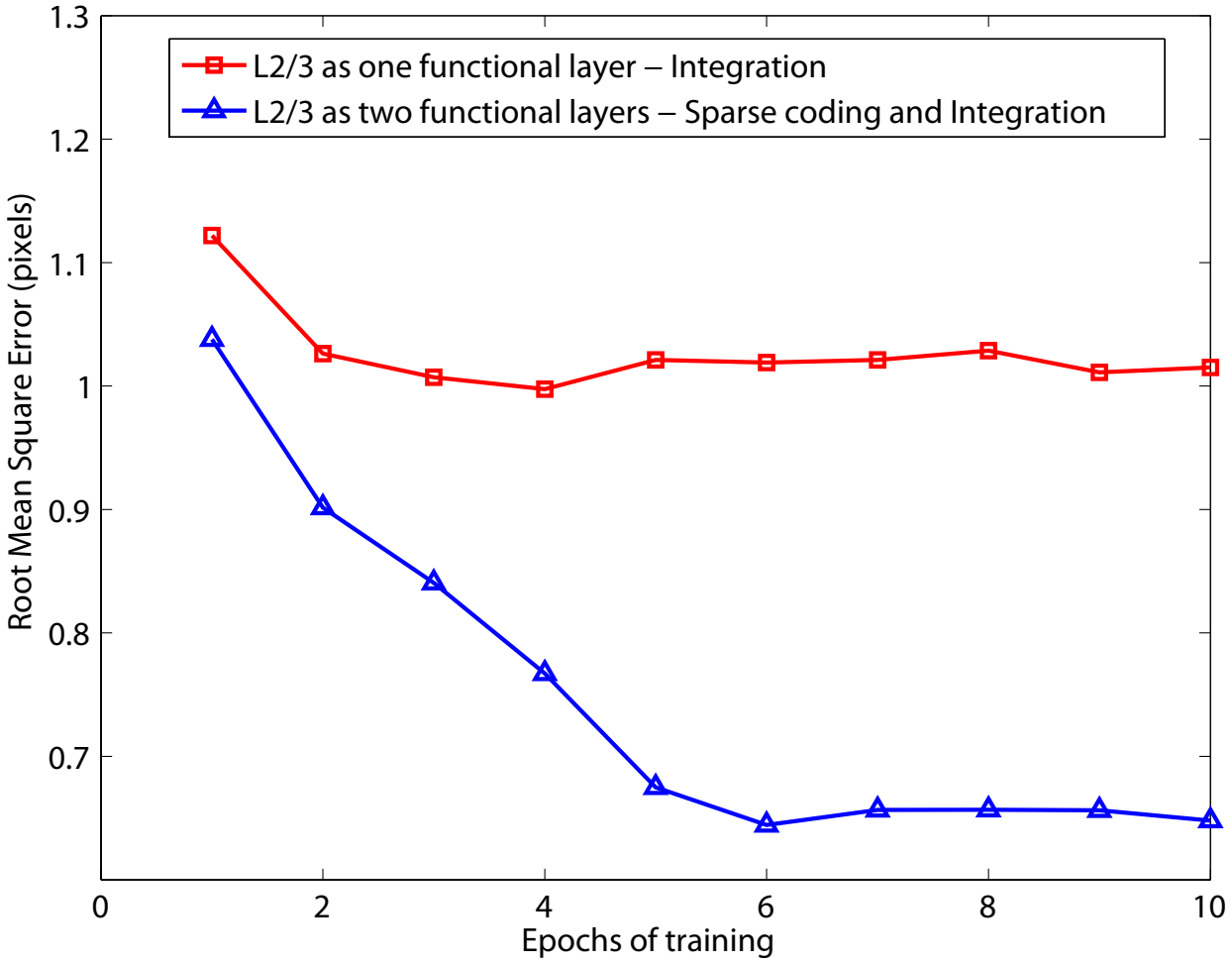


Figure 3.11: Comparison of our novel model of $L2/3$ where it performs both sparse coding and integration of top-down and bottom-up signals, with traditional models in which it only does integration.

3.5 Conclusions

Presented is the first spatio-temporal model of the 6-layer architecture of the cortex which incorporated temporal aspects of the stimuli in the form of top-down context signals. It outperformed simpler single-layer models of the cortex by a significant amount. Furthermore, defining the problem of binocular disparity detection as a regression problem by training a few nearby neurons to relate to the presented stimuli (as opposed to only one neuron in the case

of classification), resulted in biologically-observed smoothly changing disparity sensitivity along the neural layers.

Since the brain generates actions through numerical signals(spikes) that drive muscles and other internal body effectors (e.g. glands), regression (output signals) seems closer to what the brain does, compared to many classification models that have been published in the literature. The regression extension of the MILN [110] has potentially a wide scope of application, from autonomous robots to machines that can learn to talk. A major open challenge is the complexity of the motor actions to be learned and autonomously generated.

As presented here, an emergent-representation based binocular system has shown disparity detection abilities with sub-pixel accuracy. In contrast with engineering methods that used explicit matching between the left and right search windows, a remarkable computational advantage of our work is the potential for integrated use of a variety of image information for tasks that require disparity as well as other visual cues.

Our model suggests a computational reason as to why there is no top-down connection from $L2$ and $L3$ to $L4$ in laminar cortex; to prevent the top-down and bottom-up energies received at the cortex from mixing before they internally compete to sort out winners. Hence, we predict that the thick layer Layer 2/3 ($L2/3$) in laminar cortex carries out more functionality than what has been proposed in previous models - it provides sparse representation for top-down stimuli in $L2$, combines the top-down and bottom-up sparse representations in $L3$, and projects the output of the cortical region to higher cortices.

Utilization of more complex temporal aspects of the stimuli and using real-time stereo movies will be a part of our future work.

Chapter 4

Unsupervised Binocular Feature Learning

The behavior of a neuron in a neural network is greatly influenced by at least two characteristics of its synaptic connections; the weights and the domains. The weights are the strength of the synaptic connections, and by "domain" we refer to the part(s) of the sensory array (e.g., the retinal receptive field) from which the neuron receives input. The success of artificial neural networks (ANN) for computational problems stems from their ability to modify the connection weights between neurons, i.e., dynamic weights. However, the domain is static in typical ANNs. For example, in a computer vision application neurons get input from the entire input image (globally connected networks) or a fixed square or circular patch of the input image (locally connected networks). In the presented work, for the first time we model and study neurons with *dynamic domains* and weights. Dynamic domains are consistent with our knowledge of neurobiology and have desirable computational implications. In the case of binocular depth perception, we show that unsupervised learning of "domain disparity" together with "weight disparity" results in a fair coverage of disparity space. We present a novel algorithm, Dynamic Lobe Component Analysis (DSLCA) which models the learning process of neurons to develop domain and weight characteristics. This work is a step towards creating a neurally plausible, minimally supervised learning system for 3D shape

perception.

4.1 Introduction

Ever since the seminal work of Hubel and Wiesel on visual receptive fields [43], the properties of the receptive fields of neurons in primary visual cortex (V1) have been extensively measured (e.g. [21, 18]). These studies have shown that most early visual receptive fields are localized in retinal space and selective to temporal and spatial characteristics of the stimuli, such as orientation, chromatic contrast, direction of movement, and binocular disparity [20, 18, 60, 9].

Binocular disparity, the difference between left and right retinal images due to the slight difference in the viewing angle of the left and right eyes, is the strongest of the depth cues in short range vision. Depth information inferred from binocular disparity help us perceive intricate shapes of the 3D objects and perform dexterous manual tasks such as threading a needle, for example.

Much progress has been made in creating computational models of stereo vision. Below we divide them into three main categories:

1. **Explicit matching:** Methods in this category, mostly used in traditional computer vision and image processing, first detect discrete features and then explicitly match them across two views according to a matching measure, e.g., correlation coefficient. Well-known work in this category include [37], [22] and [126].

2. **Static, hand-designed filters:** Filters are designed to compute profile-sensitive values (e.g. Gabor filters [115], [90], and phase information [30], [107]) from images and

then utilize these continuous values for feature matching. Then, an algorithm or a network maps from the matched features to disparity output [38].

3. **Learned filters:** These models develop disparity-selective filters (i.e., neurons) from experience, without doing explicit matching, and map the responses to disparity outputs (e.g. [56], [58], [44], [31], [99], [42]).

4.1.1 Stereo Where-What Networks

The algorithms and the model presented here are the continuation of the Lobe Component Analysis (LCA) [112] and the Where-What Networks (WWN) [46]. The LCA learning algorithm is modeled after the laminar architecture of the cortex and has been proven to be dually (both spatially and temporally) optimal [114]. The Where-What Networks (WWN), have been shown promising for simultaneous attention and recognition, while handling variations in scale, location and type as well as inter-class variations [46, 63, 108]. Moreover they have shown sub-pixel accuracy in disparity detection in challenging natural images [99].

As a result of the structure and the learning algorithms of the WWNs, each neuron develops specificity (selectivity) to both location (due to local connectivity and top-down Where signals) and profile (due to lateral competition and top-down What signals) of a neighborhood in the input space manifold. In this work we add binocular receptive fields to the Where-What Networks. This results in locally-connected neurons which are selective to four properties of the input: type, location, scale and disparity. Such networks will be able to simultaneously perceive the location, type, distance and shape of the visual objects. Although the current work focuses on explaining the development process of the binocular receptive fields without supervision or top-down signals, we believe that top-down projections

are vastly present even in the earliest visual areas (e.g., V1) [27]. This work is a first step towards creating an integrated Stereo Where-What Networks, and top-down connections will be added in future works.

In this work we present a novel version of LCA, named Dynamic Synapse Lobe Component Analysis (DSLCA), which allows the autonomous retraction and growth of synapses during the neural learning process. Wang et. al (2011) [105] showed that utilizing *Synapse Maintenance* with LCA results in development of more efficient feature representations. Here, we build on their work by allowing the synapses not only to cut (retract), but also to grow on the borders of the receptive fields. This allows dynamic change in the shape of the receptive fields of the neurons, which in turn results in superior representations and encoding of domain disparity in the case of stereo vision.

Despite the progress made in building a comprehensive stereo vision system, both in computer vision and biological vision research communities, the available models and methods are limited and fail in the case of occlusion, weak textures, etc. In this work we make an attempt to present a learning algorithm which improves upon the existing methods. The contributions include, but are not limited to: **a)** unsupervised learning of disparity selective filters. **b)** arbitrary domain of left and right receptive fields. **c)** local, as opposed to global, receptive fields. **d)** capturing both domain and weight disparity. The significance of each of these novelties and the terminology used will be elaborated in the rest of the paper.

4.1.2 Domain versus Weight Disparities

It has been long known that most neurons in the striate cortex of monkeys and cats are selective to binocular disparity [81, 83]. However, how these neurons encode disparity is still largely unknown. In an influential study, Anzai, Ohzawa and Freeman (1999) [6] presented

neurophysiological data and argued that binocular disparity can be encoded in two different ways:

1. **Position encoding:** Left and right RFs of a binocular neuron have the same spatial profile while their left and right retinal positions are not in correspondence.

2. **Phase encoding:** Left and right RFs of a binocular neuron have different spatial profiles (phase) while their retinal positions match between the left and right retinae.

Anzai et al. [6] showed that most binocular neurons utilize both types of encoding, i.e., position and shift encoding, to represent the range of disparities needed for stereo vision. Since their naming stemmed from the view that early visual neurons have sinusoidal RF profiles (hence, defined by position and phase) we choose to use a different terminology; domain and weight encoding instead of position and shift encodings, respectively.

Here we try to formally define "weights" and "domains" for a binocular neuron.

4.1.3 Weights

In a network of neurons a neuron n_i is identified by its weight vector,

$$\mathbf{v}_i = (\mathbf{v}_i^l, \mathbf{v}_i^r)$$

where $\mathbf{v}_{ij} \in \mathbb{R}$ and $\mathbf{v}_i^l$ and $\mathbf{v}_i^r$ correspond to the synapses from the left and right eyes, respectively. In other words, weights are the profile of the linear filter that the neuron represents.

