Cortex-Inspired Developmental Learning Networks for Stereo Vision

By

Mojtaba Solgi

Comprehensive Examination

Department of Computer Science and Engineering

2012
ABSTRACT

Cortex-Inspired Developmental Learning Networks for Stereo Vision

By

Mojtaba Solgi

Developmental networks, such as Where-What Networks (WN), have been shown promising for simultaneous attention and recognition, while handling variations in scale, location and type as well as inter-class variations [38, 52, 96]. Moreover, in a simplistic setting, they have shown sub-pixel accuracy in disparity detection in challenging natural images [88]. However, the existing work for stereo vision requires training with groundtruth depth maps and is not able to scale to real world problems due to inefficient feature selection. This dissertation will focus 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 regime that requires only the general shape of the objects (e.g., plane, sphere, etc.) rather than a pixel level depth map 3) integration of depth perception with location, type and scale.
# TABLE OF CONTENTS

## 1 Background

1.1 Physiology of binocular vision ........................................ 1
  1.1.1 Eye ........................................ 2
  1.1.2 Visual Pathway ....................................... 2
  1.1.3 Retina ......................................... 3
  1.1.4 LGN ........................................ 4
  1.1.5 Primary Visual Cortex .................................. 4
  1.1.6 Disparity ....................................... 5
  1.1.7 Geometry of Binocular Vision ............................... 5
  1.1.8 Encoding of Binocular Disparity ............................ 7

1.2 Existing Work in Computational Modeling of Binocular Vision .... 7
  1.2.1 Energy Model ...................................... 9
  1.2.2 Wiemer et. al. 2000 .................................. 10
  1.2.3 Works based on LLISOM ................................. 10

## 2 Completed Work

2.1 Transfer of Learning ........................................ 18
  2.1.1 Introduction to Perceptual Learning ........................... 18
  2.1.2 Model .......................................... 21
  2.1.3 Simulation ....................................... 38
  2.1.4 Results ......................................... 44
  2.1.5 Discussion ........................................ 46
  2.1.6 Conclusion ....................................... 48

2.2 Disparity Detection on Natural Images ........................... 49
  2.2.1 Introduction ....................................... 50
  2.2.2 Network Architecture and Operation ........................ 54
  2.2.3 Experiments and Results ................................ 61
  2.2.4 Regression ........................................ 65
  2.2.5 Discussion ....................................... 70
  2.2.6 Conclusions ...................................... 71

## 3 Progress and Proposed Future Work

3.1 Progress ............................................... 73
  3.1.1 Stereo Network for Disparity Map Generation ............... 73
  3.1.2 Dynamic Synapse LCA ................................. 74
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. Furthermore, the details of LCA (Lobe Component Analysis) and MILN (Multilayer In-place Learning Networks) are discussed and compared with other models of visual neural networks. At the end of the chapter, related works on disparity models are presented. Most material on biological visual systems is adapted from Kandel 2000 [41] and Ramtohul 2006 [75], and those about LCA and MILN are largely adapted from Weng & Luciw 2009 [102].

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.
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 pupil’s 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 LGN, at the base of the same side of the brain. LGN in turn does some processing on the signals and projects to the V1 of the opposite 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
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 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 renotopic 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 [72]) such as occlusion, relative size, motion parallax, perspective, shading, blur, and relative motion (DeAngelis 2000 [16], Gonzalez & Perez 1998 [31]). 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 stereopsis 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(fixed) 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 [7], such vertical disparities occur when objects are located 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 $\beta_1$ and $\beta_2$. The vertical disparity, denoted by $v$, is the difference between these two angles, $v = \beta_2 - \beta_1$ [7].
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. 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 [14]. 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. [6]. They discovered that neurons in the cat striate cortex
respond selectively to the objects with different binocular depth. In 1997 Poggio and Fischer [28] did a similar experiment with an awake macaque monkey that confirmed the previous evidence by Barlow et. al. [6]. Since the visual system of these animals to a great extent resembles that of human, researchers believe that there are disparity-selective neurons in the human visual cortex as well. Poggio & Fischer [28] 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.

<table>
<thead>
<tr>
<th>Disparity selective cell type</th>
<th>Placement of stimuli</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuned-excitatory</td>
<td>Stimuli at zero disparity</td>
</tr>
<tr>
<td>Tuned-inhibitory</td>
<td>Stimuli at all disparities except those near zero disparity</td>
</tr>
<tr>
<td>Near</td>
<td>Stimuli at negative disparity</td>
</tr>
<tr>
<td>Far</td>
<td>Stimuli at positive disparity</td>
</tr>
</tbody>
</table>

Table 1.1: Four basic types of disparity selective neurons.

