Developmental Stereo: Emergence of Disparity Preference in Models of the Visual Cortex

Mojtaba Solgi and Juyang Weng, *Fellow, IEEE*

***Abstract*—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 subpixel average error in regression, and 0.90 success rate in classification, given limited resources.**

***Index Terms*—Binocular vision, neuromorphic modeling, spatiotemporal, six-layer laminar cortical architecture.**

I. 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. Furthermore, the existing engineering methods for binocular matching are not only computationally expensive, but also hard to integrate other visual cues to help the perception of depth. It is important to look at the problem from a different angle—How the brain solves the problem of binocular vision? In particular, what are the computational mechanisms that regulate the development of the visual nervous system, and what are the role of gene-regulated cortical architecture and spatiotemporal aspects of such mechanisms?

Although steropsis seems to be a spatial problem, the temporal information appears to help stereopsis due to the physical continuity underlying the physicality of dynamics. Biological agents exploit spatial and temporal continuity of the visual stimuli to enhance their visual perception. In the real world, objects do not come into and disappear from the field of view randomly, but rather, they typically move continuously across the field of view, given their motion is not too fast for the brain to respond. At the pixel level, however, values are very discontinuous as image patches sweep across the field of view. Our model assumes that visual stimuli are largely spatially continuous. Motivated by the cerebral cortex, it utilizes the temporal context in the later cortical areas, including the intermediate areas and motor output area, to guide the development of earlier areas, [In Section II-B, (4) the activation level of the neurons from the *previous time step* is used to supervise $L2$.] These later areas are more *"abstract"* than the pixel level, and thus provide needed information as temporal context. However, how to use such emergent information is a great challenge.

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 [8], [13], and [39].
2) **Hand-designed filters:** Filters are designed to compute profile-sensitive values (e.g., Gabor filters [24] and [37], and phase information [10] and [30]) 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 [14].
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., [11], [16], and [19]).

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 potentail 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 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.

Manuscript received April 30, 2009; revised October 05, 2009. First published December 11, 2009; current version published February 05, 2010.

The authors are with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824 USA (e-mail: solgi@cse.msu.edu; weng@cse.msu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAMD.2009.2038360

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 [22]. Solutions to the correspondence problem have been explored using area-based, feature-based, pixel-based, and phase-based as well as Bayesian approaches [8]. 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 [10].

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) [7]. SOM type of networks that use both top-down and bottom-up inputs has not be studied until recently [26], [27], [31], [33]. 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 [21]. Further, we experimentally show that such a disparity based internal topographic grouping leads to improved disparity classification.

Neurophysiological studies (e.g., [12] and [3]) have shown that the primary visual cortex in macaque monkeys and cats has a laminar structure with a local circuitry similar to our model in Fig. 3. However, a computational model that explains how this laminar architecture contributes to classification and regression was unknown. LAMINART [23] 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 [15] reported performance of the laminar cortical architecture for classification and recognition, and Weng *et al.* 2008 [33] reported the performance advantages of the laminar architecture (paired layers) over a uniform neural area. Franz and Triesch 2007 [11] 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 and Weng 2008 [28], 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 subpixel precision, in contrast with the prior scheme of classification in Solgi and Weng 2008 [28].

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 subpixel precision. The model of the intercortical connections we present here was informed by the work of Felleman and Van Essen [9], and that for the intracortical connections was informed by the work of Callaway [2] and Wiser and Callaway [38], as well as others.

Luciw and Weng 2008 [20], 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 [15], [28], [33].

It has been known that orientation preference usually changes smoothly along the cortex [1]. Chen *et al.* [4] 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 subpixel precision among the network learning models for stereo. Applying the novelties mentioned in 1) and 2) showed surprisingly drastic accuracy differences in performance. 4) The first study of smoothly changing disparity sensitivity map; 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 eight categories [36]. The learning method used in this work falls in Type 2 of the classification proposed in [36], 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., [29]). 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. 9). 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.

II. Network Architecture and Operation

The networks applied in this paper are extensions of the previous models of multilayer in-place learning network (MILN) [33]. To comply with the principles of AMD [34], 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: single-layer architecture for classification and 6-layer architecture for regression. An overall sketch of the networks is illustrated in Fig. 1. In this particular study, we deal with networks consisting of a sensory array (marked as *Input* in Fig. 1), a stereo feature-detection cortex, which may be a single layer of neurons or have a 6-layer architecture inspired by the laminar architecture of human cortex, and a motor cortex that functions as a regressor or a classifier.

A. Single-Layer Architecture

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

1) **Fetching input in Layer 1 and imposing supervision signals (if any) in motor cortex**: When the network is being trained, $\mathbf{z}^{(M)}$ is imposed originating from outside (e.g., by a teacher). In a classification problem, there are c motor cortex neurons and c possible disparity classes. The true class being viewed is known by the teacher, who communicates this to the system. Through an internal process, the firing rate of the neuron corresponding to the true class is set to one, and all others set to zero.

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

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

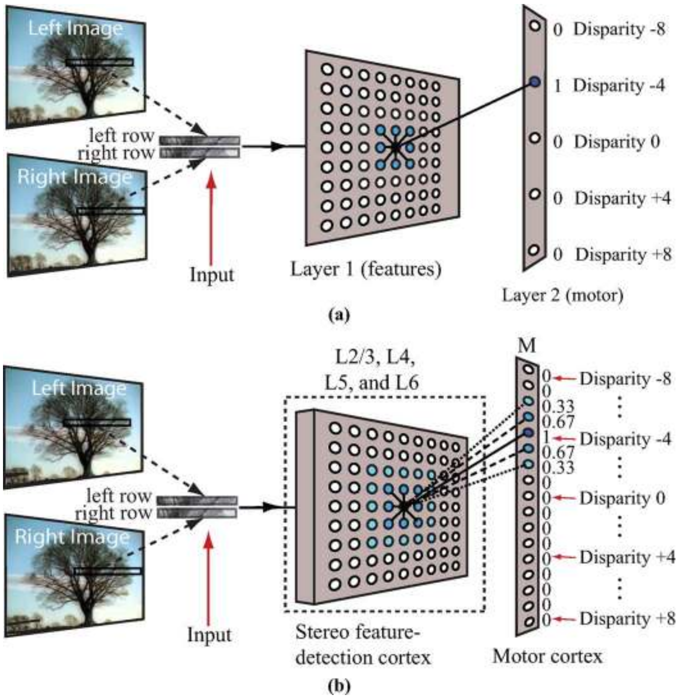

Fig. 1. (a). The binocular network single-layer architecture for classification. (b). The binocular network 6-layer architecture for regression. Input to the networks (on the left) consists of a pair of rows taken from slightly different positions (depending on the degree of disparity) of a set of natural images. Two image patches are extracted from the same image position in the left and right image planes. Feature-detection cortex neurons self-organize from bottom-up and top-down signals. Each motor neuron is marked by the disparity it is representative for (ranging from -8 to $+8$). Each circle is a neuron. Activation level of the neurons is shown by the darkness of the circles: the higher the activation, the darker the neurons are depicted. The diagram shows an instance of the network during training phase when the disparity of the presented input is -4. In (a) the stereo feature-detection cortex is a single layer of Lobe Component Analysis (LCA) [35] neurons. A rectangular kernel sets the activation of only Disparity -4 neuron to 1 and all the others to 0. In (b), the stereo feature-detection cortex has a 6-layer laminar architecture (see Fig. 3). A triangular kernel, centered at the neuron of Disparity -4, imposes the activation level of Disparity -4 neuron and four of its neighbors to positive values and all the others to 0. The lines between neurons in the motor cortex and feature detection cortex represent two-way synaptic connections. The denser the line, the stronger the connection. (a) Single-layer architecture. (b) 6-layer architecture.

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

3) **Competition Via Lateral Inhibition**: A neuron's preresponse is used for intralevel competition. k neurons with the highest preresponse win, and the others are inhibited. If $r_i = \mathrm{rank}(\hat{z}_i^{(L1)}(t))$ is the ranking of the preresponse of the i'th neuron (with the highest active neuron ranked as 0), we have $z_i^{(L1)}(t) = s(r_i)\hat{z}_i^{(L1)}(t)$, where

$$\mathrm{s}(r_i) = \begin{cases} \frac{k-r_i}{k} & \text{if } 0 \le r_i < k \\ 0 & \text{if } r_i \ge k \end{cases}. \qquad (2)$$

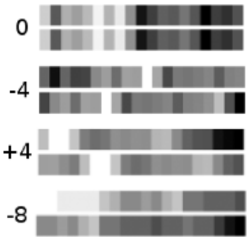

Fig. 2. Examples of input, which consists of two rows of 20 pixels each. The top row is from the left view and the bottom row is from the right view. The numbers on the left side of the bars exhibit the amount of shift/disparity.