4.1.4 Domains

The entire input array to the neural network is $\mathbf{X} \in \mathbb{R}^d$, where d is the number of input dimensions, e.g., $d = 200 \times 200$ pixels in our experiments. We define a domain function f_i for each neuron, n_i, where:

$$f_i : \mathbf{X} \to \{0, 1\}^d$$

For each element of input array vector, $\mathbf{X}_k$, the binary function f_i determines whether or not the neuron n_i gets input from that specific element. Given this definition, the domain of a neuron can be a region of any shape—connected or disconnected—in in either of left and right input arrays in a binocular system. Fig. 4.1 shows an example of a domain function for a neuron. In Fig. 4.1, two binary images of the letter **A** are given as the left and right components of the input array $\mathbf{X}$. Each small square indicates a pixel, and the values of the domain function f are printed within the pixels. This means that the example neuron receives input only from the pixels marked as 1 in the left and right images.

As it is marked in the figure, the lower left corner of the letter **A** is at the image coordinates $l = (9, 2)$ in the left image and at $r = (8, 3)$ coordinate in the right image. Therefore the disparity is $l - r = (1, -1)$, which is $+1$ vertical disparity and -1 horizontal disparity.

4.1.5 Correspondence between weights and domains

Each element of a neuron's weight vector, $\mathbf{v}_{ij}$ gets input from an element of the input array $\mathbf{X}_k$ where $\mathbf{X}_k$ is in the neuron's domain, i.e., $f_i(\mathbf{X}_k) = 1$. This correspondence is determined randomly or in order prior to learning.

Using this terminology we observe that the correspondence between left and right RFs can be any of the following four categories:

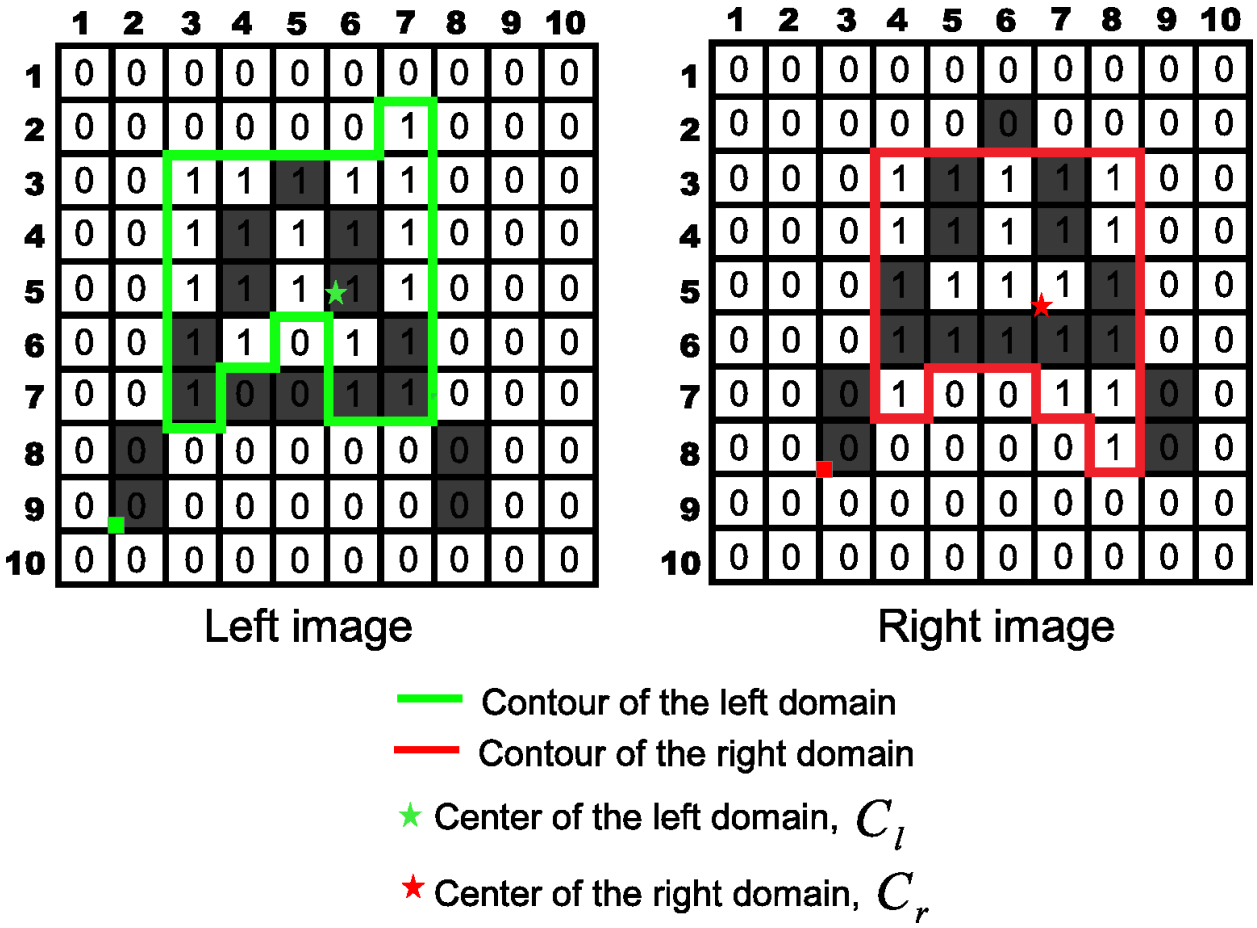


Figure 4.1: Domain function for a sample neuron. Left and right binary images of the letter **A** are shown where each pixel is depicted by a small square and the shade of the pixel indicates pixel intensity. The borders of the domain of the example neuron are marked by green and red lines in the left and right image, respectively. Value 1 for a pixel shows it is a part of the neuron's domain, and value 0 shows the opposite. The star marks the center the left and right domains (formulized in Eq. 4.14 and Eq. 4.15). The square marks show the lower left corner of the left and right images to highlight the horizontal and vertical disparities between the two images.

1. **Domains the same and weights the same:** No disparity encoding is possible in this case, except zero disparity, since left and right RFs are the same in both domain and weights.

2. **Domains different but weights the same:** Only disparities that involve similar weights are represented.

3. **Domains the same but weights different:** Only disparities with the same domains are represented.

4. **Domains different and weights different:** A whole gamut of disparities, including difference in domain or profile of left and right inputs, are represented.

It is clear that having different domains in the left and right RFs of a neuron is advantageous due to displacement of left and right projections of physical features. Also, because of occlusion and different viewing angles between the left and right eyes/cameras, there is often very different local profiles in the left and right images, even if they are from the same physical/distal features on the object. Therefore, an efficient coding of binocular disparity should be of Category (4). As mentioned earlier, Anzai et al. [6] found that in fact this is the case in biological neurons.

To our knowledge, no computational learning model has yet explained the development of Category (4) binocular neurons. Hoyer and Hyvärinen (2000) [42] presented a learning model for binocular receptive fields based on Independent Component Analysis (ICA). However, their model was limited to encoding only by different weights in left and right RFs (Category 3). Solgi and Weng (2010) [99] presented a developmental learning model which reached sub-pixel accuracy in binocular disparity detection. This model was still limited due to the use

of elongated left and right receptive fields (20×1 pixels) and not utilizing domain difference (still Category 3).

4.1.6 How to develop binocular neurons with domain and weight disparities

For the first time, in this article we present a learning model to address this important question. The proposed network is a neuromorphic model of the laminar architecture of the cortex. The synaptic learning algorithm proposes a novel idea for the dynamic retraction (cutting) and protraction (growing) of the synapses, which not only resembles the process of neural development in the brain, but also results in the formation of "domain difference" between the left and right RFs.

In the remaining of the paper, we first present the architecture and algorithms of the network in Section II. Then, Section III discusses the results of the network applied on large-scale datasets of images. Finally, Section IV summarizes and concludes the paper.

4.2 The Network Architecture

4.2.1 Dynamic Synapse Lobe Component Analysis (DSLCA)

Nearly all the conventional supervised and unsupervised learning algorithms, such as Principal Component Analysis (PCA), Support Vector Machines (SVM) and Lobe Component Analysis (LCA), assume a fixed number of input dimensions. This assumption requires the user of these algorithms, often laboriously, prepare datasets which contain only *relevant* dimensions. In a vision application, for example, this means cropping images so that they

have only the object of interest, e.g., human faces.

It is obvious that the aforementioned assumption greatly limits the effectiveness of the algorithms. For real-time practical applications, it is impossible to do such preprocessing. For example, an autonomous mobile robot equipped with cameras receives a stream of video in which most of the pixels (input dimensions) are irrelevant to the task at hand. Therefore, it is necessary to attend only to the relevant parts of the sensory data (foreground) and largely discard the irrelevant part (background). Although attention networks (e.g, WWNs) solve this problem at a macro level, i.e., attend to top-left corner of the input image, neurons still pick up part of the background, and therefore, develop inefficient representations with mixed foreground and background features. The goal of this part of the article is to use inspirations from neurophysiology to modify the LCA algorithm so it dynamically retracts irrelevant (e.g, background) synapses and grows relevant (foreground) synapses.

Fig. 4.2 illustrates the concepts of relevant vs. irrelevant feature dimensions for the case of stereo vision. For an efficient binocular receptive-field, we would like a neuron that picks up only the foreground and only the binocular area, i.e., the area marked with $\mathbf{F}_b$. As it is shown in [99], the original LCA algorithm will not be able to achieve this, since LCA does not differentiate between foreground and background dimensions of the input vector.

Wang et. al [105] showed that utilizing *synapse maintenance* in developmental networks can develop receptive fields that get input only from the foreground while cutting out the background. Wang et. al's results depend on the assumption that, statistically, variations in background pixels are considerably higher than those in foreground. We develop on that idea by adding the following extensions:

1. Local (as opposed to global) receptive fields that do not necessarily cover the entire foreground. This makes the assumption of less varying foreground more plausible.

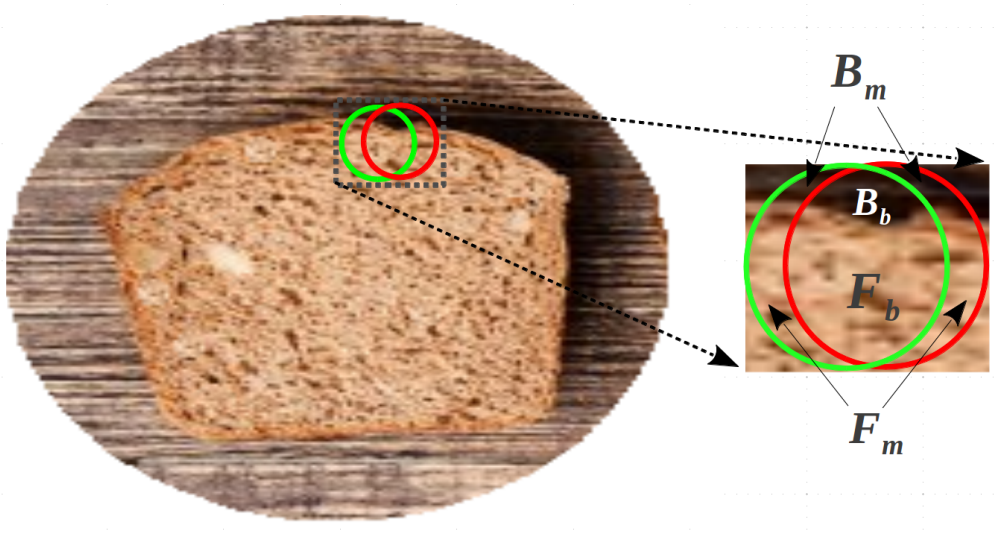


Figure 4.2: Demonstration of different parts of input to a binocular neuron. The image shows a slice bread (foreground) on a table (background). $\mathbf{B}_m$: background monocular, $\mathbf{B}_b$: background binocular, $\mathbf{F}_m$: foreground monocular, $\mathbf{F}_b$: foreground binocular. An efficient binocular neuron should pick up only the binocular part of the foreground, $\mathbf{F}_b$.