Julesz 1971 [39] 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 [31]. Later experiments revealed the existence of two additional categories, named tuned near and tuned far [70]. Fig. 1.9 depicts the 6 categories identified by Poggio et. al. 1988 [70].

Despite neurophysiological data and thrilling discoveries in binocular vision, a computational model was missing until 1990 when Ohzawa et. al. [63] 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 [77] 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 [63] 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. [60]

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. [77] completed the previous energy model by Ohzawa et. al. [63]. 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. [36] 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 from 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 example of the maps they reported.

### 1.2.3 Works based on LLISOM

Laterally Interconnected Synergetically Self-Organizing Maps by Mikkulainen et. al. [57] 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. [57] 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 [75] 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 [75].
Figure 1.7: Two models of disparity encoding (reprinted from [4])

(a) Position Difference Model

(b) Phase Difference Model
Figure 1.8: An example of random dot stereogram (reprinted from [76])

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 [31])
Figure 1.10: Energy Model by Ohzawa et. al. [63] (reprinted from [63])

Figure 1.11: Modified Energy Model by Read et. al. [77] (reprinted from [77])
Figure 1.12: Pre-processing to create a pool of stimuli by Wimer et al. [36] (reprinted from [36])

Figure 1.13: Self-organized maps of left and right eye receptive fields (reprinted from [36])
Figure 1.14: Schematic of the architecture for basic LLISOM (reprinted from [57])

Figure 1.15: Self-organized orientation map in LLISOM (reprinted from [57])
Figure 1.16: Two eye model for self organization of disparity maps in LLISOM (reprinted from [75])

Figure 1.17: Topographic disparity maps generated by LLISOM (reprinted from [75])
Chapter 2

Completed Work

2.1 Transfer of Learning

The material in this section are adapted from [87]. Please refer to the original paper for details.

2.1.1 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 light on the underlying mechanisms of learning and plasticity. The subject of PL has long attracted researchers interested in behavioral [19, 29], modeling [30, 84, 91] and physiological [46, 83] 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 3, 5, 20, 23, 24, 37, 65, 71, 74, 108). For example, in a well-known study, [83] 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. [42] 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., 67, 106, 109, 110). 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\(^1\). 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., [1, 90, 111]. 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., 18, 51, 71), and perceptual experience improves the readouts from sensory cortex by modifying (reweighting) the connections from low-level representation areas to

\(^1\)In fact, a major goal of machine learning research is to create algorithms which can generalize using as few training examples as possible (no specificity).
high-level decision making areas [50, 68]. 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 [109] 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 [96]. 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 developmental neurobiological studies showing that the brain develops its network in an individual’s life time (see, e.g., 21, 41).

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 [106]). 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.

2.1.2 Model

The overall architecture – Introduction to WWN

Where-What Networks [38] are a visuomotor version of the brain-inspired model outlined in [96], 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 $x$ in the bottom-up source, but instead a joint feature $(x, z)$ consisting of both bottom-up feature vector $x$ from receptors and top-down feature vector $z$ from effectors. In order for such a neuron to win the lateral competition and subsequently 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 [54], stereo vision without explicit feature matching to generate disparity outputs [88] and early language acquisition and language-based generalization [59]. 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 [99], 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 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) [102] 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.

The learning algorithm

The learning algorithm in WWN is inspired by the 6-layer structure of the laminar cortex [10]. 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.

### 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, $w_{b,i}^{(L4)}$, and the bottom-up input to the neuron, $b_i^{(L4)}$, 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.

$$\hat{z}_i^{(L4)} = \frac{b_i^{(L4)}}{||b_i^{(L4)}||} \cdot \frac{w_{b,i}^{(L4)}}{||w_{b,i}^{(L4)}||}$$  \hspace{1cm} (2.1)
Figure 2.3: How training and exposure accompanied by off-task processes can cause the learning effects to transfer. Each circle schematically represents a column of neurons with laminar architecture (see a column’s details in part C of the figure), solid lines show connections made (or improved) via direct perceptual learning, and dashed lines are the connections made or improved via off-task processes. (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. (C) A cortical column from the internal layer magnified, along with its neighboring columns.
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 [27] and [64], 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:

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

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, $\omega$, 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.

**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.

**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:

\[
\hat{z}_i^{(L3)} = \frac{1}{2}(b_i^{(L3)} + e_i^{(L3)})
\]  

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 \( 3 \times 3 \) neighborhood (in the 2D sheet of neuronal columns) are excited. The responses level of the excited columns are set to the response level of their neighboring winner column, multiplied by an exponentially declining function of their distance (in the 2D grid of columns) to the winner columns:

\[
z_i^{(L3)} \leftarrow e^{-\frac{d^2}{2}} z_{\text{winner}}^{(L3)}
\]  

where the distance \( d = 1 \) for immediate neighbors of the winner columns, and \( d = \sqrt{2} \) for the diagonal neighbors in the \( 3 \times 3 \) 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).