4) **Smoothing Via Lateral Excitation**: Lateral excitation means that when a neuron fires, the nearby neurons in its local area are more likely to fire. This leads to a smoother representational map. The topographic map can be realized by not only considering a nonzero-responding neuron i as a winner, but also its 3×3 neighbors, which are the neurons with the shortest distances from i (less than two).
5) **Hebbian Updating With LCA**: After inhibition, the top-winner neuron and its 3×3 neighbors are allowed to fire and update their synapses. We use an updating technique called lobe component analysis [35]. 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 (1) does not have a top-down part. The response $\mathbf{z}^{(M)}$ is computed in the same way otherwise, with its own parameter k controlling the number of noninhibited neurons.

B. 6-Layer Cortical Architecture

The architecture of the feature-detection cortex of the 6-layer architecture is sketched in Fig. 3. We use no hand-designed feature detector (e.g., Laplacian of Gaussian, Gabor filters, etc.), as it would be against the paradigm of AMD [34]. The five layers in the stereo feature detection cortex are matched in functional-assistant pairs (referred as feedforward-feedback pairs in [3]). $L6$ 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.

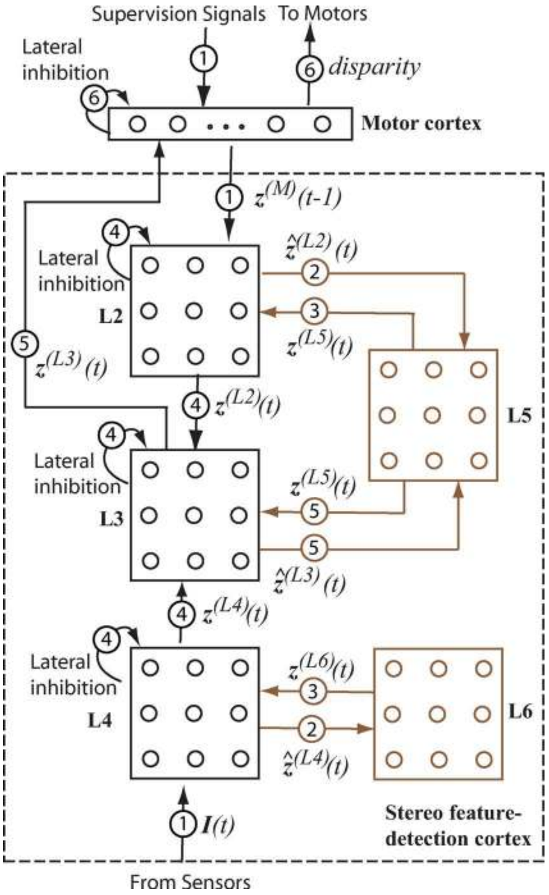

Fig. 3. Architecture diagram of the 6-layer laminar cortex studied in this paper, which also introduces some notation. The numbers in circles are the steps of the algorithm described in Section II. See the text for notations. Parts depicted in brown (gray in black and white copies) are not implemented in our computer simulation.

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

$$z_j^{(M)}(t) = \begin{cases} 1 - \frac{d(i,j)}{\kappa} & \text{if } d(i,j) < \kappa \\ 0 & \text{if } d(i,j) \geq \kappa \end{cases} \tag{3}$$

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

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

$$\mathbf{e}^{(L2)}(t) = \mathbf{z}^{(M)}(t-1). \tag{4}$$

Preresponse in $L4$ **and** $L2$: Neurons in $L4(L2)$ compute their *preresponse* (response prior to competition) solely based on their bottom-up(top-down) input. They use the same equation as in (1), except $L4$ only has bottom-up and $L2$ only has top-down

$$\hat{z}_i^{(L4)}(t) = \frac{\mathbf{b}^{(L4)}(t) \cdot \mathbf{w}_{b,i}^{(L4)}(t)}{||\mathbf{b}^{(L4)}(t)||||\mathbf{w}_{b,i}^{(L4)}(t)||} \tag{5}$$

and

$$\hat{z}_i^{(L2)}(t) = \frac{\mathbf{e}^{(L2)}(t) \cdot \mathbf{w}_{e,i}^{(L2)}(t)}{\|\mathbf{e}^{(L2)}(t)\| \|\mathbf{w}_{e,i}^{(L2)}(t)\|} \tag{6}$$

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

Since the LCA algorithm already incorporates the regulatory mechanisms (i.e., lateral inhibition and excitation) in the functional layers ($L2$, $L3$, and $L4$), assistant layers ($L5$ and $L6$) do not have "actual" neurons in our implementation. They are modeled only to explain the important role of $L5$ and $L6$ in the cortical architecture: providing signals to regulate lateral interactions in $L2$, $L3$, and $L4$[3].

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 $L5$

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

$$z_i^{(L3)}(t) = (1-\alpha) \cdot b_i^{(L3)}(t) + \alpha \cdot e_i^{(L3)}(t) \tag{7}$$

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

6a. Response of Motor Neurons in Testing: The activation level of the motor neurons is not imposed during testing, rather it is computed utilizing the output of feature-detection cortex, and used as context information in the next time step. The neurons take their input from $L3$ [i.e., $\mathbf{b}_i^{(M)}(t) = \mathbf{z}^{(L3)}(t)$]. Then, they compute their response using the same equation as in (5), and laterally compete. The response of the winner neurons is scaled using the same algorithm as in (2) (with a different k for the motor layer), and the response of the rest of the neurons will be suppressed to zero. The output of the motor layer is the response weighted average of the disparity of the winner neurons

$$\text{disparity} = \frac{\sum\limits_{n_i \text{iswinner}} d_i \times z_i^{(M)}(t)}{\sum\limits_{n_i \text{iswinner}} z_i^{(M)}(t)} \tag{8}$$

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

6b. Hebbian Updating with LCA in Training: The top winner neurons in $L4$ and motor cortex and also their neighbors in neural plane (excited by 3 × 3 short-range lateral excitatory connections) update their bottom-up connection weights. Lobe component analysis (LCA) [35] 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.

III. Analysis

A. Elongated Input Fields Using Top-Down

The neighborhood of the input space to which a neuron n_i is tuned (the neuron wins given input from that neighborhood) is called the spatial input field[1] of that neuron, denoted by $\Omega_i \subset \mathbb{R}^n$. We assume that for each neuron n_i the subspace Ω_i has a uniform distribution[2] along any direction (axis) d with mean value $\mu_{i,d}$ and standard deviation $\sigma_{i,d}$. The d'th element of the input vector $\mathbf{x}$ is denoted by x_d.

Proposition 1: The higher the variation of data along a direction in the input field of a neuron, the less is the contribution of that direction of input to the neuron's chance to win in lateral competition.

According to the principles of LCA learning [32], after development each neuron n_i is tuned to the mean value of its input field[3], $\mu_{i,d}$, along any direction d. Therefore, the average deviation of input from the neurons tuned weight is $\sigma_{i,d}$ for any direction d. It is evident that the larger this deviation $\sigma_{i,d}$ is, the less it is statistically probable that the input *matches* with the neuron's tuned weight along that direction, which in turn implies that the less is the contribution of x_d on the neuron's final chance to win in lateral competition with other neurons in the same layer.

Proposition 2: Top-down connections help neurons develop input fields with higher variation along the irrelevant dimensions of input (elongated input fields).

Given uniform distribution in input data, the neurons always develop in such a way that input space is divided equally among their input fields, in a manner similar to Voronoi diagrams. In other words, they develop *semi-isomorphic* input fields. Therefore, we expect that

$$\sigma_{i,d_1} = \sigma_{i,d_2} \tag{9}$$

for any neuron n_i, and directions d_1 and d_2 along the uniform distribution manifold. However, when the neurons develop using top-down input, the projection of their input field on the

[1] a plot of the relationship between position in the input field and neural response [6]. It is also referred to as *input field profile*.

[2] which is a reasonable assumption given the data is patches from natural images.

[3] from now on, wherever we refer to "input field" we mean "input field profile" or equivalently "spatial input field."