2. Both synapse growth and retraction instead of only synapse retraction.

3. Binocular receptive fields instead of monocular.

We use the term Dynamic Synapse Lobe Component Analysis (DSLCA) to refer to the modified version of LCA. Algorithm 1 shows the algorithm used for synapse retraction and growth in DSLCA.

Every synapse incrementally computes a parameter named the *variance* of the synapse (σ_i in Algorithm 1). The variance is the main parameter that controls the cutting (pruning) and growing (expansion) of the synapses. We would like to prune the synapses with high variance and expand those with low variance. High variance in a synapse's input signifies unstable input patterns which should be ignored (e.g., background or noise), hence the pruning. On the other hand, low variance synapses receive stable input that correlates with the neuron's representation of the input space, hence the expansion. In order to discriminate high and low synapstic variance, we set dynamic thresholds for each as a proportion of the average

variance across all the synapses in a neuron. Variance values less than $\xi_1\bar{\sigma}$ are considered low variance and variance values greater than $\xi_2\bar{\sigma}$ are considered high variance in Algorithm 1, where $\bar{\sigma}$ captures the average variance computed incrementally. The *synaptogenic factor* ($f(\sigma_i, \bar{\sigma})$ in Algorithm 1) is used to scale the synaptic weight when a synapse's variance is in between the two thresholds. The synaptogenic factor is an inverse function of the variance and insures a smooth transition from cutting to scaling to growing of the synapses.

Algorithm 1 Dynamic Synapse Lobe Component Analysis

for time steps t of the network running **do**
  for neurons n in the internal Y area **do**
    Perform LCA updating using synapse age
    if n is allowed to fire **then**
      for i'th synapse in neuron n **do**
        Incrementally update σ_i (Eq. 4.1)
      Compute $\bar{\sigma}(n)$ (Eq. 4.3)
      for i'th synapse in neuron n, n_i **do**
        Compute the *synaptogenic factor* (Eq. 4.4)
        if $\sigma_i > \xi_2\bar{\sigma}$ **and** synapse_age $> n_c$ **then**
          Cut the synapse n_i
        else if $\sigma_i < \xi_1\bar{\sigma}$ **and** synapse_age $> n_g$ **then**
          Grow synapses in the neighborhood of n_i
        else if $\xi_1\bar{\sigma} \leq \sigma_i \leq \xi_2\bar{\sigma}$ **then**
          $v_i \leftarrow f(\sigma_i, \bar{\sigma})v_i$

Below are the equations used in Algorithm 1.

$$\sigma_i(n) = \begin{cases} 0 & \text{if } n \leq n_0 \\ w_1(n)\sigma_i(n) + w_2(n)|v_i - p_i| & \text{otherwise} \end{cases} \tag{4.1}$$

where

$$w_2 = \frac{1 + \mu(n)}{n}, w_1(n) = 1 - w_2(n)$$

$$n_0 = 10$$

and $\mu(n)$ is the amnesic function, calculated as the following:

$$\mu(n) = \begin{cases} 0, & \text{if } n < t_1 \\ c(n - t_1)/(t_2 - t_1), & \text{if } t_1 < n < t_2 \\ c + (t - t_2)/r, & \text{if } n > t_2 \end{cases} \tag{4.2}$$

where $t_1 = 10$, $t_2 = 10^3$, $c = 2$ and $r = 10^4$.

$$\bar{\sigma}(n) = \frac{1}{N} \sum_{i=1}^{N} \sigma_i(n). \tag{4.3}$$

where N is the number of dimensions in the input vector.

$$f(\sigma_i, \bar{\sigma}) = \begin{cases} \frac{1}{\sigma_i + \epsilon} - \frac{1}{\xi_2 \bar{\sigma} + \epsilon} & \text{if } \sigma_i < \xi_1 \bar{\sigma} \\ k(\frac{1}{\sigma_i + \epsilon} - \frac{1}{\xi_2 \bar{\sigma} + \epsilon}) & \text{if } \xi_1 \bar{\sigma} \leq \sigma_i \leq \xi_2 \bar{\sigma} \\ 0 & \text{if } \sigma_i > \xi_2 \bar{\sigma} \end{cases} \tag{4.4}$$

where $\xi_1 = 0.9$, $\xi_2 = 1.5$, k is a constant to guarantee the function is continuous, and ϵ is a very small positive constant to avoid division by zero.

4.2.1.1 Synapse age versus neuron age

The original LCA algorithm (Eq. 4.5), as well as most other online learning methods, uses the number of times a neuron has fired, the neuronal age, to calculate the retention and learning rates (Eq. 4.6).

$$\mathbf{v}_j(t) = \omega_1 \mathbf{v}_j(t-1) + \omega_2 y_j \mathbf{x}(t), \tag{4.5}$$

$$\omega_1 = 1 - \omega_2, \omega_2 = \frac{1 + \mu(n_j)}{n_j}, \tag{4.6}$$

Since in the Dynamic Synapse extension of LCA synapses are allowed to grow and retract all the time, a global neuronal age applies only to the "oldest" synapse. The other synapses should use the frequency of the neuron firing after they were grown as their age—A neuron maybe 1000 updates old, but a newly grown synapse is perhaps only 2 updates old. Hence, Eq. 4.5 and 4.6 change to the following for Dynamic Synapse LCA:

$$\mathbf{v}_{ij}(t) = \omega_1 \mathbf{v}_{ij}(t-1) + \omega_2 y_j \mathbf{x}_j(t), \tag{4.7}$$

$$\omega_1 = 1 - \omega_2, w_2 = \frac{1 + \mu(n_{ij})}{n_{ij}}, \tag{4.8}$$

where n_{ij} is the "age" of the i'th synapse of the j'th neuron.

4.3 Analysis

In order to quantify the disparity selectivity of the DSLCA neurons, we introduce the following two metrics for domain and weight disparities, respectively.

4.3.1 Measure for weight disparity

To quantify the weight disparity selectivity of a binocular neuron to a specific disparity $\mathbf{d} = (d_x, d_y)$, we compute the normalized cross-correlation of its left and right RFs while the left RF is shifted by $\mathbf{d}$. Formally, let us assume $\mathbf{v}_i^l$ is the weight vector of the set of all synapses in the left RF of neuron n_i:

$$\mathbf{v}_i^l = (\mathbf{v}_{ij} | \mathbf{v}_{ij} \in L(n_i)) \tag{4.9}$$

where $\mathbf{v}_{ij}$ is the weight of the j'th synapse of the neuron n_i and $L(n_i)$ is the set of all synapses of the neuron in the left RF, ordered by their location. Similarly, the weight vector of all the synapses in the right RF of n_i is:

$$\mathbf{v}_i^r = (\mathbf{v}_{ij} | \mathbf{v}_{ij} \in R(n_i)) \tag{4.10}$$

where $R(n_i)$ is the set of all synapses of the neuron, n_i, in the right RF, ordered by their location[1]. We define:

$$\mathbf{d}(n_i, dx, dy) = \mathbf{v}_i^l \otimes \mathbf{v}_i^r(dx, dy) \tag{4.11}$$

where $\otimes$ is the normalized cross-correlation of two vectors defined as the following for two arbitrary vectors $\mathbf{f}$ and $\mathbf{g}$, where $\mathbf{g}$ is shifted by (dx, dy):

$$\mathbf{f} \otimes \mathbf{g}(dx, dy) = \frac{1}{N} \sum_{x,y} \frac{\big(\mathbf{f}(x, y) - \bar{\mathbf{f}}\big)\big(\mathbf{g}(x + dx, y + dy) - \bar{\mathbf{g}}\big)}{\sigma_{\mathbf{f}} \sigma_{\mathbf{g}}} \tag{4.12}$$

where N is the number of dimensions in $\mathbf{f}$ and $\mathbf{g}$. Also, $\bar{\mathbf{f}}$, $\bar{\mathbf{g}}$, $\sigma_{\mathbf{f}}$ and $\sigma_{\mathbf{g}}$ are the mean and standard deviation values of all the elements in the vectors $\mathbf{f}$ and $\mathbf{g}$.

Finally, the weight disparity of the neuron n_i is defined as the value of (dx, dy) (a 2D vector) where the normalized cross-correlation of its left and right weight vectors are maximized, i.e.,

$$weight\_disparity(n_i) = \max_{(dx,dy)} \mathbf{v}_i^l \otimes \mathbf{v}_i^r(dx, dy) \tag{4.13}$$

[1]Such ordering is critical to establish correspondence between the same indices of the left and right RFs. e.g., $\mathbf{v}_{i1}^l$ and $\mathbf{v}_{i1}^r$ must be connected to the same locations in the left and right input images, respectively.

The weight correlation map of a neuron is a map of the normalize cross-correlation of the left and right weight vectors (Eq. 4.11) while the disparity vector (dx, dy) slides in a 2D window of size $(w, h) = (16, 16)$.

4.3.2 Measure for domain disparity

For each neuron, n_i, we measure the *center of mass* of the left RF, $\mathbf{C}_l(n_i)$ as the following:

$$\mathbf{C}_l(n_i) = E\big(loc(\mathbf{v}_i^l)\big) \tag{4.14}$$

where E denotes the expectation (mean), $\mathbf{v}_i^l$ is weight vector of the synapses in the left RF, defined in Eq. 4.9, and *loc* returns the (x, y) coordinate of the synapse. Similarly, the center of mass for the right RF is defined as:

$$\mathbf{C}_r(n_i) = E\big(loc(\mathbf{v}_i^r)\big) \tag{4.15}$$

where $R(n_i)$ is is defined in Eq. 4.10. Finally, domain disparity is measured by the difference between the center of mass of the left and right RFs:

$$domain\_disparity(n_i) = \mathbf{C}_l(n_i) - \mathbf{C}_r(n_i) \tag{4.16}$$

The domain disparity is a 2D vector where the first and second elements represent horizontal and vertical disparities, respectively. See Sec. 4.4.3 for example results of developed weight and domain disparities in our experiments.

4.4 Results

4.4.1 Toy Example: Moving Receptive Fields

To illustrate how DSLCA results in binocular RFs which dynamically relocate themselves, we first present a toy example. The example demonstrates the ability of the proposed network to dynamically retract and grow synapses to cover the locally relevant parts of the input.

A 30×30 pixel image was randomly selected from a dataset of natural images[2] as background. Then a 16×16 pixels circular foreground was also randomly selected from natural images and was superimposed on the left and right background images at different disparities. Fig. 4.3 shows that, after learning, only a part of the synapses in the original circular receptive field ($t = 0$) will survive while some are retracted (blue) and new synapses (green) grow until both left and right receptive fields cover the entire foreground. There was $15 \times 15 \times 10$ neurons in the network.