**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:

\[
w_{b,i}^{(L4)} \leftarrow \beta_1 w_{b,i}^{(L4)} + \beta_2 z_i^{(L4)} b_i^{(L4)}
\]
where $\beta_1$ and $\beta_2$ determine retention and learning rate of the neuron, respectively:

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

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.

$\mu$ is a monotonically increasing function of $m_i$ that prevents the learning rate $\beta_2$ from converging to zero as $m_i$ increases.

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

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, $b_i^{(L4)}$, 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:

$$
w_{e,i}^{(L2)} \leftarrow \beta_1 w_{e,i}^{(L2)} + \beta_2 z_i^{(L2)} e_i^{(L2)}
$$

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).

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 2.1.2 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 \(n_i\) represents. 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, \( \gamma_i \), 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.,

\[
\gamma_i = \alpha \sum_j e^{-c \Delta t_{j}}
\]

where \( \alpha \) 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 2.1.2. 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., [107]).

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.
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.

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., \( L_1F_1 \) (green) and \( L_2F_2 \) (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 2.1.2.

- **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., \( L_1 \) with \( F_2 \) and \( L_2 \) with \( F_1 \) shown in Fig. 2.3A. In the general case, the exposure can be active or passive engagement with stimuli. Section 2.1.2 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., \( L_1 \) and \( F_2 \)) and internal (feature) columns of neurons (e.g., \( L_1F_2 \)) 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 2.1.2) 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.

**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:

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

where the sign $\Leftarrow$ 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.

**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 2.1.2) 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

$$X(L1F1) \Leftarrow Y(L1F1) \Leftarrow [Z_L(L1), Z_F(F1)]$$  (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

$$X(L2F2) \Rightarrow Y(L2F2) \Rightarrow [Z_L(L2), Z_F(F2)]$$  \hspace{1cm} (2.13)

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

**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:

$$X(L2F1) \Rightarrow Y(L2F1) \Rightarrow [Z_L(L2), Z_F(F1)]$$  \hspace{1cm} (2.14)

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

$$X(L1F2) \Rightarrow Y(L1F2) \Rightarrow [Z_L(L1), Z_F(F2)].$$  \hspace{1cm} (2.15)

**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 2.1.2. 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:

$$Z_L(L1) \rightleftharpoons Y(L1F2).$$  \hspace{1cm} (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:

$$Z_F(F1) \rightleftharpoons Y(L2F1).$$  \hspace{1cm} (2.17)

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

$$Z_L(L2) \rightleftharpoons Y(L2F1) \hspace{1cm} (2.18)$$

from Eq. 2.14 and

$$Z_F(F2) \rightleftharpoons Y(L1F2) \hspace{1cm} (2.19)$$

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.
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 $F_1$ and $L_2$ happen to be spontaneously activated. Fig. 2.4 illustrates the example below, while showing only $5\times5$ neuronal columns in the internal area ($50\times50\times2$ 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 $10 \times 10$ 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 $F_1$, 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 $L_2$ is one and from $L_1$ 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. 2.27, 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 $3 \times 3$
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 2.1.2. 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. Each small square represents a neuron, and each grid represents a layer in the model laminar cortex. Neurons at the same location on each grid belong to the same neuronal column. For each of the Layers 2, 3 and 4 the pattern on the grid on the left shows the pre-reponse of the neuron (activation level before lateral competition), and the grid on the right shows the final response of the neuron. See Section 2.1.2 for a step-by-step description of the neural activation patterns.

An experimental example of this type of transfer is [106] 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
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., $L_2F_1$) 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 $F_1$ at location $L_1$, and its transfer to the second location $L_2$. 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 $L_2F_1$ are the columns that are specifically recruited during the off-task processes.

2.1.3 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 [106]. Similar to the behavioral study by [106], 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. 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: $loc_1$ and $loc_2$. 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 $10 \times 10$ 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 $loc_1$ at $(r, c) = (20, 20)$ or $loc_2$ at $(r, c) = (75, 20)$ in the $100 \times 100$ 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 [106], which used adaptive staircases. This should not affect our results, as our model is based on the Lobe Component Analysis theory [102] 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 106), 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., [88] in a su-
pervised learning setting and 66 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:

**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$^3$. To develop stable representations $10^4$ natural image
patches (randomly selected from larger images) of size $100 \times 100$ were used. Fig. 2.6 shows that
the neuronal columns in this stage develop features that resemble oriented edges and blobs.

$^3$The images used are available from [http://research.ics.tkk.fi/ica/imageica/](http://research.ics.tkk.fi/ica/imageica/)
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.