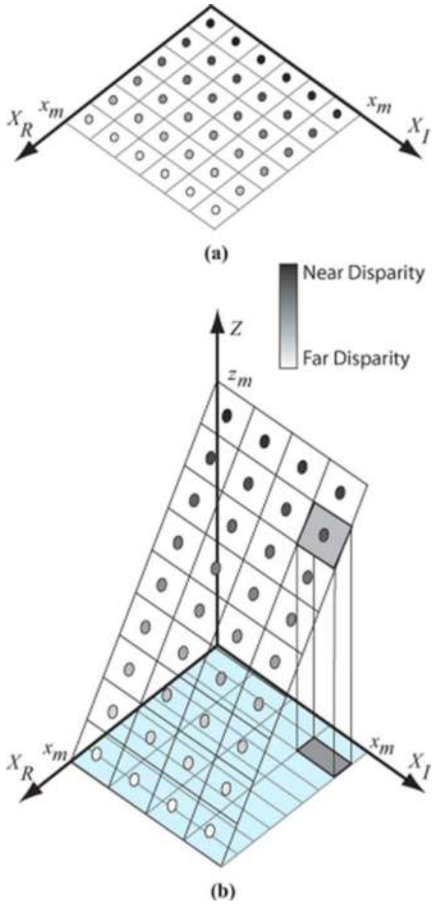

Fig. 4. Each circle represents a neuron, and the shade of circles represents the degree of disparity the neuron is tuned to. The areas shown around neurons are the input fields of neurons. (a) The quantization of input space by neurons without top-down input. The input fields of neurons has the same amount of variation in either of directions relevant and irrelevant input (shown as a square for the sake of visualization simplicity, should be Voronoi diagrams). (b) The quantization of input space by neurons with top-down input. For simplicity we assume the there is a linear relation between relevant part of bottom-up input, X_R, and the top-down input, Z. The input fields of the neurons are still isomorphic (shown as squares) on the input manifold. However, the projection of the input fields on the bottom-up space is no longer isomorphic, but elongated along the irrelevant axis, X_I.

bottom-up input space is not isomorphic anymore. Instead, the bottom-up input field of the neuron is *elongated* along the direction of irrelevant input (See Fig. 4). Assuming linear dependence of Z on X_R in Fig. 4), we have

$$\sigma_{i,d_{ir}} = \lambda\beta\sigma_{i,d_{rel}} \qquad (10)$$

where d_{ir} d_{rel} respectively represents any irrelevant and relevant dimensions of the bottom-up input, and β and λ are constants. According to the triangle similarity (see Fig. 4), when we project the input space onto bottom-up space, the constant λ is a function of the ratio of the range of top-down input, z_m, to the bottom-up input, x_m

$$\lambda = \frac{\sqrt{x_m^2 + z_m^2}}{x_m} = \sqrt{1 + \left(\frac{z_m}{x_m}\right)^2} \qquad (11)$$

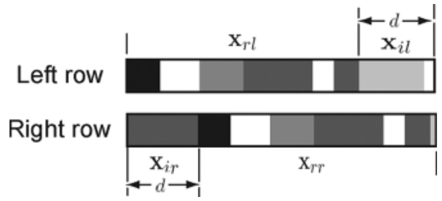

Fig. 5. Top-down connections enable neurons to pick up relevant receptive fields. If a neuron is supervised by the top-down connections to detect a particular disparity d, the irrelevant subspace includes those areas where object images do not overlap, i.e., $\mathbf{x}_{il}$ and $\mathbf{x}_{ir}$. The first subindex indicates whether it is the irrelevant or relevant part of the input space (i and r respectively), and the second subindex shows whether it is from the left view or right view (l and r respectively).

where any element of the bottom-up input vector, x_d, is confined to $x_d \in (0, x_m)$, and any element of the top-down input vector, z_d, is confined to $z_d \in (0, z_m)$ for any direction d. Hence,

$$\lambda > 1 \qquad (12)$$

[4]. The value of β is a function of relative top-down coefficient, α, in (1), and also the ratio of the number of relevant and irrelevant dimensions in input. In the settings we used in this paper, an estimation of β is as follows

$$\beta \simeq \alpha \frac{dim(\mathbf{x})}{dim(\mathbf{z})} = 0.4 \times \frac{32}{8} = 1.6 \qquad (13)$$

where $dim(\mathbf{x})$ and $dim(\mathbf{z})$ are the average[5] number of dimensions (number of elements) in the bottom-up and top-down input vectors. Therefore, the following inequality always holds

$$\beta > 1. \qquad (14)$$

Equations (10), (12), and (14) together imply that

$$\sigma_{i,d_{ir}} > \sigma_{i,d_{rel}} \qquad (15)$$

which is the variation of input fields of the neurons is higher along the irrelevant dimensions, and the reasoning is complete.

Combining Proposition 1 and Proposition 2, we conclude that:

Theorem 1: As a result of top-down connections, neurons autonomously develop input fields in which they are relatively less sensitive to irrelevant parts of the input.

B. Top-Down Connections Help Recruit Neurons More Efficiently

According to the rules of in-place learning [31], neurons don't know whether their inputs are from bottom-up or top-down, neither do they know where they are in the cortical architecture. Each neuron can be thought as an autonomous agent that learns on its own without the help of any controlling mechanism from

[4] $\lambda = \sqrt{2}$ given $z_m = x_m$.

[5] dimensions change according to degree of disparity (see Fig. 5).

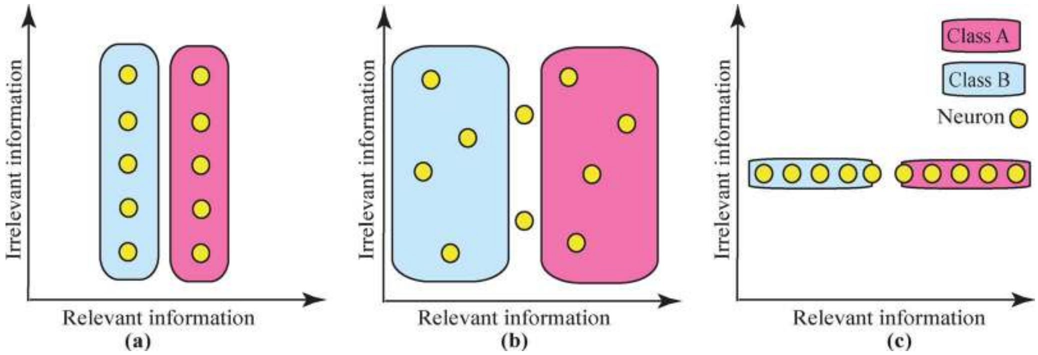

Fig. 6. The deviation of samples along any direction in the input space recruits neurons along this direction. (a) The subspace of relevant information has smaller variance than the irrelevant information. Neurons spread more along the direction of irrelevant subspace. In other words, more neurons characterize the values in the irrelevant space (e.g., 5 neurons per unit distance versus 2 per unit distance). (b) Scale the relevant input by a factor of 2, increasing the standard deviation by a factor of two. Then, neurons spread in both direction with similar densities. (c) Further scale down the irrelevant input, enabling neurons to spread exclusively along the relevant direction (i.e., invariant to irrelevant direction).

outside. Adding top-down connections to a neuron increases its input dimensionality from X to $X \times Z$ where

$$U = X \times Z = \{(x, z) | x \in X, z \in Z\} \tag{16}$$

where × is the *Cartesian product* operator meaning that the new space $X \times Z$ includes inputs from both bottom-up and top-down input spaces. X and Z are respectively bottom-up and top-down input spaces, defined as the following

$$X = \{x = \mathbf{b}_i | \mathbf{b}_i \text{ is the bottom-up input of any neuron } n_i\} \tag{17}$$

$$Z = \{z = \mathbf{e}_i | \mathbf{e}_i \text{ is the top-down input of any neuron } n_i\}. \tag{18}$$

In general, bottom-up input space X of each neuron is composed of the relevant subspace R, the part of input space that is relevant to the motor output, and irrelevant subspace I, the part of input space that is not relevant to the the motor output

$$X = R \times I. \tag{19}$$

It is evident that the top-down input from the space Z is relevant to the output. Thus, we write

$$U = X \times Z = (I \times R) \times Z = I \times (R \times Z) \tag{20}$$

representing that when top-down input is present the new relevant subspace consists of both subspaces R and Z. Besides, the top-down inputs are relatively very variant compared to bottom-up input, since during supervision each value is set to either zero or a nonzero value. Therefore, the following property holds the following.

1) Property 1: Adding top-down signals to a neuron increases the dimensionality and variance of its relevant input subspace.

Furthermore, the following property is true given any distribution of input.

2) Property 2: Neurons are more recruited along the direction of higher variation in input space.

A rigorous mathematical proof of this property is beyond the scope of this paper, however, an intuitive illustration is given in Fig. 6.

Combining Properties 1 and 2, we conclude that:

3) Property 3: Adding top-down connections to neurons results in the recruitment of the neurons more along the direction of relevant input subspace and hence improves the performance of the network.

Even if the top-down signals are not available during testing (in case we don't use context signals during testing), they have already helped neurons tune along the direction of relevant input subspace.

To sum up, we argued that the top-down signals help improve the network performance by increasing the variance of the input space along the direction of relevant input space.