4.4.2 Natural Video: Ragged and Displaced Receptive Fields

In this section we present the results of applying DSLCA to natural binocular videos. A network of $100 \times 15 \times 15 = 22,500$ DSLCA neurons was used in this experiment. The initial receptive fields of neurons were two (left and right) circular areas of radius 8 pixels (approximately $2 \times \pi \times 8^2 \approx 400$ connections per neuron). The training input to the network was from binocular videos taken by Minoru stereo camera, a low-cost commercial webcam. A total of 50,000 binocular video frames, each of size 100×100 pixels, were used to train the network for the presented results. A few samples of the videos are shown in Fig. 4.4. The green squares in Fig. 4.4 which resemble the focus of attention (or fovea) of the agent

[2]Available from http://research.ics.tkk.fi/ica/imageica/

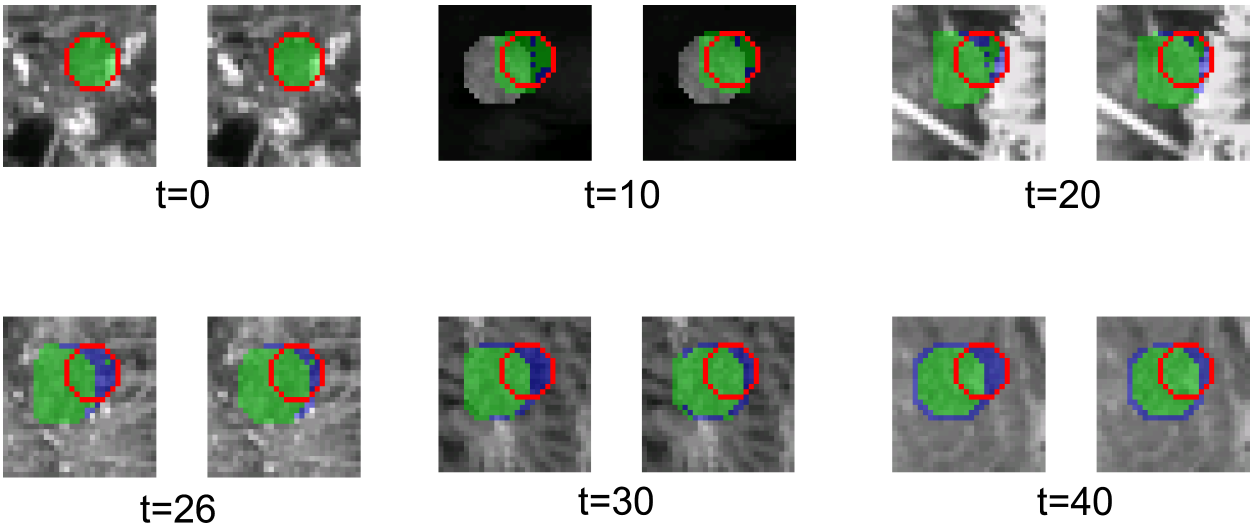


Figure 4.3: An intuitive illustration of the mechanisms of the DSLCA. Each image pair shows left and right views of a circular foreground area and a changing background. The color-coded pixels show the receptive field of one neuron in the large network. At time $t = 0$, the left and right receptive fields both have a circular contour. As the simulation progresses, the synapses connected to background area die (blue pixels) while new synapses grow to the foreground area (green pixels). Note that only the video frames for which the neuron has fired are shown in this illustration. For the majority of iterations (95% of video frames) the neuron does not fire. Those iterations are not shown here.

were selected randomly on the object, from the exact same (x, y) coordinate in left and right images.

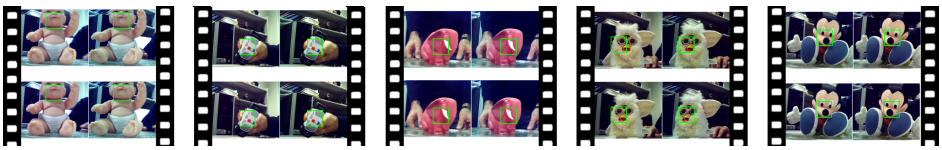


Figure 4.4: A few video frame examples of the input to the network. Each pair shows the left and right images, while the green square shows the center of attention of the learning agent, simulating fovea. The center of attention is at exactly same position in the left and right images. However, features captured in the green square are slightly shifted, both horizontally and vertically, due to disparity.

Fig. 4.5 illustrates the developed receptive fields (weights) of a sample of 15×15 neurons (one out of 100 layers) after development. As it can be seen from the two zoomed-in neurons, the developed RFs have changed from an initial circular shape to arbitrary ragged shapes

both in left and right domains. Each developed neuron is selective to following three aspects of the input: 1) location, due to local connectivity of weights 2) pattern, due to various developed weight patterns 3) disparity (will be quantified below). We say the neurons are location-specific, pattern-specific and disparity-specific.

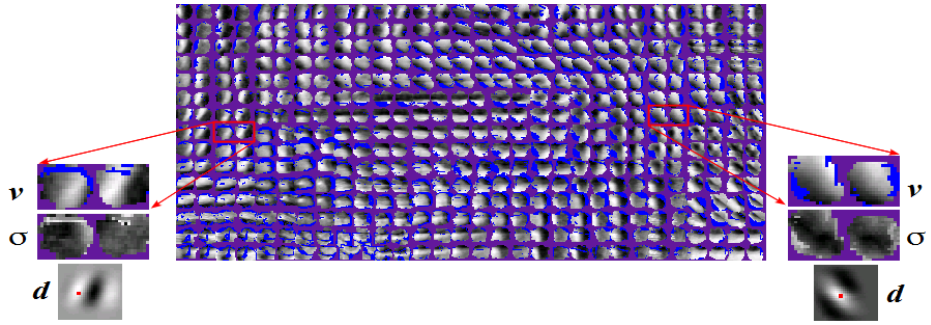


Figure 4.5: Weight map of a 15×15 grid of developed neurons. Each pair (e.g., the two pairs highlighted in red boxes) represents the left and right receptive fields of a binocular neurons. Note that the initial circular receptive fields are distorted after development. Blue pixels represent synapses which are retracted (dead synapses) due to the mechanisms in DSLCA algorithm (See Algorithm 1). **v**: weight vector of the highlighted neuron, zoomed-in. σ: visualization of the synapse deviation ($\sigma_i(n)$ in Eq. 4.1) for each synapse (live or dead) of the highlighted neuron. Note that the synapses with highest deviation (bright pixels) are retracted. **d**: The correlation coefficient map of the left and right RFs of the highlighted neuron, computed according to Eq. 4.11. The red dots on the maps indicate the highest correlation.

4.4.3 Developed weight and domain disparities

Fig. 4.5**d** shows the weight disparity map of the highlighted neurons, and Fig. 4.6 shows the weight disparity map of a grid of neurons, defined in Sec. 4.3.1. In both figures the red dot indicates the highest correlation value, the location of which is the weight disparity of the neuron (Eq. 4.13).

The calculated domain disparities for a grid of 15×15 neurons, defined in Sec. 4.3.2 is

drawn in red vectors in Fig. 4.7.

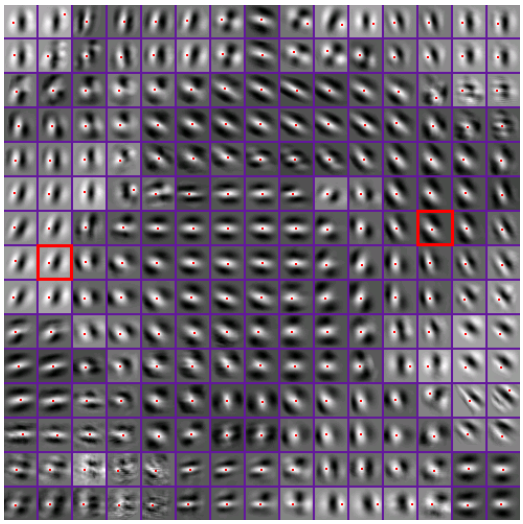


Figure 4.6: The weight correlation map (Eq. 4.11) of a 15×15 grid of developed neurons (same neurons as in Fig. 4.5). This map is a measure of *weight disparity*. The highest point for each neuron (indicated by a red dot in each small square) represents the highest weight disparity tuning for the neuron (Eq. 4.13). The two red boxes represent the same neurons highlighted in Fig. 4.5. Red dot at the exact center of a square means the neuron is selective to zero disparity both in horizontal and vertical directions, while deviation of the red dot from the center represents disparity selectivity (both horizontal and vertical).

If we treat the weight and domain disparities of each neuron as a 2D vector (horizontal and vertical disparities), such vectors easily visualize the disparity selectivity of the neuron. Fig. 4.7 shows a quiver map of both weight and domain disparities for a 15×15 grid of neurons. The addition of the two disparity vectors (shown in green in Fig. 4.7) represents the composite selectivity of a neuron when both weight and domain disparities are taken into account.

To verify the meaningfulness of the developed disparity representations depicted in Fig. 4.7, we tested the network on a pair of images and plotted the disparity representation of the winner neurons. See Fig. 4.8.

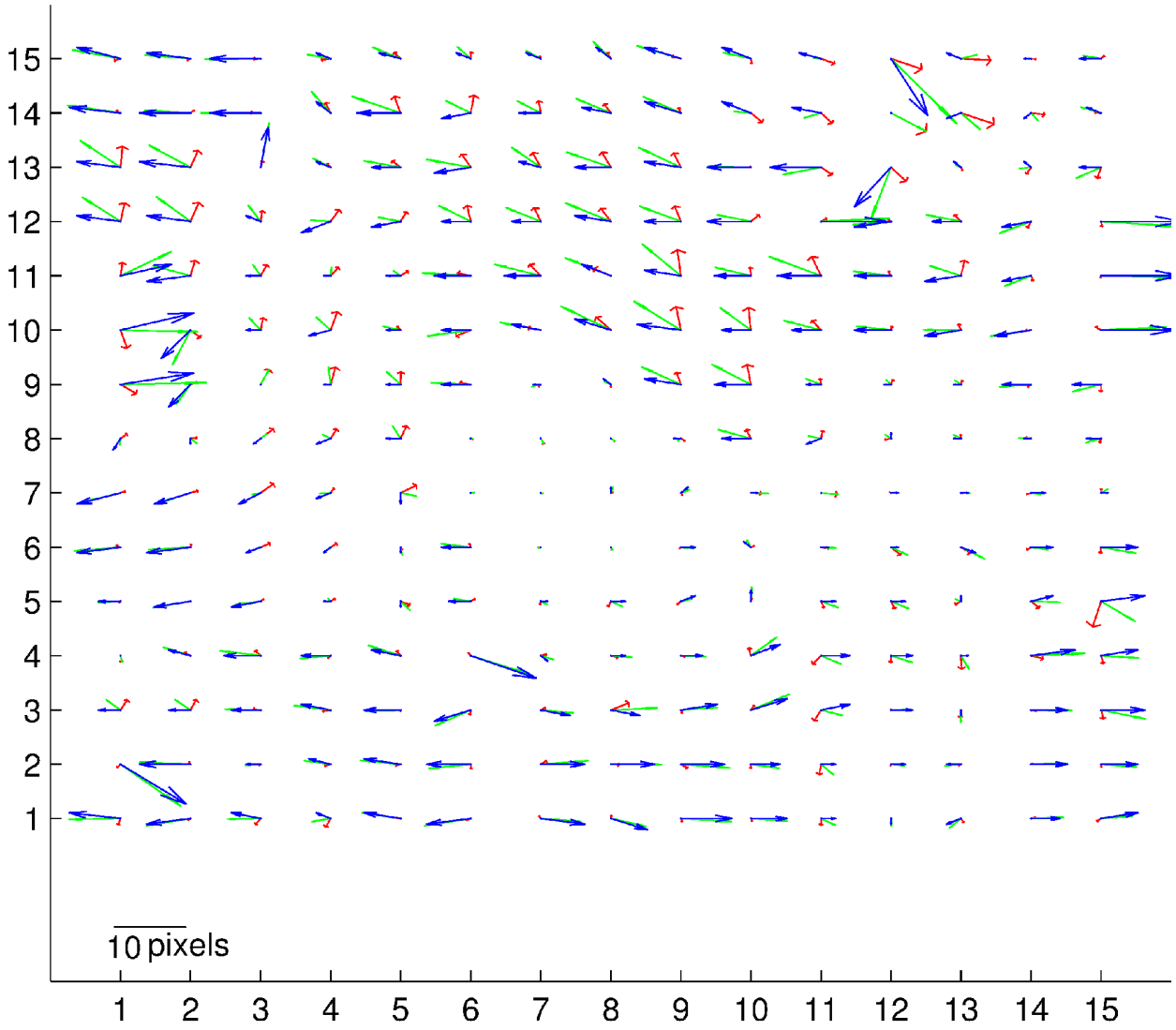


Figure 4.7: Disparity selectivity of a 15 × 15 grid of developed neurons (same neurons as in Figs. 4.5 and 4.6). Weight disparity (blue arrows, Eq. 4.13) are based on the highest correlation point between the left and right RF (represented by red dots in Fig. 4.6). Domain disparities (red arrows) are calculated as the difference between the "center of the mass" of the left and right RFs of the neuron (Eq. 4.16)

.