**Perceptual learning at $\text{loc}_{1,\text{ori}1}$ and $\text{loc}_{2,\text{ori}2}$**

Each of the offsets in range $[-5, +5]$, excluding 0, were trained 100 times at $\text{loc}_{1,\text{ori}1}$ for 5 consecutive sessions. Then the same training procedure was simulated for Vernier stimuli at $\text{loc}_{2,\text{ori}2}$. The training regime followed the double-training procedure employed in [106].

**Off-task processes and transfer**

In the last step of the simulation, new and improved connections between feature neurons at $\text{loc}_1$ and concept neurons for $\text{ori}2$ were formed during off-task processes, due to spontaneous firing of these neurons and the Hebbian rule. Therefore, the skill originally learned for $\text{loc}_{2,\text{ori}2}$ was transferred to $\text{loc}_{1,\text{ori}2}$. In a similar scenario, the skill learned for $\text{loc}_{1,\text{ori}1}$ was transferred to $\text{loc}_{2,\text{ori}1}$. 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 2.1.2. Input to the network was a $100 \times 100$ 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 2.1.2). 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 (rewetting) rule (see Equations 2.5 and 2.8) during training and off-task processes.

2.1.4 Results

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 approached zero (Fig. 2.7), similar to human observers. We fitted the performance data with a Weibull function using psignifit [104] 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.

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, \( \text{loc1_ori1} \) 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 \( \text{loc1_ori1} \) (no transfer). Threshold decreases for \( \text{loc2_ori2} \) as a result of training (green curve). At the end of the 9th training session, threshold for the two untrained conditions \( \text{loc1_ori2} \) and \( \text{loc2_ori1} \) drops to the same level as the trained conditions. (C, D) 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 [106] for comparison.

The training of \( \text{loc2_ori2} \) 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 \( \text{loc1_ori2} \) and \( \text{loc2_ori1} \) 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 improvement in performance, defined as \( 100 \times \frac{T_b-T_a}{T_b} \) 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 [106].

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:

\[ d_{i,j} = (w_{i,j}^{\text{preL}} - w_{i,j}^{\text{postT}})^2 \]  

(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 \( w_{i,j}^{\text{preL}} \) and \( w_{i,j}^{\text{postT}} \) 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 \( d_{\text{sensory}} = 0.0098 \) while the average value for ascending and descending connections between the sensory area and the concept areas was \( d_{\text{rewighting}} = 0.247 \). 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.

### 2.1.5 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.
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, [46] 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 [48] 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., 71) or a two-way cascade of internal areas with unsupervised learning, without top-down supervision [34]. 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 [43, 44, 61]. 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.
**Previous models**

A prominent model in perceptual learning literature is the Hebbian channel reweighting model by [18]. [69] 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 [81], 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., [47, 83]. Moreover, their lack of top-down connections from higher areas to representation areas is inconsistent with overwhelming neuroanatomic evidence, as reviewed by, e.g., [22].

Similar to [69], several previous models which have been tested on different perceptual learning tasks (e.g., [92] on motion perception, [112] on bisection, [94] 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, [2] 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.

### 2.1.6 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 [58], (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.

### 2.2 Disparity Detection on Natural Images

The material in this section are adapted from [88]. Please refer to the original 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.

2.2.1 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 stereopsis 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 2.2.2 Eq. 2.24 the activation level of the neurons from the previous time step is used to supervise $L_2$). 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.

Existing methods for stereo disparity detection fall into three categories:
1. **Explicit matching**: Approaches in this category detect discrete features and explicitly match them across two views. Well-known work in this category include [32], [17] and [113].

2. **Hand-designed filters**: Filters are designed to compute profile-sensitive values (e.g. Gabor filters [103], [78], and phase information [25], [95]) 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 [33].

3. **Network learning models**: These models develop disparity-selective filters (i.e. neurons) from experience, without doing explicit matching, and map the responses to disparity outputs (e.g. [49], [36], [26]).

Categories (1) and (2) employ explicit left and right match through either an explicit search or implicit gradient-based search. They are generally called explicit matching approaches. Category (1) fails to perform well in image regions with weak texture or when a patch of the image is missing in either of left or right images (i.e. occlusion), as it requires searching for the best match using texture cues. Category (2) methods have the potential advantage of detecting other visual information such as edges and shading, which can be used in an integrated visual recognition system. However, this category suffers from inability to adapt to experience - hand-designed filters cannot possibly capture the statistics of any new environment, regardless of how complicated their design is. Methods in Category (3) not only develop filters that integrate other visual information, but they can also adapt to changing visual environment. Moreover, in contrast with category (2), a unified neuromorphic network learns to deal with both feature matching and disparity computation.