C. Why Use 6-Layer Architecture?

In this section, we analytically investigate why and how the 6-layer laminar architecture outperforms the single-layer architecture model. Fig. 7 compares the algorithms by which the activation level of the neurons in single-layer and 6-layers architectures is computed. In single-layer architecture (the top row in Fig. 7), the top-down and bottom-up energies are first computed and proportionally added according to (21)

$$z_i = (1 - \alpha) \cdot E_{b,i} + \alpha \cdot E_{e,i} \tag{21}$$

$$E_{b,i} = \frac{\mathbf{b}_i \cdot \mathbf{w}_{b,i}}{\|\mathbf{b}_i\| \|\mathbf{w}_{b,i}\|}, E_{e,i} = \frac{\mathbf{e}_i \cdot \mathbf{w}_{e,i}}{\|\mathbf{e}_i\| \|\mathbf{w}_{e,i}\|}. \tag{22}$$

The notation here is consistent with those in (5), (6), and (7)[6]. In most real world sensory data, such as stereo pairs in our case, the bottom-up sensory vector [$\mathbf{b}_i$ in (22)] is significantly more

[6] Except we dropped the time and layer ID components, for the sake of simplicity.

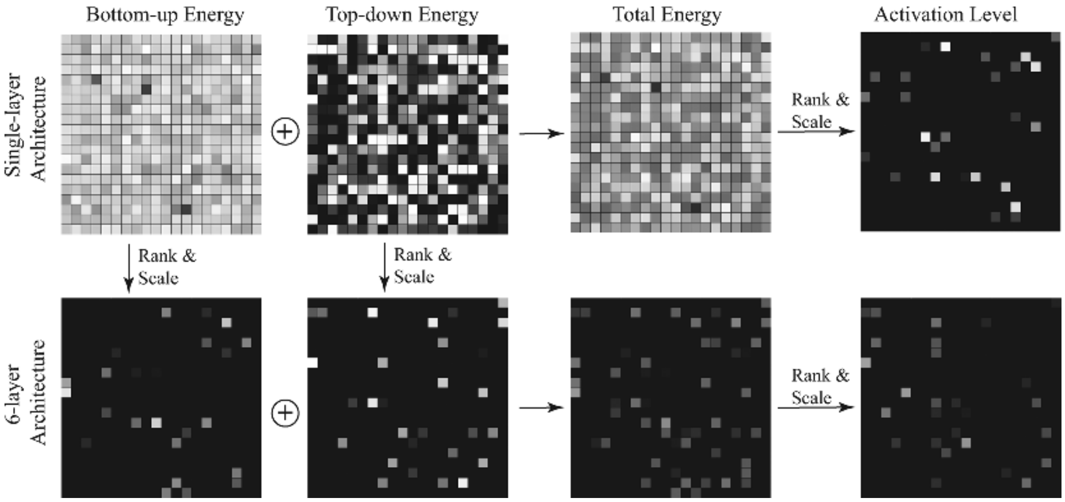

Fig. 7. The mechanisms of neuron winner selection (via lateral inhibition) in single-layer and 6-layer architectures. The maps are taken from a snap-shot of the 20 × 20 neurons in the networks performing on real data. Each small square projects the value for a neuron in that particular position [black(white): minimum(maximum) values]. The top row shows the steps in the single-layer architecture, and the bottom row shows the steps for the 6-layer architecture (which shares some steps with the single-layer architecture). ⊕ represents the operation of taking weighted average of two vectors [similar to (7)].

uniform than the top-down supervision/context vector[7]. In the case of binocular disparity detection, the input pair of images is often featureless with similar intensities for the majority of pixels, while the top-down context/supervision vector is relatively more variant. As a result we have

$$var(E_b) \ll var(E_e) \tag{23}$$

where E_b and E_e are two random variables that can get any of the values $E_{b,i}$ and $E_{e,i}$, respectively. Here, we show that as a result of the lack of variation in bottom-up stimuli in such a single-layer architecture, activation level of the feature detection neurons is mostly determined by only top-down energy and the bottom-up energy is almost discarded. Obviously, this greatly reduces the performance of the network, as the top-down context signals are misleading when the input to the network at time t is considerably different from the input at time $t-1$. We call this effect " *hallucination*".

Let us define $\tilde{\mathbf{E}}_b = \mathbf{E}_b - \bar{\mathbf{E}}_b\mathbf{I}$ where $\bar{\mathbf{E}}_b$ is the mean value of the elements in $\mathbf{E}_b$ (scalar value) and $\mathbf{I}$ is the unit matrix of the same size as $\mathbf{E}_b$. Also, $\tilde{\mathbf{E}}_e = \mathbf{E}_e - \bar{\mathbf{E}}_e\mathbf{I}$ in the same manner, and $\tilde{\mathbf{z}} = (1-\alpha)\cdot\tilde{\mathbf{E}}_b + \alpha\cdot\tilde{\mathbf{E}}_e$. Since $\tilde{\mathbf{z}}$ is only a constant term different from $\mathbf{z}$, we have

$$\mathrm{rank}(z_i) = \mathrm{rank}(\tilde{z}_i) \tag{24}$$

which is, the rank of each element z_i in $\mathbf{z}$, $\mathrm{rank}(z_i)$, is the same as the rank of the corresponding element $\tilde{z}_i$ in $\tilde{\mathbf{z}}$, $\mathrm{rank}(\tilde{z}_i)$. In addition, the rank of each element $\tilde{z}_i = (1-\alpha)\cdot\tilde{E}_{b,i} + \alpha\cdot\tilde{E}_{e,i}$ is mostly determined by its top-down component, $\tilde{E}_{e,i}$. The reason is because (23) induces the absolute value of the top-down component for most of the neurons is much greater than the absolute value of the bottom-up component, i.e., $|\tilde{E}_{e,i}| \gg |\tilde{E}_{b,i}|$. Hence, the ranking of neurons' activation is largely effected only by their top-down component, and the reasoning is complete.

On the other hand, in the case of 6-layer architecture (the bottom row in Fig. 7), the bottom-up and top-down energies are ranked separately in $L4$ and $L2$, respectively, before they get mixed and compete again to decide the winner neurons in $L3$. Therefore, as a result of separation of bottom-up and top-down energies in different laminar layers, the 6-layer architecture manages to out-perform the single-layer architecture, specially when the imperfect context top-down signals are active (as opposed to supervision top-down signals which are always perfect).

D. Recovery From Hallucination

Fig. 8 is an intuitive illustration of how ranking top-down and bottom-up energy separately, as done in the 6-layer laminar architecture, will lead to recovery from a *hallucination* state, while the single layer architecture cannot recover. This analysis is consistent with the results presented in Fig. 13.

In Fig. 8, the input space of neurons is shown on the two axes; top-down input is represented by the horizontal axis, and bottom-up input is represented by the vertical axis. The input signals to the networks are depicted in filled curves along the axes. Distribution of the two classes A and B are shown in rounded rectangles which are wider along the direction of the top-down input since, as discussed earlier in Section III-C, top-down input is more variant than the bottom-up which results in recruitment of neurons more along the top-down direction according to Property 2. The two classes are shown to be linearly separable[8] along the direction of top-down input, but not along the bottom-up input, because top-down signals are always relevant during training. We assume that only top 2 neurons fire (e.g., $k = 2$).

[7] Variance of the elements of the bottom-up sensory vector [$\mathbf{b}_i$ in (22)] is significantly lower than variance of the elements of the top-down supervision/context vector [$\mathbf{e}_i$ in (22)].

[8] Shown linearly separable only for the sake of illustration simplicity in the figures.