4.5 Conclusion

For the first time, this paper presents a model of the development of disparity selectivity while the connections, i.e., synapses, can dynamically grow or retract to better represent the input space. The adoption of the dynamic synapse neurons results in the development of

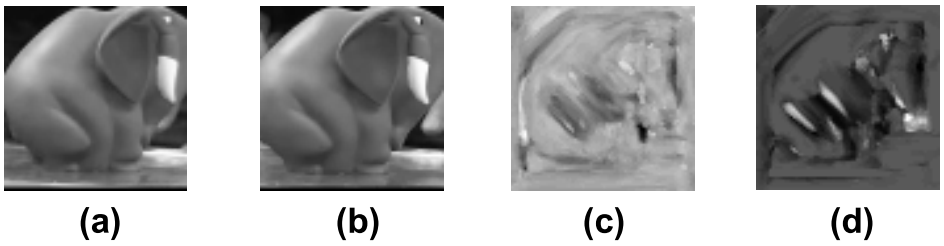


Figure 4.8: Estimated disparity map using the unsupervised learning network. **(a)** left image **(b)** right image **(c)** depth map mid-development **(d)** depth map after development.

binocular receptive fields which are not only selective to the phase shift in the left and right images (weight disparity), but for the first time, they also encode selectivity to the position of the features in the left and right inputs (domain disparity).

There have been numerous cortex-inspired learning networks developed in the past few decades, e.g., [125]. Previous versions of WWNs have successfully handled simultaneous perception of type, location and scale of the visual objects [46, 63, 108]. The large scale of the WWN network used here (9 million weight parameters) and its successful application on Big Data (50 thousand stereo video frames) is a promising first step towards creating a real time stereo system for simultaneous shape, type and location perception.

Chapter 5

Stereo Network for Shape and Disparity Detection

In this part of the thesis, we aimed at creating an end-to-end system for simultaneous disparity and shape recognition. We utilized the *dynamic synapse* mechanisms developed in our previous work (Chapter) for background elimination as well as more efficient binocular feature extraction and a two-pathway Where-What Network for separate representation of the where information (location and disparity here) and the what information (shape here).

5.0.1 Importance and Novelty

The traditional (and intuitive) approach for shape recognition in computer vision has been to infer the shape of the objects based upon one of the multiple depth cues such as shading, binocular disparity, motion, etc. This has created an extensive literature in shape recognition, named *shape-from-X* where X is one of the cues. Our approach, however, is drastically different. Instead of laboriously hand-crafting feature detectors for X, say X=shading, the network builds a holistic representation of as many of the cues as possible and associates them with the object shapes. This approach is indeed more similar to the developmental learning process in biological visual systems; i.e., an animal's visual system uses all the available depth cues, in an integrated fashion, to create a 3D representation of an attended

object.

To our knowledge, an integrated learning system for detection of shape, disparity and 2D location of visual objects is unprecedented in the literature. Moreover, being inspired by the developmental processes and the cortical architecture of human vision, this work is a step towards a better understanding of biological stereo vision.

5.1 Network Architecture

Similar to previous versions of the Where-What Networks, the neural networks used in this work have an internal feature detection area, a dorsal pathway to represent where/location information, and a ventral pathway to represent what/type information. See Fig. 5.1.

The internal feature detection area gets bottom-up information from the left and right input images. Each neuron has an initial bottom-up circular local receptive field of a fixed initial diameter. Indeed, the shape of bottom-up receptive field changes according to the DSLCA algorithm described in Chapter . There are two-way global connections between the internal area and the where area, as well as between the internal area and the what area. In Fig. 5.1, bottom-up connections are shown in red, and top-down connections are shown in blue. The where area is a 3-dimensional cube of neurons in which first, second and third dimensions represent horizontal location (x), vertical location (y) and disparity (d), respectively. The what area is a number of neurons (5 in this case) each representing a certain object shape.

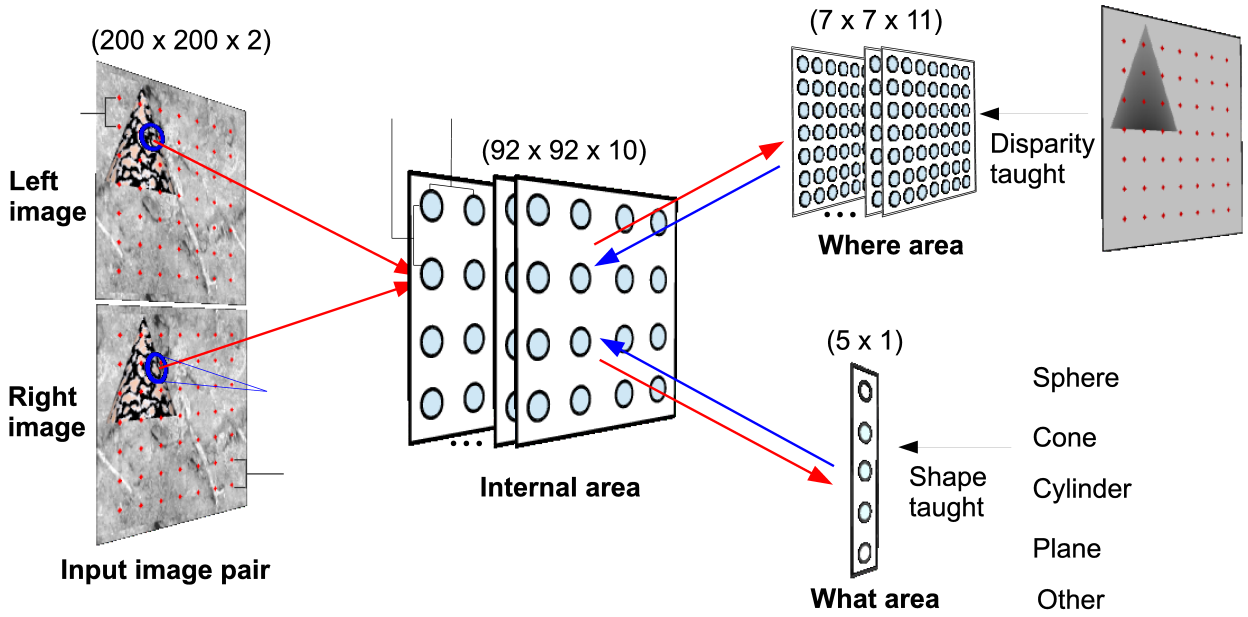


Figure 5.1: Schematic diagram of the Where-What Network used in the experiments. Input was an image pair of 200×200 pixels each, where background was a random patch from natural images and foreground was a quadratic shape generated by POV-Ray [1]. There were $92 \times 92 \times 10$ neurons in the internal area, each neuron taking a circular patch of diameter 17 from each of the left and right images. The where area has $7 \times 7 \times 11$ neurons which represent 7×7 locations and 11 different discrete disparity values at each location, disparity values $-5, ..., 0, ..., +5$. Disparity quantization on the image was such that each disparity index was one pixel different from its neighbors. The small red dots on the training disparity map (top, right) correspond to the red dot locations marked on the left and right images. The what area has 5 neurons representing the shape classes "sphere", "cone", "cylinder", "plane" and "other". There are two-way global bottom up connections between the internal area and both the where and what areas. The number of neurons in the internal and the where and what areas are chosen based on the limitation in our computational resources. The model, however, is not limited to any specific size parameters.

5.2 Experiments

In order to realize the capability of the WWNs in detecting the shape, location and disparity of visual objects in a complex background, the following experiments were conducted.

5.2.1 Input images

3D scenes of objects of basic shapes on backgrounds were generated using a powerful ray tracing program called the Persistence of Vision Raytracer, or POV-Ray. None of the publicly available stereo image datasets were appropriate for the purpose of our experiments. Using this tool gave us the flexibility of having an abundant source of training images/videos. We tried to make the scenes as natural as possible by using a single light source (as opposed to ambient) and using natural images for texture by wrapping images of natural scenes around the objects [41]. 10 different texture types (an even mixture of natural image and synthetic textures) were used in the experiments. See Fig. 5.2.

The input image pairs were 200×200 pixels each which was relatively large compared with other neural networks used for computer vision applications.

There were 5 shape classes "sphere", "cone", "cylinder", "plane" and "other", where "other" was any shape other than the four main shapes. Each shape was painted by one of the 10 textures and was presented in one of the 7×7 locations (marked by red dots in Fig. 5.1.

5.2.2 Internal area

There were $92 \times 92 \times 10$ neurons in the internal area where each neuron had a circular local bottom-up receptive field of diameter 17 from each of the left and right input images. In

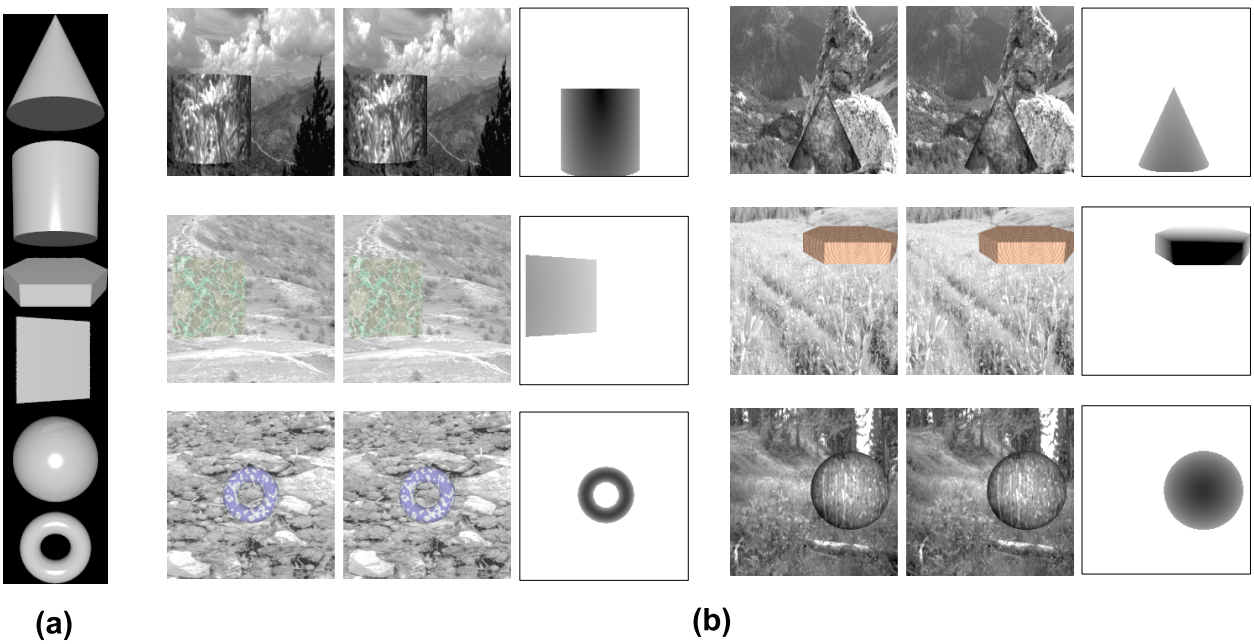


Figure 5.2: **(a)** The basic shapes used in the experiments. There were four main classes; "sphere", "cone", "cylinder" and "plane" and the class "other" which could be shapes such as hexagon and donut shape. **(b)** Sample input images to the network. Each of the six pairs shows left and right images of a scene where a shape is placed against a background in one of the 7×7 locations. Also, the disparity map used during training for each pair is shown to its right. The darker a pixel in the disparity map, the closer the point. The background texture is a random patch of natural images taken from the 13 natural images database [41]. The foreground texture is an even mixture of synthetic (but natural-looking) textures, generated by POV-Ray, and natural image textures from the same image set [41].

addition, each neuron had global connections to all the neurons in both the where and what areas. The initial center of receptive field for each neuron was 2 pixels different from its immediate neighbors. Therefore, the entire image was covered by the internal area neurons (92×2 covers 184 pixels and the remaining 16 pixels are left for padding around the image). The deep 10 layers in the internal area were necessary in order for the network to be able to capture all the variations in texture and disparity in the bottom-up signals.

5.2.3 Where area

There were $7 \times 7 \times 11$ neurons in the where area. Each of the 7×7 neurons in each layer of the area represented one of the 7×7 locations on the image (marked by red dots both on the training disparity map and the input image pair in Fig. 5.1). Each of the 11 neurons along the third dimension represents one of the disparities in range -5 to $+5$.