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 [56]. Solutions to the correspondence problem have been explored using area-, feature-, pixel- and phase-based, as well as Bayesian approaches [17]. 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.
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) [15]. SOM type of networks that use both top-down and bottom-up inputs has not be studied until recently [80, 85, 97, 98]. 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 [55]. Further, we experimentally show that such a disparity based internal topographic grouping leads to improved disparity classification.

Neurophysiological studies (e.g. [11] and [10]) 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. 2.11. However, a computational model that explains how this laminar architecture contributes to classification and regression was unknown. LAMINART [73] 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 [35] reported performance of the laminar cortical architecture for classification and recognition, and Weng et. al. 2008 [98] reported the performance advantages of the laminar architecture (paired layers) over a uniform neural area. Franz & Triesch 2007 [26] 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 [86], 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 [86].

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 [21] and that for the intra-cortical connections was informed by the work of Callaway [9] and Wiser & Callaway [105] as well as others.

Luciw & Weng 2008 [53] 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 [35, 86, 98].

It has been known that orientation preference usually changes smoothly along the cortex [8]. Chen et. al. [12] 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

53
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 [101]. The learning method used in this work falls in Type 2 of the classification proposed in [101], 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. [89]). 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. 2.12). 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 the paper, 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.

2.2.2 Network Architecture and Operation

The networks applied in this paper are extensions of the previous models of Multilayer In-place Learning Network (MILN) [98]. To comply with the principles of Autonomous Mental Development (AMD) [99], these networks autonomously develop features of the presented input, and no
hand-designed feature detection is needed.

To investigate the effects of supervisory top-down projections, temporal context, and 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. 2.9.

In this particular study, we deal with networks consisting of a sensory array (marked as Input in Fig. 2.9), 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.

**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, $z^{(M)}$ 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 $\hat{z}^{(L1)}_i$ – called pre-response, linearly from the bottom-up part and top-down part

$$\hat{z}^{(L1)}_i(t) = (1 - \alpha) \cdot \frac{b^{(L1)}(t) \cdot w^{(L1)}_{b,i}(t)}{\|b^{(L1)}(t)\| \|w^{(L1)}_{b,i}(t)\|} + \alpha \cdot \frac{z^{(M)}(t) \cdot w^{(L1)}_{e,i}(t)}{\|z^{(M)}(t)\| \|w^{(L1)}_{e,i}(t)\|}$$

(2.21)

where $t$ denotes time, $w^{(L1)}_{b,i}(t)$ and $w^{(L1)}_{e,i}(t)$ are this neuron’s bottom-up and top-down weight vectors, respectively, $b^{(L1)}$ is the bottom-up input vector to Layer 1, and $z^{(M)}(t)$ is the firing rates of motor cortex neurons (supervised during training, and not active during testing). The relative top-down coefficient $\alpha$ 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 $r_i = \text{rank}(\hat{z}^{(L1)}_i(t))$ is the ranking of the pre-response of the $i$’th neuron (with the highest active neuron ranked as 0), we have $z^{(L1)}_i(t) = s(r_i)\hat{z}^{(L1)}_i(t)$, where

$$s(r_i) = \begin{cases} \frac{k-r_i}{k} & \text{if } 0 \leq r_i < k \\ 0 & \text{if } r_i \geq k \end{cases}$$

(2.22)

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
as a winner, but also its $3 \times 3$ 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 $3 \times 3$ neighbors are allowed to fire and update their synapses. We use an updating technique called lobe component analysis [100]. 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. 2.21 does not have a top-down part. The response $z^{(M)}$ is computed in the same way otherwise, with its own parameter $k$ controlling the number of non-inhibited neurons.

6-layer Cortical Architecture

The architecture of the feature-detection cortex of the 6-layer architecture is sketched in Fig. 2.11. We use no hand-designed feature detector (e.g. Laplacian of Gaussian, Gabor filters, etc.), as it would be against the paradigm of AMD [99]. The five layers in the stereo feature detection cortex are matched in functional-assistant pairs (referred as feedforward-feedback pairs in [10]). $L3$ 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 [10]). $L6$ 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 constant. 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:

\[
\begin{align*}
    z_j^{(M)}(t) &= \begin{cases} 
        1 - \frac{d(i,j)}{\kappa} & \text{if } d(i,j) < \kappa \\
        0 & \text{if } d(i,j) \geq \kappa 
    \end{cases} 
\end{align*}
\]  \hspace{1cm} (2.23)

where \( M \) denotes Motor Cortex, \( d(i,j) \) is the distance between neuron \( n_i \) and neuron \( n_j \) in the neural plane, and \( \kappa \) 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 \( L_2 \) neurons via top-down connections.

\[
e^{(L_2)}(t) = z^{(M)}(t - 1)
\]  \hspace{1cm} (2.24)