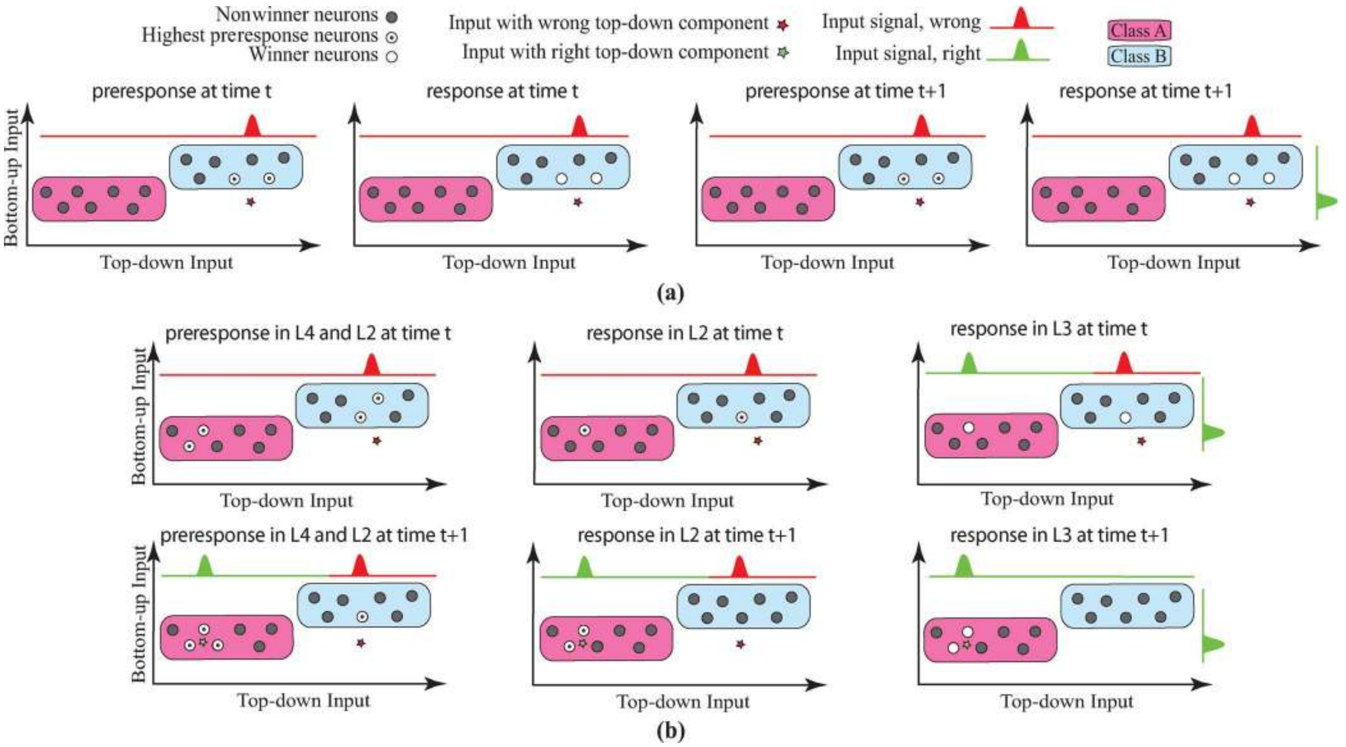

Fig. 8. Schematic illustration of how 6-layer architecture, as opposed to single-layer architecture, makes recovery possible. A sample from class A is given to the network during testing (after the network is developed) while the context top-down signals are related to class B [wrong top-down signals depicted in red (darker) in the figure] . This causes the input to the neurons to be considered as a malicious (wrong) input [denoted by red (darker) stars] and lie out of the input distributions. This figure illustrates the state of the networks after receiving such an input. (a) Single-layer architecture. At time t, two closest neurons to the input have the highest preresponses ($k = 2$). They win and fire. The winner neurons cause the top-down context input to slightly change/adapt to their top-down values. However, this change is not beneficial as the top-down component is still wrong. Therefore, at time $t + 1$ the input will still be classified as class B, which is wrong. (b) In a 6-layer architecture, neurons in $L4$ compete for bottom-up energy and two vertically closest neurons to the input have the highest preresponse and win. In the same manner, two horizontally closest neurons to the input in $L2$ have the highest preresponse and win. Then when the preresponse of neurons in $L3$ is computed it is very probable that some neurons from the correct class A have high preresponses and win in the next step (1st row of (b) far right graph). As a result, top-down input will have a *right* component as well. Because of this right component of the top-down signal, at the next time step $t+1$, the network receives a right input [shown by light star in the 2nd row of (b) far left graph] besides the wrong input. Therefore, we see that one of the final winner neurons is in the correct class A. At the next time step $t + 1$ the network recovers to the state where the top-down signals are right again. (a) Single-layer architecture. (b) 6-layer architecture.

In a single-layer architecture [Fig. 8(a))], given an input with wrong top-down component of class B while the input actually belongs to class A (e.g., when context is unrelated to the bottom-up input), the network will be trapped in a hallucination state, because the high variation of the top-down signal leaves a very small chance for the input to lie close to neurons in class A. Fig. 8(a) illustrates that having a similar bottom-up input at time $t + 1$ (according to spatial continuity of the input) will not change the situation.

On the other hand, in a 6-layer architecture, the neurons compete for top-down energy (in $L2$) and bottom-up energy (in $L4$) separately. In the first row, far left plot of Fig. 8(b) two neurons in class B have high preresponses because of the wrong (misleading) top-down input, and two other neurons in class A have high preresponses because of the right (correct) bottom-up input. As a result, there is a high chance that there are winners among the class A neurons. As the new sample comes in at time $t + 1$ (with the same or very similar bottom-up component due to spatial continuity of input), it is expected that only neurons in the correct class A win as both their bottom-up and top-down component are closer to the input. Finally the network *recovers* in the far right plot in Fig. 8(b) as both the winner neurons are from the correct class A, and the top-down input will be right from then on.

IV. 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.

A. 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 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. 9.

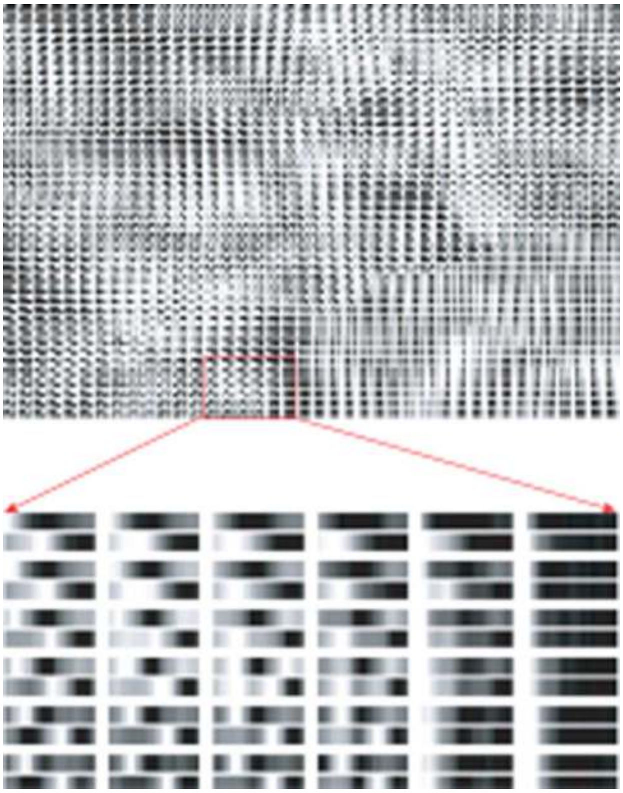

Fig. 9. Bottom-up weights of 40 × 40 neurons in feature-detection cortex using top-down connections. Connections of each neurons are depicted in 2 rows of each 20 pixels wide. The top row shows the weight of connections to the left image, and the bottom row shows the weight of connections to the right image.

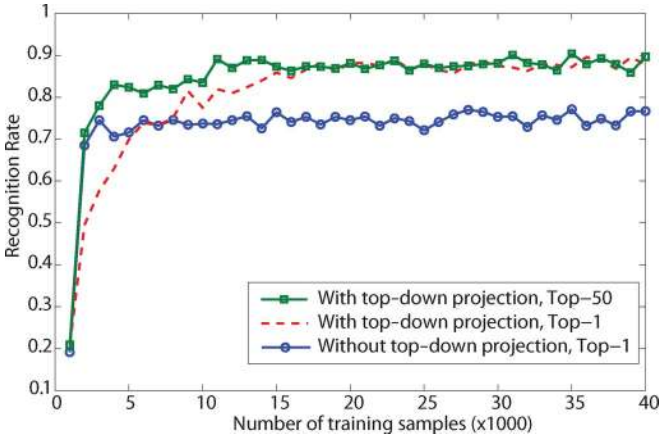

Fig. 10. The recognition rate versus the number of training samples. The performance of the network was tested with 1000 testing inputs after each block of 1000 training samples.

1) The Effect of Top-Down Projection: As we see in Fig. 10, 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.