5.2.4 What area

There were 5 neurons in the what area, each representing one of the object shape classes.

It is important to mention that the limited size of the network is due to limited computational resources rather than an inherent limitation in our model. Namely, the 2 pixels staggering distance of receptive-field centers in the internal area is due to limitation in the number of neurons we could afford to have in the internal area so that our simulations would run in a manageable time. Similarly, the 7×7 number of locations in the where area is chosen because of limitation of our computational resources. According to our theory, the network can handle any number of locations given enough resources.

5.3 Results

Thanks to dual optimality of the Where-What Networks [114], and the advantage of the DSLCA algorithm for foreground/background separation [105] and binocular disparity feature extraction [100], the network learns to recognize object shape, location and disparity with impressive accuracy. Fig. 5.3 plots the recognition rate of the network for shape detection (dotted curve) and disparity error on disparity detection of the stereo pair (solid curve). Disparity error was computed as the root mean square error (RMSE) of estimated disparity for each of the 7×7 points on the image.

Fig. 5.4 shows the decline of location error as a function of training epochs. After 3 epochs location error fell below 10 pixels. To compute the location error, the average of the row-column location of all the winning neurons in the where area was considered as "detected" location. This values was contrasted with the row-column location of the centroid pixel of the foreground object. The error was calculated as RMSE error, same as disparity error explained above.

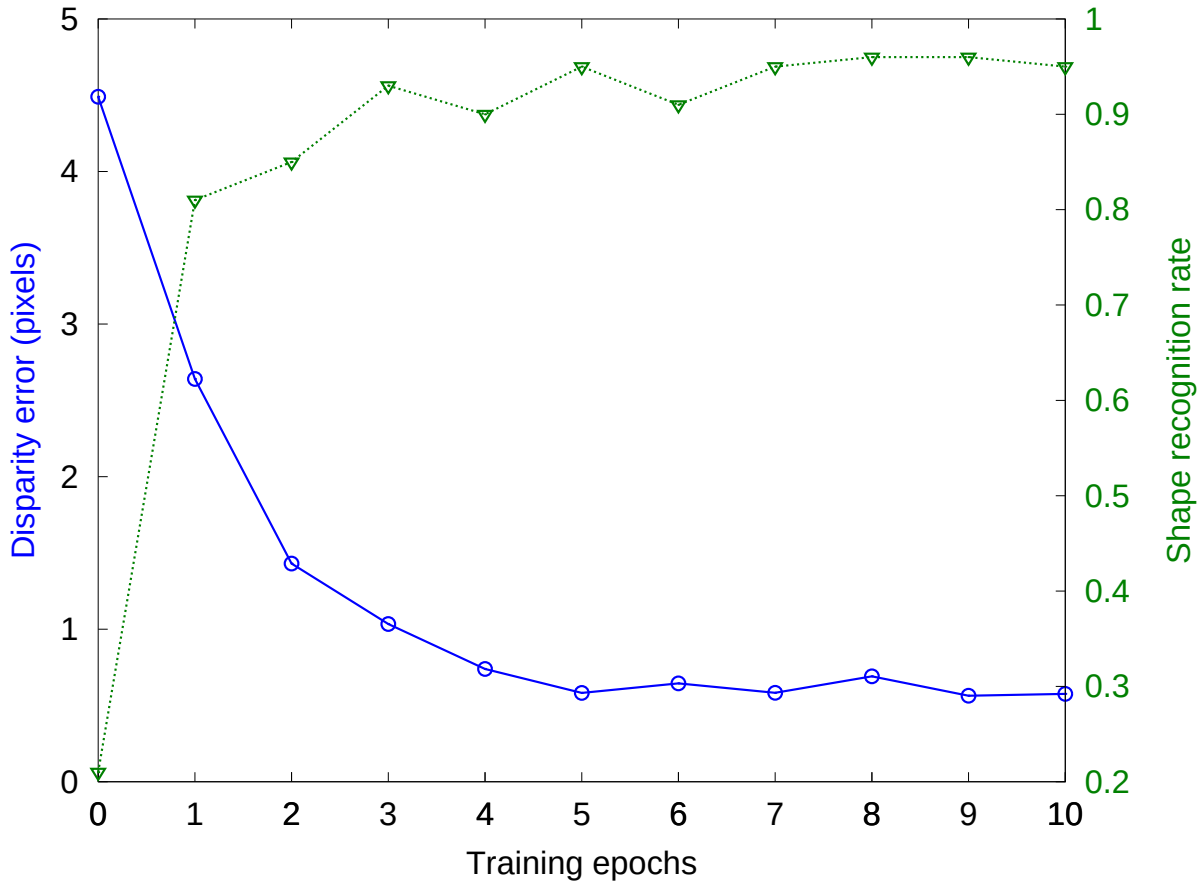


Figure 5.3: Simultaneous shape and disparity recognition by the network. The figure shows disparity error, computed as the root mean square error (RMSE) of the detected disparity, and recognition rate, the ratio at which the network reports the correct object shape, both in disjoint testing.

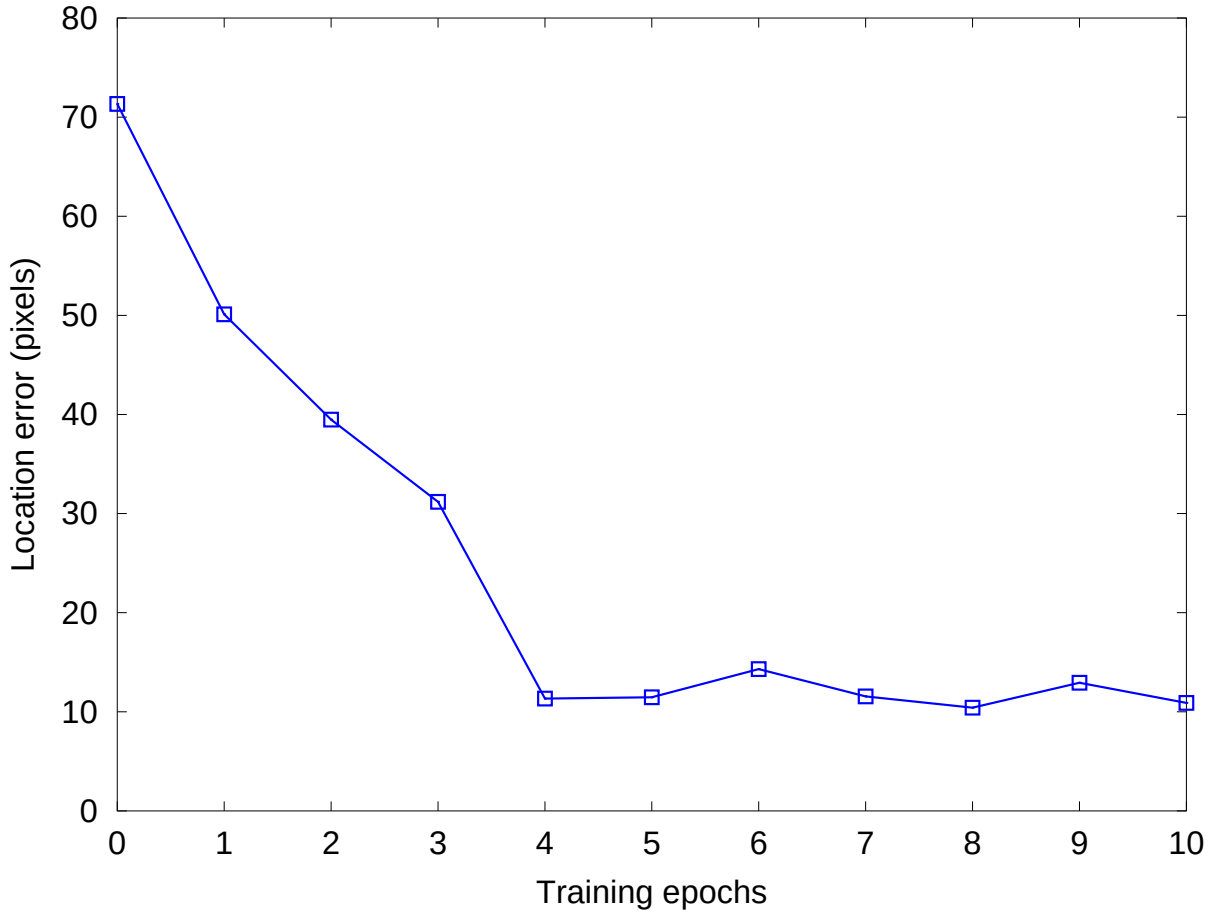


Figure 5.4: Error at detecting the location of the foreground object. The center of all the firing neurons in the Where area was considered as the detected location, and it was contrasted with the centroid of the foreground object (figure) to compute the distance error.

Chapter 6

Concluding Remarks

This thesis presented theory as well as computational experiments and results on learning models of the visual cortex. In Chapter 2, a model for specificity and transfer in perceptual learning was laid out. For the first time, we put forward a detailed computational model of perceptual learning that successfully explained how gated self-organization can result in both specificity and transfer in perceptual learning experiments [97]. Our model replicates the data from human subject studies (See Section 2.4).

The main focus of the thesis, however, was on integrated shape, disparity and location perception utilizing multiple depth cues, especially binocular disparity. Our early work on this subject (Chapter 3) achieved sub-pixel accuracy in challenging natural image datasets. This work was significant both in its approach to disparity detection, i.e., learned filters and non-explicit matching, and the engineering-grade performance of the developed disparity detector. We then focused on unsupervised binocular feature learning in our stereo networks. The detailed study of feature learning for stereo input led us to introduce the concepts of domain versus weight disparities. This approach was consistent with our knowledge of neurobiology and resulted in better representation of horizontal and vertical disparity space (Refer to Chapter 4 for more details). Finally, in Chapter 5 we devised a version of the Where-What Networks for integrated stereo shape, location and disparity detection. The promising results presented in Section 5.3 are a strong first step towards creating a stereo

detection and recognition system for real-time applications in engineering.

The novelty of the completed Dissertation is summarized below:

leftmargin=2cm, rightmargin=0cm Stereo Lobe Component Analysis with both synapse retraction and *synapse growth*.

leftmargin=2cm, rightmargin=0cm The use of stereo *local receptive-fields* in a developmental network for stereo.

leftmargin=2cm, rightmargin=0cm Integration of *multiple depth cues*, e.g., binocular disparity, perspective, shading, etc. in a single unified network for generating behaviors that involve type, shape and location.

leftmargin=2cm, rightmargin=0cm Simultaneous *shape, disparity, location and type* perception in complex, natural image background without using separate procedures for each.

leftmargin=2cm, rightmargin=0cm Allow these four types to help each other in the network.

The importance of the work completed in this thesis can be summarized in the following two points:

- In terms of performance, the integration of the four sources of information, i.e., shape, disparity, location and type, realized a robust vision system with an increased performance over cost ratio.
- In terms of representation, the cortex-inspired novel architectures and algorithms resulted in a superior representation of both the top-down and the bottom-up information, as evident from the improved results.

This thesis was an important, although small, step towards understanding the underlying mechanisms of human depth perception, especially stereoscopic depth perception, and applying it to create an engineering-grade system for computer vision and robotics applications. For the first time, we showed that a cortically inspired neural model can learn to detect shape, disparity and location of objects on complex backgrounds given a training procedure that was consistent with developmental learning.

6.1 Limitations and Future Work

Despite the encouraging performance reported in Chapter , this thesis is far from a comprehensive model of the brain. However, we believe that it is a necessary step towards an engineering model of how the visual cortex makes sense of the visual world.

A noticeable limitation of the model is its lack of scalability to match the capabilities of human vision. With the current settings of the experiments (only 5 object classes, 49 locations and 11 disparity levels) the network has around 9 million connections. Moreover, the number of connections (synapses) in our model grow linearly with the number of neurons in the output areas (where and what). On the other hand, the developed human visual system is capable of recognizing around 30,000 object classes with locations distinguishable with one minute of a degree (one-sixtieth of a degree) difference in visual angle [39], hundreds of different disparity magnitudes as well as enormous variation in orientation, scale, etc. Simple calculations show that to extend our networks to human-level performance in vision, we would need several orders of magnitude more connections in a modeled network than in the human visual pathways to hypothetically match the human-level performance. This shows a fundamental shortcoming in our models, aside from the need for faster computers.