2. **Pre-response in \( L_4 \) and \( L_2 \)** – Neurons in \( L_4(L_2) \) 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. 2.21, except \( L_4 \) only has bottom-up and \( L_2 \) only has top-down.

\[
\hat{z}_i^{(L_4)}(t) = \frac{b^{(L_4)}(t) \cdot w^{(L_4)}_{b,i}(t)}{\|b^{(L_4)}(t)\| \|w^{(L_4)}_{b,i}(t)\|}
\]  \hspace{1cm} (2.25)

and

\[
\hat{z}_i^{(L_2)}(t) = \frac{e^{(L_2)}(t) \cdot w^{(L_2)}_{e,i}(t)}{\|e^{(L_2)}(t)\| \|w^{(L_2)}_{e,i}(t)\|}
\]  \hspace{1cm} (2.26)

3. **\( L_6 \) and \( L_5 \) provide modulatory signals to \( L_4, L_2 \) and \( L_3 \)** – \( L_6 \) and \( L_5 \) receive the firing pattern of \( L_4, L_2 \) and \( L_3 \), 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 \( (L_2, L_3 \text{ and } L_4) \), assistant layers \( (L_5 \text{ and } L_6) \) 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 $L4$ [10].

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 input 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.

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

(2.27)

where $\alpha$ is the relative top-down coefficient. We will discuss the effect of this parameter in detail in Section 2.2.4.
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. $b_i^{(M)}(t) = z^{(L3)}(t)$). Then, they compute their response using the same equation as in Eq. 2.25, and laterally compete. The response of the winner neurons is scaled using the same algorithm as in Eq. 2.22 (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:

$$disparity = \frac{\sum_{n_i \text{ is winner}} d_i \times z_i^{(M)}(t)}{\sum_{n_i \text{ is winner}} z_i^{(M)}(t)} \tag{2.28}$$
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 $3 \times 3$ short-range lateral excitatory connections) update their bottom-up connection weights. Lobe component analysis (LCA) [100] 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.

### 2.2.3 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.

**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/imageica/). 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. 2.10 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. 2.12.
The Effect of Top-Down Projection  As we see in Fig. 2.13, 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 2.13: 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.

**Topographic Class Maps** As we see in Fig. 2.14, 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 [55]. Since the input to a neuron in feature-detection layer has two parts, the iconic input $x_b$ and the abstract (e.g. class) input $x_t$, 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 2.14: The class probability of the $40 \times 40$ 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:

$$H = \sum_{i=1}^{N} -p(n, C_i) \log(p(n, C_i))$$  \hspace{1cm} (2.29)

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

$$p(n, C_i) = \frac{f(n, C_i)}{\sum_{j=0}^{m} f(n, C_j)}$$  \hspace{1cm} (2.30)

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. 2.15 shows that the topographic representation enabled by the top-down projections generalizes better and increases the neurons’ purity significantly during training and testing.
2.2.4 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 $40 \times 40$ neurons in each of $L_2$, $L_3$ and $L_4$ layers of the stereo feature-detection cortex. correspondence between input and $L_1$ 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. 2.23 and $\alpha = 0.4$ in Eq. 2.27 for all of the experiments, unless otherwise is stated.

The Advantage of Spatio-temporal 6-layer Architecture  Fig. 2.16 shows that applying top-down context signals in single-layer architecture (traditional MILN networks [98]), increases the error rate up to over 5 pixels (we intentionally set the relative top-down coefficient, $\alpha$, as low as
0.15 in this case, otherwise the error rate would be around chance level). This observation is due to 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. $L_5$ and $L_6$) in the laminar cortex to modulate the top-down and bottom-up energies received at the cortex before mixing them.

![Effect of utilizing laminar architecture and temporal context](image)

Figure 2.16: 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 [45]. 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 $\alpha$. Different networks were trained with more than 40 thousand random training samples (as opposed to training with epochs). Fig. 2.17 shows the effect of context parameter, $\alpha$, 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 ($\alpha$ in our model) is tuned by evolution.
Smoothly Changing Receptive Fields  In two separate experiments, we studied the topographic maps formed in $L_3$.

Experiment A – $\kappa = 5$  As depicted in Fig. 2.18a, 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. 2.23) during training. Fig. 2.18b 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. 2.18b the colors change smoothly from dark blue to red).