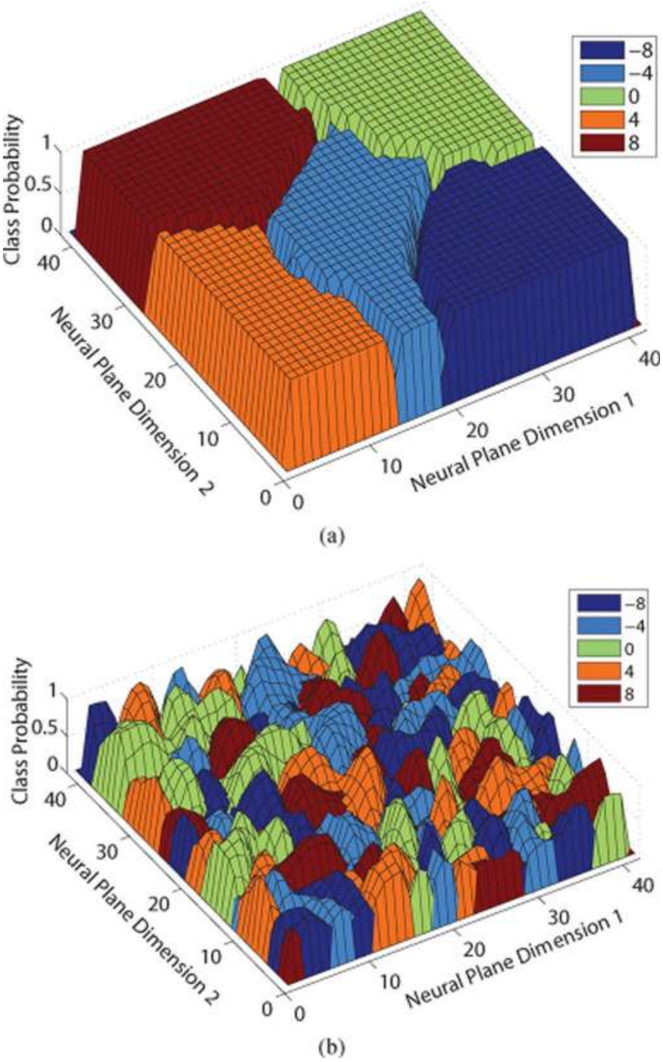

Fig. 11. The class probability of the 40 × 40 neurons of the feature-detection cortex. (a) Top-down connections are active ($\alpha = 0.5$) during development. (b) Top-down connections are not active ($\alpha = 0$) during development.

2) Topographic Class Maps: As we see in Fig. 11, 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 [21]. Since the input to a neuron in feature-detection layer has two parts, the iconic input $\mathbf{x}_b$ and the abstract (e.g., class) input $\mathbf{x}_t$, the resulting internal representation in feature-detection layer is *iconic-abstract*. It is grossly organized by class regions, but within region it is organized by iconic input information. However, these two types of information are not isolated – they are considered jointly by neuronal self-organization.

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

$$H = \sum_{i=1}^{N} -p(n, C_i) \log(p(n, C_i)) \tag{25}$$

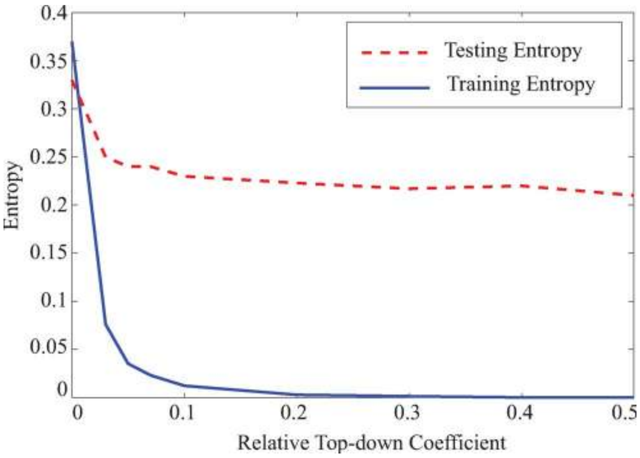

Fig. 12. The effect of top-down projection on the purity of the neurons and the performance of the network. Increasing α in (1) results in purer neurons and better performance.

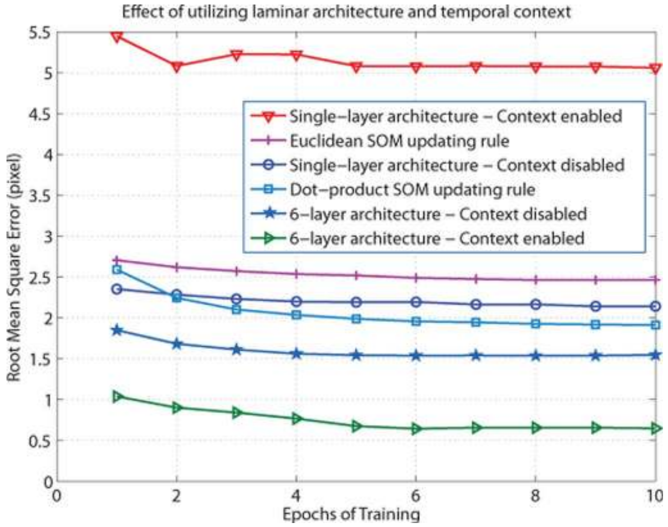

Fig. 13. How temporal context signals and 6-layer architecture improve the performance.

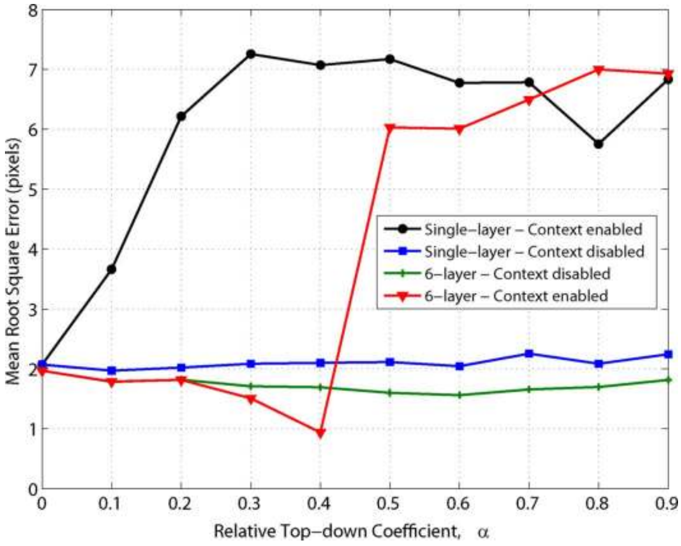

Fig. 14. The effect of relative top-down coefficient, α, on performance in disjoint recognition test on randomly selected training data.

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

$$p(n, C_i) = \frac{f(n, C_i)}{\sum_{j=0}^{m} f(n, C_j)} \tag{26}$$

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

B. 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 bspatially continuous samples were taken from each of the 5 training images. Therefore, there was $5 \times 50 \times 17 = 4250$ training samples in each epoch. For testing, 100 spatially continuous samples were taken from each of the 2 testing images (disjoint test), resulting in $2 \times 100 \times 17 = 3400$ testing samples in each epoch.

We trained networks with 40×40 neurons in each of $L2$, $L3$ and $L4$ layers of the stereo feature-detection cortex. 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 (3) and $\alpha = 0.4$ in (7) for all of the experiments, unless otherwise is stated.

1) The Advantage of Spatio-Temporal 6-Layer Architecture: Fig. 13 shows that applying top-down context signals in single-layer architecture (traditional MILN networks [33]), increases the error rate up to over 5 pixels (we intentionally set the relative top-down coefficient, α, as low as 0.15 in this case, otherwise the error rate would be around chance level). As discussed in Section III, this observation is due the absolute dominance of misleading top-down context signals provided complex input (natural images in this study). On the other hand, context signals reduce the error rate of the network to a subpixel level in 6-layer architecture networks. This result shows the important role of assistant layers (i.e., $L5$ and $L6$) in the laminar cortex to modulate the top-down and bottom-up energies received at the cortex before mixing them.

For comparison, we implemented two versions of SOM updating rules, Euclidean SOM, and dot-product SOM [18]. With the same amount of resources, the 6-layer architecture outperformed both versions of SOM by as much as at least 3 times lower error rate.