Other limitations of the presented networks include lack of deep hierarchical architecture and their slow updating rate on a single-core computer. Although it is known that the human visual system has a hierarchical deep structure [27], we used a traditional 3-layer architecture (input, internal and output layers) for the sake of keeping updating rate and training time manageable in our single-core computers. Even with this relatively simplified architecture the updating rate was 10 to 15 second which is far from real-time.

The research presented in this thesis deserves to be expanded and tested further in future works. Although the WWNs are capable of handling multiple objects (shown in [63]), trying multiple objects in a stereo setting for shape recognition can be the subject of future work. Moreover, it will be interesting and revealing to investigate the effects of more challenging datasets on the performance of the network to better understand how and when our methods fail. For example, future work can study how recognition and detection rates change as a function of partial occlusion of the foreground object, texture-less objects, or having backgrounds which resemble the trained foreground objects. Given the inherently highly parallel nature of our algorithms, future work could also take advantage of commercial Graphical Processing Units (GPUs) for parallelizing the network computations. This way, it will be possible to achieve real-time performance in networks of the size of the one used in experiments reported in Chapter 5 or larger.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] The persistence of vision raytracer (pov-ray). `http://povray.org/`. Accessed: 2013-09-26.

[2] Yael Adini, Dov Sagi, and Misha Tsodyks. Context-enabled learning in the human visual system. *Nature*, 415(6873):790–793, February 2002.

[3] M. Ahissar and S. Hochstein. The reverse hierarchy theory of visual perceptual learning. *Trends in Cognitive Sciences*, 8(10):457–464, October 2004.

[4] Merav Ahissar and Shaul Hochstein. Task difficulty and the specificity of perceptual learning. *Nature*, 387(6631):401–406, May 1997.

[5] A. Ohzawa I. Freeman R.D. Anzai. Neural mechanisms underlying binocular fusion and stereopsis: position v/s phase. In *Proc. Natl. Acad. Sci.*, pages 5438–5443, 1997.

[6] Akiyuki Anzai, Izumi Ohzawa, and Ralph D Freeman. Neural mechanisms for encoding binocular disparity: receptive field position versus phase. *Journal of Neurophysiology*, 82(2):874–890, 1999.

[7] K. Ball and R. Sekuler. Direction-specific improvement in motion discrimination. *Vision Research*, 27(6):953–965, 1987.

[8] P.O. Bishop. Vertical disparity, egocentric distance and stereoscopic depth constancy: a new interpretation. In *Proc. R. Soc. London Ser.*, pages 445–469, 1989.

[9] H.B. Blakemore, C. Pettigrew, and J.D. Barlow. The neural mechanisms of binocular depth discrimination. *J. Physiol.*, 193:327–342, 1967.

[10] W. H. Bosking, Y. Zhang, B. Shoefield, and D. Fitzpatrick. Orientation selectivity and arrangement of horizontal connections in tree shrew striate cortex. *Journal of neuroscience*, 17:2112–2127, 1997.

[11] E. M. Callaway. Local circuits in primary visual cortex of the macaque monkey. *Annual Review of Neuroscience*, 21:47–74, 1998.

[12] Edward M. Callaway. Feedforward, feedback and inhibitory connections in primate visual cortex. *Neural Netw.*, 17(5-6):625–632, 2004.

[13] C.D. Gilbert and T.N. Wiesel. Microcircuitry of the visual cortex. *Annu. Rev. Neurosci.*, 6:217–247, 1983.

[14] G. Chen, H. D. Lu, and A. W. Roe. A map for horizontal disparity in monkey v2. *Neuron*, 58(3):442–450, May 2008.

[15] D. B. Chklovskii and A. A. Koulakov. Maps in the brain: What can we learn from them? *Annual Review of Neuroscience*, 27:369–392, 2004.

[16] B. Cumming. Stereopsis: how the brain sees depth. *Current Biology*, 7(10):645–647, 1997.

[17] Y. Dan and M. Poo. Spike timing-dependent plasticity: From synapses to perception. *Physiological Review*, 86:1033–1048, 2006.

[18] Russell L De Valois, E William Yund, and Norva Hepler. The orientation and direction selectivity of cells in macaque visual cortex. *Vision research*, 22(5):531–544, 1982.

[19] G.C. DeAngelis. Seeing in three dimensions: the neurophysiology of stereopsis. *Trends in Cognitive Sciences*, 4(3), 2000.

[20] G.C. DeAngelis, R.D. Freeman, and Izumi Ohzawa. Stereoscopic depth discrimination in the visual cortex: neurons ideally suited as disparity detectors. *Science*, 249:1037–1041, 1990.

[21] Gregory C DeAngelis, Izumi Ohzawa, and RD Freeman. Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex. i. general characteristics and postnatal development. *Journal of Neurophysiology*, 69(4):1091–1117, 1993.

[22] U. R. Dhond and J. K. Aggarwal. Structure from stereo - a review. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(6):1489–1510, Nov./Dec. 1989.

[23] Barbara A. Dosher and Zhong-Lin Lu. Perceptual learning reflects external noise filtering and internal noise reduction through channel reweighting. *Proceedings of the National Academy of Sciences of the United States of America*, 95(23):13988–13993, November 1998.

[24] James Drever. Perceptual learning. *Annual Review of Psychology*, 11:131–160, 1960.

[25] M. Fahle and S. Edelman. Long-term learning in vernier acuity: effects of stimulus orientation, range and of feedback. *Vision Res*, 33(3):397–412, February 1993.

[26] D. J. Felleman and D. C. Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, 1:1–47, 1991.

[27] D. J. Felleman and D. C. Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cereb. Cortex*, 1(11):1–47, 1991.

[28] Ione Fine and Robert A. Jacobs. Comparing perceptual learning across tasks: A review. *Journal of Vision*, 2(2), 2002.

[29] A. Fiorentini and N. Berardi. Perceptual learning specific for orientation and spatial frequency. *Nature*, 287:43–44, 1980.

[30] D. J. Fleet, A. D. Jepson, and M. R. M. Jenkin. Phase-based disparity measurement. In *CVGIP: Image Understand.*, volume 53, pages 198–210, 1991.

[31] A. Franz and J. Triesch. Emergence of disparity tuning during the development of vergence eye movements. In *International Conference on Development and Learning*, pages 31–36, 2007.

[32] Tomoki Fukai and Shigeru Tanaka. A simple neural network exhibiting selective activation of neuronal ensembles: from winner-take-all to winners-share-all. *Neural Comput.*, 9(1):77–97, January 1997.

[33] Poggio GF and Fischer B. Binocular interaction and depth sensitivity of striate and prestriate cortex of behaving rhesus monkey. *J. Neurophysiol*, 40:1392–1405, 1977.

[34] E. J. Gibson. Perceptual learning. *Annual Review of Psychology*, 14(1):29–56, 1963.

[35] E. J. Gibson. *Principles of perceptual learning and development*. Lawrence Erlbaum, East Norwalk, CT, US, 1969.

[36] F. Gonzales and Perez R. Neural mechanisms underlying stereoscopic vision. *Progress in Neurobiology*, 55:191–224, 1998.

[37] W. E. L. Grimson. *From Images to Surfaces: A Computational Study of the Human Early Visual System*. MIT Press, 1981.

[38] W. E. L. Grimson and D. Marr. A computer implementation of a theory of human stereo vision. In L. S. Baumann, editor, *Proc. ARPA Image Understanding Workshop*, pages 41–45, 1979.

[39] Christopher G Healey and Amit P Sawant. On the limits of resolution and visual angle in visualization. *ACM Transactions on Applied Perception (TAP)*, 9(4):20, 2012.

[40] G. E. Hinton, S. Osindero, and Y-. W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

[41] Patrik Hoyer and Aapo Hyvrinen. Natural image collection for ica experiments. `http://http://research.ics.aalto.fi/ica/imageica/`, February 2000.

[42] Patrik O Hoyer and Aapo Hyvärinen. Independent component analysis applied to feature extraction from colour and stereo images. *Network: Computation in Neural Systems*, 11(3):191–210, 2000.

[43] D. H. Hubel and T. N. Wiesel. Receptive feilds of single neurons in the cat's striate cortex. *Journal of Physiology*, 148:574–591, 1959.

[44] T. Burwick J. Wiemer and W. Seelen. Self-organizing maps for visual feature representation based on natural binocular stimuli. *Biological Cybernetics*, 82(2):97–110, 2000.

[45] Pamela E. Jeter, Barbara Anne Dosher, and Shiau-Hua Liu. Transfer (vs. specificity) following different amounts of perceptual learning in tasks differing in stimulus orientation and position. *Journal of Vision*, 7(9), 2007.

[46] Z. Ji, J. Weng, and D. Prokhorov. Where-what network 1: "Where" and "What" assist each other through top-down connections. In *Proc. IEEE International Conference on Development and Learning*, Monterey, CA, Aug. 9-12 2008.

[47] B. Julesz. Foundations of cyclopean perception. 1971. University of Chicago Press: Chicago.

[48] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, editors. *Principles of Neural Science*. Appleton & Lange, Norwalk, Connecticut, 3rd edition, 1991.

[49] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, editors. *Principles of Neural Science*. McGraw-Hill, New York, 4th edition, 2000.

[50] A. Karni and D. Sagi. Where practice makes perfect in texture discrimination: evidence for primary visual cortex plasticity. *Proceedings of the National Academy of Sciences of the United States of America*, 88(11):4966–4970, June 1991.

[51] Gerd Kempermann. Why new neurons? possible functions for adult hippocampal neurogenesis. *J. Neurosci.*, 22(3):635–638, February 2002.

[52] J. Kirn, B. O'Loughlin, S. Kasparian, and F. Nottebohm. Cell death and neuronal recruitment in the high vocal center of adult male canaries are temporally related to changes in song. *Proceedings of the National Academy of Sciences of the United States of America*, 91(17):7844–7848, August 1994.

[53] T. Kohonen. *Self-Organizating Maps*. 1997.

[54] Chi-Tat Law and Joshua I. Gold. Neural correlates of perceptual learning in a sensory-motor, but not a sensory, cortical area. *Nature Neuroscience*, 11(4):505–513, March 2008.

[55] Tai S. Lee, Cindy F. Yang, Richard D. Romero, and David Mumford. Neural activity in early visual cortex reflects behavioral experience and higher-order perceptual saliency. *Nature Neuroscience*, 5(6):589–597, May 2002.

[56] Sidney R Lehky and Terrence J Sejnowski. Neural model of stereoacuity and depth interpolation based on a distributed representation of stereo disparity [published erratum appears in j neurosci 1991 mar; 11 (3): following table of contents]. *The Journal of Neuroscience*, 10(7):2281–2299, 1990.

[57] W. Li, V. Piëch, and C. D. Gilbert. Perceptual learning and top-down influences in primary visual cortex. *Nature neuroscience*, 7(6):651–657, June 2004.

[58] J. Lippert, D. J. Fleet, and H. Wagner. Disparity tuning as simulated by a neural net. *Journal of Biocybernetics and Biomedical Engineering*, 83:61–72, 2000.

[59] Jiajuan Liu, Zhong-Lin Lu, and Barbara A. Dosher. Augmented hebbian reweighting: Interactions between feedback and training accuracy in perceptual learning. *Journal of Vision*, 10(10), 2010.

[60] Margaret S Livingstone and David H Hubel. Anatomy and physiology of a color system in the primate visual cortex. *The Journal of neuroscience*, 4(1):309–356, 1984.

[61] H. Lu, T. Luwang, J. Weng, and X. Xue. A multilayer in-place learning network for development of general invariances. *International Journal of Humanoid Robotics*, 4(2), 2007.

[62] Zhong-Lin Lu, Jiajuan Liu, and Barbara A. Dosher. Modeling mechanisms of perceptual learning with augmented hebbian reweighting. *Vision Research*, September 2009.