Experiment B – $\kappa = 1$  However, if we define disparity detection as a classification problem, and set $\kappa = 1$ in Eq. 2.23 (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 [13] and disparity maps in par-
ticular [12, 79].
Figure 2.18: (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 [12] (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. It can be seen that these maps resemble those from the neurophysiological study presented in (a). (c,e). Disparity-probability maps in $L3$ where $\kappa = 5$ in (c) and $\kappa = 1$ (e). For each neuron, the largest disparity-probability (the disparity for which the neuron is most probable to fire) is shown by the color corresponding to that particular disparity. (d,f). Cross-correlation of disparity-probability where $\kappa = 5$ in (d) and $\kappa = 1$ in (f). Higher value of cross-correlation means higher similarity between two vectors, and hence more probable that neurons fire together for the corresponding classes.
2.2.5 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 has two functional roles:

1. Rank and scale the top-down energy received at the cortex (modulated by signals from $L_5$) in $L_2$

2. Integrate the modulated bottom-up energy received from $L_4$ to the modulated top-down energy received from higher cortical areas to determine the output signals of the cortex in $L_3$

Neuroscientists have known for a long time that there are sublayers in the laminar cortex [40]. 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 ($L_2/3$) to two separate tasks. This is different from the previous models (e.g. [9]), as they consider $L_2/3$ as one functional layer.

Fig. 2.19 illustrates the result of an experiment in which we compared two models of $L_2/3$. In the traditional model of $L_2/3$, it is modeled as one functional layer that integrates the sparse coded signals received from $L_4$ with the top-down energy. While in our novel model used in this paper, $L_2/3$ functions as 2 functional layers, namely $L_2$ and $L_3$ (see Prediction 1).

---

*Marked as Level2, layers 2 through 4B in [9] Figure 2.*
2.2.6 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 [98] 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 $L^2$ and $L^3$ to $L^4$ 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 ($L^{2/3}$) in laminar cortex carries out more functionality than what has been proposed in previous models - it provides sparse representation for top-down stimuli in $L^2$, combines the top-down and bottom-up sparse representations in $L^3$, 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 3

Progress and Proposed Future Work

3.1 Progress

3.1.1 Stereo Network for Disparity Map Generation

The objective of conventional stereo vision in the computer vision research is to build disparity maps from which depth of every pixel can be inferred (depth map). To this end, many benchmark datasets have been composed, and the performance of stereo algorithms are compared on these datasets. Perhaps the most well-known such dataset is Middlebury Stereo Dataset [82]. In this project, I applied the the WWN network on the Middlebury dataset. Fig. 3.1D shows that with $40 \times 40 = 1.6K$ neurons in the internal layer and 16 disparity levels, a rough disparity map is obtained. Increasing the number of neurons to $100 \times 100 = 10K$ and using 24 disparity levels will give a better depth map as in Fig. 3.1E. The size of the bottom-up receptive-field of the internal neurons is $20 \times 1$ pixels in either left and right images. Despite the partial success of the network for this setting, it seems that it is not scalable to a general purpose solution and suffers from problems such as the horizontal scratch effect. In the rest of this dissertation, we will focus on how to improve this method.
3.1.2 Dynamic Synapse LCA

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. The goal of this part of the project is to use inspirations from neurophysiology to modify the LCA algorithm so it dynamically retract irrelevant (e.g., background) synapses and grow relevant (foreground) synapses.

Fig. 3.2 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 $F_b$. As it is shown in Section 2.2, using the original LCA algorithm will not be able to achieve this since it does not differentiate between foreground and background dimensions of the input vector.

Figure 3.2: Demonstration of different parts of input to a binocular neuron. SRF: Sensory receptive field, $B_m$: background monocular, $B_b$: background binocular, $F_m$: foreground monocular, $F_b$: foreground binocular. An efficient binocular neuron should pick up only the binocular part of the foreground, $F_b$.

Wang et. al [93] 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.

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

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

**Algorithm 1 Dynamic Synapse LCA**

```
for time steps \( t \) of the network running do
  for neurons \( n \) in the internal \( Y \) area do
    Perform LCA competition and updating using synapse age
    if \( n \) is allowed to fire then
      for \( i \)’th synapse in neuron \( n \) do
        Incrementally update \( \sigma_i \) according to Eq. 3.1
        Compute \( \bar{\sigma}(n_i) \) according to Eq. 3.3
      for \( i \)’th synapse in neuron \( n, n_i \) do
        Compute the synaptogenic factor according to Eq. 3.4
        if \( \sigma_i > \xi_2 \bar{\sigma} \) and synapse.age > \( n_c \) then \( \triangleright \) the cutting age is set to \( n_c = 100 \)
        Cut the synapse \( n_i \)
        else if \( \sigma_i < \xi_1 \bar{\sigma} \) and synapse.age > \( n_g \) then \( \triangleright \) the growing age is set to \( n_g = 20 \)
        Grow synapses in the immediate neighborhood of \( n_i \)
        else if \( \xi_1 \bar{\sigma} \leq \sigma_i \leq \xi_2 \bar{\sigma} \) then
          \( v_i \leftarrow f(\sigma_i, \bar{\sigma})v_i \)
```

\[
\sigma_i(n) = \begin{cases} 
  0 & \text{if } n \leq n_0 \\
  w_1(n)\sigma_i(n) + w_2(n)|v_i - p_i| & \text{otherwise}
\end{cases}
\] (3.1)