In another experiment, we studied the effect of relative top-down coefficient α. Different networks were trained with more than 40 thousand random training samples (as opposed to training with epochs). Fig. 14 shows the effect of context parameter, α, in disjoint testing. It can be seen that the root

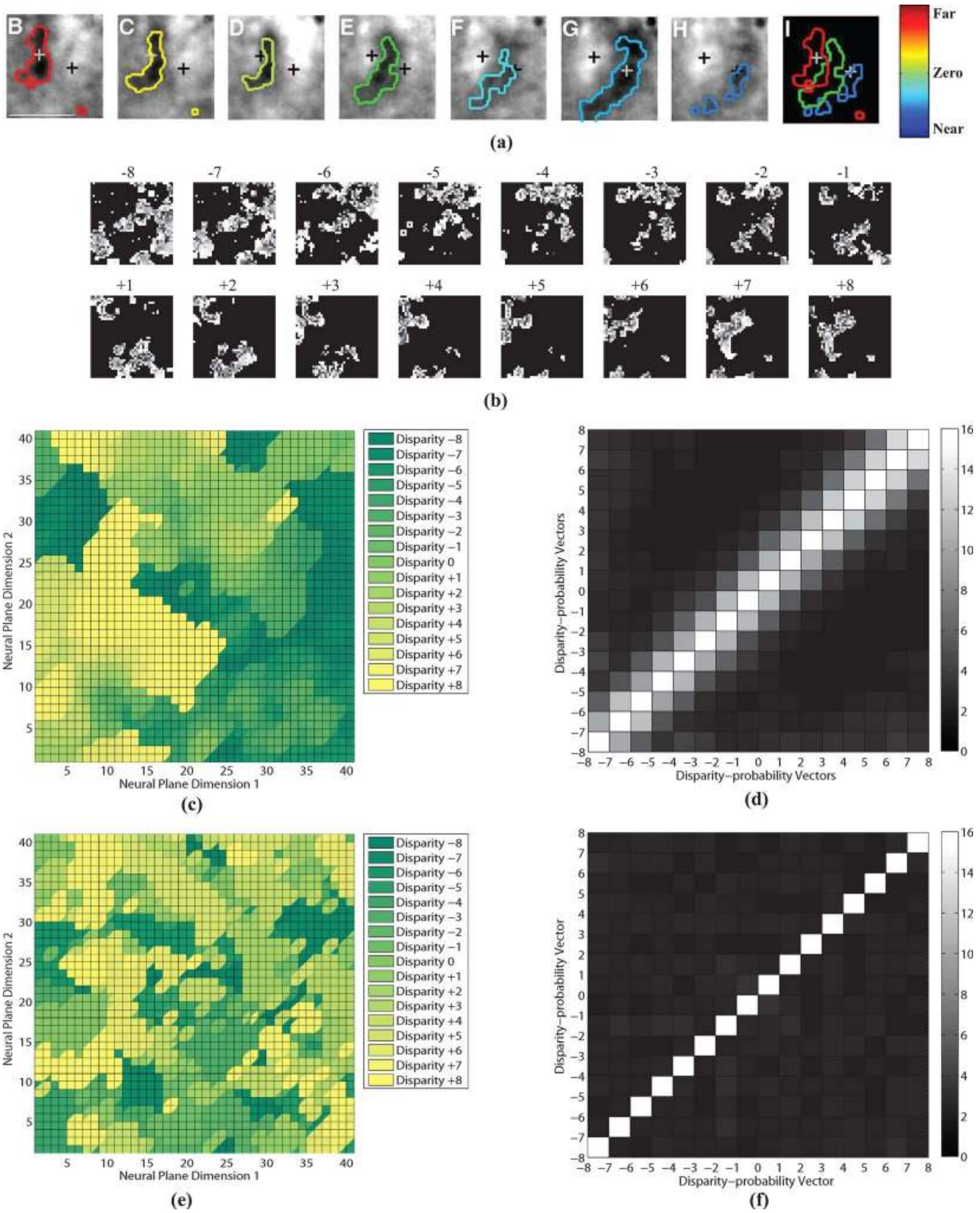

Fig. 15. (a) Map of neurons in $V2$ of macaque monkeys evoked by stimuli with seven different disparities. The position of the two crosses are constant through all the images marked as (B)-(H). Adapted from Chen *et al.* 2008 [4] (b) Disparity-probability vectors of $L3$ neurons for different disparities 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.

mean square error of disparity detection reaches to around 0.7 pixels when $\alpha = 0.4$. We believe that in natural visual systems, the ratio of contribution of top-down temporal signals (α in our model) is tuned by evolution.

2) Smoothly Changing Receptive Fields: In two separate experiments, we studied the topographic maps formed in $L3$.

Experiment A – $\kappa = 5$: As depicted in Fig. 15(a), 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 (3)] during training. Fig. 15(b) 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. 15(b) 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 (3) (only one neuron active in motor layer), then there is no smoothness in the change of the disparity sensitivity of neurons in the neural plane.

These observations are consistent with recent physiological discoveries about the smooth change of stimuli preference in topographic maps in the brain [5] and disparity maps in particular [4], [25].

V. 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 the following.

1) Prediction 1: What is known as Layer 2/3 in cortical laminar architecture[9] has two functional roles.

1) Rank and scale the top-down energy received at the cortex (modulated by signals from $L5$) in $L2$.
2) Integrate the modulated bottom-up energy received from $L4$ to the modulated top-down energy received from higher cortical areas to determine the output signals of the cortex in $L3$.

Neuroscientists have known for a long time that there are sublayers in the laminar cortex [17]. However, the functionality of these sublayers has not been modeled before. This is a step towards understanding the sublayer architecture of the laminar cortex. Our prediction breaks down the functionality of Layer 2/3 ($L2/3$) to two separate tasks. This is different from the previous models (e.g., [2]), as they consider $L2/3$ as one functional layer.

Fig. 16 illustrates the result of an experiment in which we compared two models of $L2/3$. In the traditional model of $L2/3$, it is modeled as one functional layer that integrates the sparse coded signals received from $L4$ with the top-down energy. While in our novel model used in this paper, $L2/3$ functions as 2 functional layers, namely $L2$ and $L3$ (see Prediction 1).

VI. 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.

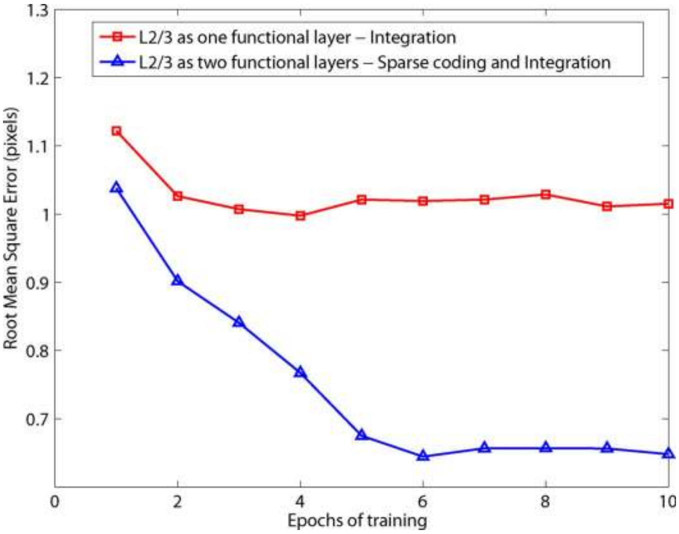

Fig. 16. Comparison of our novel model of $L2/3$ where it performs both sparse coding and integration of top-down and bottom-up signals, with traditional models in which it only does integration.

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 [33] 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 subpixel accuracy. In contrast with engineering methods that used explicit matching between the left and right search windows, a remarkable computational advantage of our work is the potential for integrated use of a variety of image information for tasks that require disparity as well as other visual cues.

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

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

[9] Marked as $Level2$, layers 2 through $4B$ in [2]Fig. 2.

Appendix
Neuronal Weight Updating

For a winner cell i, update the weights using the lobe component updating principle [35]. That reference also provides a theoretical perspective on the following. Each winner neuron updates using the neuron's own internal temporally scheduled plasticity as $\mathbf{w}_{b,i}(t) = \beta_1 \mathbf{w}_{b,i}(t-1) + \beta_2 z_i \mathbf{b}(t)$ where the scheduled plasticity is determined by its two age-dependent weights

$$\beta_1 = \frac{m_i - 1 - \mu(m_i)}{m_i}, \beta_2 = \frac{1 + \mu(m_i)}{m_i} \tag{27}$$

with $\beta_1 + \beta_2 \equiv 1$. Finally, the cell age (maturity) m_i for the winner neurons increments: $m_i \leftarrow m_i + 1$. All nonwinners keep their ages and weight unchanged. In (27), $\mu(m_i)$ is the plasticity function depending on the maturity m_i of neuron i. The neuron maturity increments every time a neuron updates its weights, starting from zero. The plasticity function prevents learning rate from converging to zero. Details are presented in [35].

Acknowledgment

The authors would like to thank M. Luciw for his helpful discussion and providing write-up for the LCA algorithm and P. Cornwell for editing the final version of the paper. They would also like to thank the reviewers. Their insightful comments and careful corrections has considerably improved the quality and readability of this paper.

References

[1] W. H. Bosking, Y. Zhang, B. Shoefield, and D. Fitzpatrick, "Orientation selectivity and arrangement of horizontal connections in tree shrew striate cortex," *J. Neurosci.*, vol. 17, pp. 2112–2127, 1997.

[2] E. M. Callaway, "Local circuits in primary visual cortex of the macaque monkey," *Annu. Rev. Neurosci.*, vol. 21, pp. 47–74, 1998.

[3] E. M. Callaway, "Feedforward, feedback and inhibitory connections in primate visual cortex," *Neural Netw.*, vol. 17, no. 5-6, pp. 625–632, 2004.