[63] M. Luciw and J. Weng. Where what network 3: Developmental top-down attention with multiple meaningful foregrounds. In *Proceedings of IJCNN*, 2010. accepted to appear.

[64] M. D. Luciw and J. Weng. Motor initiated expectation through top-down connections as abstract context in a physical world. In *Proc. 7th International Conference on Development and Learning (ICDL'08)*, 2008.

[65] Matthew Luciw and Juyang Weng. In *IEEE World Congress on Computational Intelligence*, pages 4233–4240, Barcelona, Spain, July 2010.

[66] M.D. Luciw and J. Weng. Topographic class grouping with applications to 3d object recognition. In *Proc. International Joint Conf. on Neural Networks*, Hong Kong, June 2008. accepted and to appear.

[67] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Freeman, New York, 1982.

[68] R. Miikkulainen, J. A. Bednar, Y. Choe, and J. Sirosh. *Computational Maps in the Visual Cortex*. Springer, Berlin, 2005.

[69] M. Mishkin, L. G. Unterleider, and K. A. Macko. Object vision and space vision: Two cortical pathways. *Trends in Neuroscicence*, 6:414–417, 1983.

[70] Kajal Miyan and Juyang Weng. Where-what network 3: Developmental top-down attention for multiple foregrounds and complex backgrounds. In *IEEE 9th International Conference on Development and Learning*, pages 280–285, Ann Arbor, MI, August 2010.

[71] T. Bishop P. O. Nikara and J. D. Pettigrew. Analysis of retinal correspondence by studying receptive fields of binocular single units in cat striate cortex. *Exp. Brain Res.*, 6:353–372, 1968.

[72] F. Nottebohm. Neuronal replacement in adult brain. *Brain Res Bull*, 57(6):737–749, April 2002.

[73] National Eye Institute [NEI] of the U.S. National Institute of Health. `http://www.nei.nih.gov/photo` (first visited 04/24/09).

[74] R. C. O'Reilly. Six principles for biologically based computational models of cortical cognition. *Trends in Cognitive Sciences*, 2(11):455–462, November 1998.

[75] A. J. O'Toole and D. J. Kersten. Learning to see random-dot stereograms. *Perception*, 21:227–243, 1992.

[76] A. Parker, A. J. Cumming, and J. Read. A simple model accounts for the reduced response of disparity-tuned v1 neurons to anti-correlated images. *Vis. Neurosci.*, 19:735–753, 2002.

[77] S. Paslaski, C. VanDam, and J. Weng. Modeling dopamine and serotonin systems in a visual recognition network. In *Proc. Int'l Joint Conference on Neural Networks*, pages 3016–3023, San Jose, CA, July 31 - August 5 2011.

[78] Marina Pavlovskaya and Shaul Hochstein. Perceptual learning transfer between hemispheres and tasks for easy and hard feature search conditions. *Journal of Vision*, 11(1):8:1–13, 2011.

[79] Alexander A Petrov, Barbara A. Dosher, and Zhong-Lin Lu. A computational model of perceptual learning through incremental channel re-weighting predicts switch costs in non-stationary contexts. *Journal of Vision*, 3(9), 2003.

[80] Alexander A. Petrov, Barbara Anne A. Dosher, and Zhong-Lin L. Lu. The dynamics of perceptual learning: an incremental reweighting model. *Psychological review*, 112(4):715–743, October 2005.

[81] JD Pettigrew, T Nikara, and PO Bishop. Binocular interaction on single units in cat striate cortex: simultaneous stimulation by single moving slit with receptive fields in correspondence. *Experimental Brain Research*, 6(4):391–410, 1968.

[82] G. F. Poggio, F. Gonzalez, and F. Krause. Stereoscopic mechanisms in monkey visual cortex: Binocular correlation and disparity selectivity. *JNeuSci*, 8:4531–4550, December 1988.

[83] GF Poggio and B Fischer. Binocular interaction and depth sensitivity in striate and prestriate cortex of behaving rhesus monkey. *J Neurophysiol*, 40(6):1392–1405, 1977.

[84] T. Poggio, M. Fahle, and S. Edelman. Fast perceptual learning in visual hyperacuity. *Science*, 256(5059):1018–1021, May 1992.

[85] N. Qian. Binocular disparity and the perception of depth. *Neuron*, 18:359–368, 1997.

[86] R. D. Raizada and S. Grossberg. Towards a theory of the laminar architecture of cerebral cortex: computational clues from the visual system. *Cereb Cortex*, 13(1):100–113, January 2003.

[87] V. S. Ramachandran and O. Braddick. Orientation-specific learning in stereopsis. *Perception*, 2:371–376, 1973.

[88] T. Ramtohul. A self-organizing model of disparity maps in the primary visual cortex. Master's thesis, School of Informatics, University of Edinburgh, 2006.

[89] J. Read. Early computational processing in binocular vision and depth perception. *Progress in Biophysics and Molecular Biology 87*, pages 77—-108, 2005.

[90] Jenny Read and Bruce Cumming. Sensors for impossible stimuli may solve the stereo correspondence problem. *Nat Neurosci*, September 2007.

[91] A. W. Roe, A. J. Parker, R. T. Born, and G. C. DeAngelis. Disparity channels in early vision. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 27(44):11820–11831, October 2007.

[92] P. R. Roelfsema and A. van Ooyen. Attention-gated reinforcement learning of internal representations for classification. *Journal of Neural Computation*, 17:2176–2214, 2005.

[93] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386–408, November 1958.

[94] A. Schoups, R. Vogels, N. Qian, and G. Orban. Practising orientation identification improves orientation coding in v1 neurons. *Nature*, 412(6846):549–553, August 2001.

[95] Nicolas Bredeche Shi, Shi Zhongzhi, and Jean daniel Zucker. Perceptual learning and abstraction in machine learning. *IEEE Transactions on Systems, Man and Cybernetics, Part C*, 36:172–181, 2003.

[96] Y. F. Sit and R. Miikkulainen. Self-organization of hierarchical visual maps with feedback connections. *Neurocomputing*, 69:1309–1312, 2006.

[97] Mojtaba Solgi, Taosheng Liu, and Juyang Weng. A computational developmental model for specificity and transfer in perceptual learning. *Journal of Vision*, 13(1):1–23, January 2013.

[98] Mojtaba Solgi and Juyang Weng. Developmental stereo: Topographic iconic-abstract map from top-down connection. In *Proc. the First of the Symposia Series New developments in Neural Network (NNN'08)*, 2008.

[99] Mojtaba Solgi and Juyang Weng. Developmental stereo: Emergence of disparity preference in models of the visual cortex. *IEEE Transactions on Autonomous Mental Development*, 1(4):238–252, 2010.

[100] Mojtaba Solgi and Juyang Weng. Stereo where-what networks: Unsupervised binocular feature learning. In *Proc. Int'l Joint Conf. Neural Networks*, pages 1–8, Dallas, TX, USA, August 4-9 2013.

[101] Ida J. Stockman. *Movement and Action in Learning and Development: Clinical Implications for Pervasive Developmental Disorders*. Elsevier Academic Press, 2004.

[102] A. F. Teich and N. Qian. Learning and adaptation in a recurrent model of v1 orientation selectivity. *J Neurophysiol*, 89(4):2086–2100, April 2003.

[103] Misha Tsodyks and Charles Gilbert. Neural networks and perceptual learning. *Nature*, 431(7010):775–781, October 2004.

[104] Lucia M. Vaina, V. Sundareswaran, and John G. Harris. Learning to ignore: psychophysics and computational modeling of fast learning of direction in noisy motion stimuli. *Cognitive Brain Research*, 2(3):155–163, July 1995.

[105] Y. Wang, X. Wu, and J. Weng. Synapse maintenance in the where-what network. In *Proc. Int'l Joint Conference on Neural Networks*, pages 2823–2829, San Jose, CA, July 31 - August 5 2011.

[106] Yair Weiss, Shimon Edelman, and Manfred Fahle. Models of perceptual learning in vernier hyperacuity. *Neural Computation*, 5:695–718, 1993.

[107] J. Weng. Image matching using the windowed Fourier phase. *International Journal of Computer Vision*, 11(3):211–236, 1993.

[108] J. Weng. A 5-chunk developmental brain-mind network model for multiple events in complex backgrounds. In *Proc. Int'l Joint Conf. Neural Networks*, pages 1–8, Barcelona, Spain, July 18-23 2010.

[109] J. Weng and M. D. Luciw. Optimal in-place self-organization for cortical development: Limited cells, sparse coding and cortical topography. In *Proc. 5th International Conference on Development and Learning (ICDL'06)*, Bloomington, IN, May 31 - June 3 2006.

[110] J. Weng, T. Luwang, H. Lu, and X. Xue. Multilayer in-place learning networks for modeling functional layers in the laminar cortex. *Neural Networks*, 21:150–159, 2008.

[111] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen. Autonomous mental development by robots and animals. *Science*, 291(5504):599–600, 2001.

[112] J. Weng and N. Zhang. Optimal in-place learning and the lobe component analysis. In *Proc. World Congress on Computational Intelligence*, Vancouver, Canada, July 16-21 2006.

[113] Juyang Weng and Wey-Shinan Hwang. From neural networks to the brain: autonomous mental development. *Computational Intelligence Magazine, IEEE*, 1(3):15–31, 2006.

[114] Juyang Weng and M. Luciw. Dually optimal neuronal layers: Lobe component analysis. *IEEE Transactions on Autonomous Mental Development*, 1(1):68–85, May 2009.

[115] Peter Werth, Stefan Scherer, and Axel Pinz. Subpixel stereo matching by robust estimation of local distortion using gabor filters. In *CAIP '99: Proceedings of the 8th International Conference on Computer Analysis of Images and Patterns*, pages 641–648, London, UK, 1999. Springer-Verlag.

[116] F. A. Wichmann and N. J. Hill. The psychometric function: I. fitting, sampling, and goodness of fit. *Percept Psychophys*, 63(8):1293–1313, November 2001.

[117] A. K. Wiser and E. M. Callaway. Contributions of individual layer 6 pyramidal neurons to local circuitry in macaque primary visual cortex. *Journal of neuroscience*, 16:2724–2739, 1996.

[118] Lu-Qi Xiao, Jun-Yun Zhang, Rui Wang, Stanley Klein, Dennis Levi, and Cong Yu. Complete transfer of perceptual learning across retinal locations enabled by double training. *Current Biology*, 18(24):1922–1926, December 2008.

[119] A. J. Yu and P. Dayan. Uncertainty, neuromodulation, and attention. *Neuron*, 46:681692, 2005.

[120] Cong Yu, Stanley A. Klein, and Dennis M. Levi. Perceptual learning in contrast discrimination and the (minimal) role of context. *Journal of Vision*, 4(3), 2004.

[121] Jun-Yun Zhang, Gong-Liang Zhang, Lu-Qi Xiao, Stanley A. Klein, Dennis M. Levi, and Cong Yu. Rule-based learning explains visual perceptual learning and its specificity and transfer. *The Journal of Neuroscience*, pages 12323–12328, September 2010.

[122] Ting Zhang, Lu-Qi Xiao, Stanley A. Klein, Dennis M. Levi, and Cong Yu. Decoupling location specificity from perceptual learning of orientation discrimination. *Vision Research*, August 2010.

[123] L. Zhaoping, M. Herzog, and P. Dayan. Nonlinear observation and recurrent preprocessing in perceptual learning. *Network*, 14(6873):790–793, February 2003.

[124] L. Zhaoping, M. H. Herzog, and P. Dayan. Quadratic ideal observation and recurrent preprocessing in perceptual learning. *Network: Computation in Neural Systems*, 14:233–247, 2003.

[125] Konstantinos C Zikidis and Athanasios V Vasilakos. Asafes2: A novel, neuro-fuzzy architecture for fuzzy computing, based on functional reasoning. *Fuzzy Sets and Systems*, 83(1):63–84, 1996.

[126] C. L. Zitnick and T. Kanade. A cooperative algorithm for stereo matching and occlusion detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):675–684, Jul. 2000.