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:

\[
\mu(n) = \begin{cases} 
  0, & \text{if } n < t_1 \\
  c(n - t_1)/(t_2 - t_1), & \text{if } t_1 < n < t_2 \\
  c + (t - t_2)/r, & \text{if } n > t_2
\end{cases}
\] (3.2)

where \( t_1 = 10, t_2 = 10^3, c = 2 \) and \( r = 10^4 \).

\[
\bar{\sigma}(n) = \frac{1}{d} \sum_{i=1}^{d} \sigma_i(n).
\] (3.3)
where $d$ is the number of dimensions in the input vector.

$$f(\sigma_i, \bar{\sigma}) = \begin{cases} \frac{1}{\sigma_i + \epsilon} - \frac{1}{\xi_2 \bar{\sigma} + \epsilon} & \text{if } \sigma_i < \xi_1 \bar{\sigma} \\ \frac{1}{\sigma_i + \epsilon} & \text{if } \xi_1 \bar{\sigma} \leq \sigma_i \leq \xi_2 \bar{\sigma} \\ 0 & \text{if } \sigma_i > \xi_2 \bar{\sigma} \end{cases}$$ (3.4)

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

**Synapse age versus neuron age**

The original LCA algorithm (Eq. 3.7) uses the number of times a neuron has fired, the neuronal age, to calculate the retention and learning rates (Eq. 3.8).

$$v_j(t) = \omega_1 v_j(t - 1) + \omega_2 y_j x(t), \quad (3.5)$$

$$\omega_1 = 1 - \omega_2, \quad \omega_2 = \frac{1 + \mu(n_j)}{n_j}, \quad (3.6)$$

Since in the Dynamic Synapse extension synapses are allowed grow and retract all the time, a global neuronal age applies only to the “oldest” neurons and the other synapses should use the frequency of the neuron firing after they were grown as their age. Hence, Eq. 3.7 and 3.8 change to the following for Dynamic Synapse LCA:

$$v_{ij}(t) = \omega_1 v_{ij}(t - 1) + \omega_2 y_j x_j(t), \quad (3.7)$$

$$\omega_1 = 1 - \omega_2, \quad \omega_2 = \frac{1 + \mu(n_{ij})}{n_{ij}}, \quad (3.8)$$

where $n_{ij}$ is the “age” of the $i$’th synapse of the $j$’th neuron.

**Initial results**

To test the ability of the proposed network to dynamically retract and grow synapses to cover the locally relevant parts of the input the following experiment was carried out.

A $30 \times 30$ pixel image was randomly selected from natural images dataset as background.
Then a $16 \times 8$ pixels ellipsoidal foreground was also randomly selected from natural images and was superimposed on the left and right background images at different disparities. Fig. 3.3 shows that only a part of synapses in the original circular receptive field ($t = 0$) will survive while some are retracted 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 $Y$ area of the network.

![Figure 3.3](image.png)

Figure 3.3: Demonstration of dynamic synapse growth and retraction in the receptive field of a sample neuron. Each pair shows the left and right input images with the ellipsoidal foreground. $t$ is the time index in the simulation. The bright green color shows the border of the receptive field of the neuron in either of the images at each time step. The transparent red pixels show the synapses from the initial circular receptive fields, while the transparent green pixels show the synapses expanded during development.

### 3.1.3 Network for Shape from Stereo

In this part of the project, we utilize the developed DSLCA algorithm to create a developmental stereo network. Instead of training the network with a ground truth depth map which is developmentally implausible and expensive to produce, we aim to train the network only with high level shape and depth information, e.g., the surface type (plane or sphere) and the object orientation. Fig. 3.4 shows the setup of the experiment. An object is placed in front of random natural image, attached to a planar board. Two cameras feed left and right images to the developmental network which is trained for location of the object, orientation of the plane and type of the object.
3.2 Timeline

A timeline of proposed work is as follows.


- January-February, 2013 — Finish shape from stereo work. Publish the results and analysis.


3.3 Novelty of Completed Thesis

The novelty of the completed Dissertation is summarized in this section.
• Dynamic Synapse Lobe Component Analysis with both synapse retraction and synapse growth is novel.

• The use of local receptive-fields in a developmental network for stereo is novel.

• Learning of disparity-tuned binocular neurons without explicitly teaching the disparity values

• Integration of multiple depth cues, e.g., binocular disparity, perspective, shading, etc. in a single network for depth perception

• Simultaneous shape, type and location perception