[4] G. Chen, H. D. Lu, and A. W. Roe, "A map for horizontal disparity in monkey v2," *Neuron*, vol. 58, no. 3, pp. 442–450, May 2008.

[5] D. B. Chklovskii and A. A. Koulakov, "Maps in the brain: What can we learn from them?," *Annu. Rev. Neurosci.*, vol. 27, pp. 369–392, 2004.

[6] B. Cumming, "Stereopsis: How the brain sees depth," *Curr. Biol.*, vol. 7, no. 10, pp. 645–647, 1997.

[7] Y. Dan and M. Poo, "Spike timing-dependent plasticity: From synapses to perception," *Physiol. Rev.*, vol. 86, pp. 1033–1048, 2006.

[8] U. R. Dhond and J. K. Aggarwal, "Structure from stereo – A review," *IEEE Trans. Syst., Man, Cybern.*, vol. 19, no. 6, pp. 1489–1510, Nov. 1989.

[9] D. J. Felleman and D. C. Van Essen, "Distributed hierarchical processing in the primate cerebral cortex," *Cereb. Cortex*, vol. 1, pp. 1–47, 1991.

[10] D. J. Fleet, A. D. Jepson, and M. R. M. Jenkin, "Phase-based disparity measurement," in *CVGIP: Image Understand.*, 1991, vol. 53, pp. 198–210.

[11] A. Franz and J. Triesch, "Emergence of disparity tuning during the development of vergence eye movements," in *Proc. Int. Conf. Develop. Learn.*, London, U.K., 2007, pp. 31–36.

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

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

[14] W. E. L. Grimson and D. Marr, L. S. Baumann, Ed., "A computer implementation of a theory of human stereo vision," in *Proc. DARPA Image Understand. Workshop*, 1979, pp. 41–45.

[15] T. Luwang, J. Weng, H. Lu, and X. Xue, "A multilayer in-place learning network for development of general invariances," *Int. J. Human. Robot.*, vol. 4, no. 2, pp. 281–320, 2007.

[16] T. Burwick, J. Wiemer, and W. Seelen, "Self-organizing maps for visual feature representation based on natural binocular stimuli," *Biol. Cybern.*, vol. 82, no. 2, pp. 97–110, 2000.

[17] E. R. Kel, J. H. Schwartz, and T. M. Jessell, Eds., Principles of Neural Science3rd ed. Norwalk, Connecticut, Appleton & Lange, 1991.

[18] T. Kohonen*, Self-Organizating Maps*. New York: Springer-Verlag, 1997.

[19] J. Lippert, D. J. Fleet, and H. Wagner, "Disparity tuning as simulated by a neural net," *J. Biocybern. Biomed. Eng.*, vol. 83, pp. 61–72, 2000.

[20] M. D. Luciw and J. Weng, "Motor initiated expectation through top-down connections as abstract context in a physical world," in *Proc. 7th Int. Conf. Develop. Learn.*, Monterey, CA, 2008.

[21] M. D. Luciw and J. Weng, "Topographic class grouping with applications to 3d object recognition," in *Proc. Int. Joint Conf. Neural Netw.*, Hong Kong, Jun. 2008.

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

[23] R. D. Raizada and S. Grossberg, "Towards a theory of the laminar architecture of cerebral cortex: Computational clues from the visual system," *Cereb. Cortex*, vol. 13, no. 1, pp. 100–113, Jan. 2003.

[24] J. C. A. C. Read and B. G. G. Cumming, "Sensors for impossible stimuli may solve the stereo correspondence problem," *Nat. Neurosci.*, Sep. 2007.

[25] A. W. Roe, A. J. Parker, R. T. Born, and G. C. DeAngelis, "Disparity channels in early vision," *J. Neurosci.*, vol. 27, no. 44, pp. 11820–11831, Oct. 2007.

[26] P. R. Roelfsema and A. van Ooyen, "Attention-gated reinforcement learning of internal representations for classification," *J. Neur. Comput.*, vol. 17, pp. 2176–2214, 2005.

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

[28] M. Solgi and J. Weng, "Developmental stereo: Topographic iconic-abstract map from top-down connection," in *Proc. 1st Symp. Series New Develop. Neural Netw.*, Auckland, New Zealand, Nov. 2008.

[29] I. J. Stockman*, Movement and Action in Learning and Development: Clinical Implications for Pervasive Developmental Disorders*. Amsterdam, The Netherlands: Elsevier, 2004.

[30] J. Weng, "Image matching using the windowed Fourier phase," *Int. J. Comput. Vis.*, vol. 11, no. 3, pp. 211–236, 1993.

[31] J. Weng and M. D. Luciw, "Optimal in-place self-organization for cortical development: Limited cells, sparse coding and cortical topography," in *Proc. 5th Int. Conf. Develop. Learn.*, Bloomington, IN, May 31–3, 2006.

[32] J. Weng and M. D. Luciw, "Dually optimal neural layers: Lobe component analysis," *IEEE Trans. Autonom. Mental Develop.*, vol. 1, pp. 86–97, May 2009.

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

[34] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen, "Autonomous mental development by robots and animals," *Science*, vol. 291, no. 5504, pp. 599–600, 2001.

[35] J. Weng and N. Zhang, "Optimal in-place learning and the lobe component analysis," in *Proc. World Congress Comput. Intell.*, Vancouver, BC, Canada, Jul. 16–21, 2006.

[36] J. Weng and W.-S. Hwang, "From neural networks to the brain: Autonomous mental development," *IEEE Comput. Intell. Mag.*, vol. 1, no. 3, pp. 15–31, Aug. 2006.

[37] P. Werth, S. Scherer, and A. Pinz, "Subpixel stereo matching by robust estimation of local distortion using Gabor filters," in *Proc. 8th Int. Conf. Comput. Anal. Images Patterns*, London, U.K., 1999, pp. 641–648.

[38] A. K. Wiser and E. M. Callaway, "Contributions of individual layer 6 pyramidal neurons to local circuitry in macaque primary visual cortex," *J. Neurosci.*, vol. 16, pp. 2724–2739, 1996.

[39] C. L. Zitnick and T. Kanade, "A cooperative algorithm for stereo matching and occlusion detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 7, pp. 675–684, Jul. 2000.

Mojtaba Solgi received the B.Sc. degree from Amirkabir University of Technology, Iran, in 2007, and the M.Sc. degree from Michigan State University (MSU), East Lansing, in 2009, both in computer science.

He is currently a doctoral student at MSU, where he is a member of the Embodied Intelligence Laboratory. He is also a member of the interdepartmental Graduate Specialization in Cognitive Science at MSU. His research involves the study of computational models for autonomous development of mental capabilities — especially for stereo vision and recognition. He was a member of RoboCup Soccer Simulation (http://www.robocup.org/) Technical Committee (2006–2007), and he is currently serving as site coordinator for Mental Development Repository website (http://www.mentaldev.org/).

Juyang Weng received the B.Sc. degree in computer science from Fudan University, Shanghai, China, in 1982, and the M.Sc. and Ph.D. degrees in computer science from University of Illinois, Urbana, in 1985 and 1989, respectively.

He is now a Professor at the Department of Computer Science and Engineering, Michigan State University, East Lansing. He is also a Faculty Member of the Cognitive Science Program and the Neuroscience Program at Michigan State University. Since the work of Cresceptron (ICCV 1993), he expanded his research interests to biologically inspired systems, especially the autonomous development of a variety of mental capabilities by robots and animals, including perception, cognition, behaviors, motivation, and abstract reasoning skills. He has published over 200 research articles on related subjects, including task muddiness, intelligence metrics, mental architectures, vision, audition, touch, attention, recognition, autonomous navigation, and other emergent behaviors. He and his coworkers developed SAIL and Dav robots as research platforms for autonomous development.

Dr. Weng was a member of the Executive Board of the International Neural Network Society (2006–2008), a program chairman of the NSF/DARPA funded Workshop on Development and Learning 2000 (1st ICDL), a program chairman of 2nd ICDL (2002), the chairman of the Autonomous Mental Development Technical Committee of the IEEE Computational Intelligence Society (2004–2005), the Chairman of the Governing Board of the International Conferences on Development and Learning (ICDL) (2005–2007, http://cogsci.ucsd.edu/ triesch/icdl/), a general chairman of 7th ICDL (2008), and the general chairman of 8th ICDL (2009). He is an Associate Editor of IEEE Transactions on Pattern Recognition and Machine Intelligence and an Associate Editor of IEEE Transactions on Image Processing. He is also an Editor-in-Chief of *International Journal of Humanoid Robotics* and an Associate Editor of the new IEEE Transactions on Autonomous Mental Development.