Spatio–Temporal Adaptation in the Unsupervised Development of Networked Visual Neurons

Dongyue Chen, Liming Zhang, *Senior Member, IEEE*, and Juyang (John) Weng, *Fellow, IEEE*

***Abstract*—There have been many computational models mimicking the visual cortex that are based on spatial adaptations of unsupervised neural networks. In this paper, we present a new model called neuronal cluster which includes spatial as well as temporal weights in its unified adaptation scheme. The "in-place" nature of the model is based on two biologically plausible learning rules, Hebbian rule and lateral inhibition. We present the mathematical demonstration that the temporal weights are derived from the delay in lateral inhibition. By training with the natural videos, this model can develop spatio–temporal features such as orientation selective cells, motion sensitive cells, and spatio–temporal complex cells. The unified nature of the adaption scheme allows us to construct a multilayered and task-independent attention selection network which uses the same learning rule for edge, motion, and color detection, and we can use this network to engage in attention selection in both static and dynamic scenes.**

***Index Terms*—Attention selection, developmental algorithm, Hebbian learning rule, lateral inhibition, receptive fields (RFs).**

I. Introduction

IT IS known that a cell in the primary visual cortex responds to light stimuli in a restricted region of the visual field called its receptive field (RF) [1]. As early as in 1962, Hubel and Wiesel reported that some neurons in the visual cortex of cats responded preferentially to stimuli with particular spatial orientations [2]. Such spatial property enables these neurons to detect oriented edges. Since then, various filters were proposed to mimic the neuronal spatial synaptic weights with orientation selectivity [3]–[6]. Moreover, some studies show the reversed temporal profile of the dynamic RF, which makes such neurons sensitive to motion [7], [8]. The corresponding spatio–temporal RF models were proposed later [9]–[11]. All these studies simulate the cells by fixed RFs or filters, without addressing their developmental process: how biological life, in particular biological networks, comes about through epigenesis (i.e., complex interactions of the cells with the environment), and how they continue to adapt to the spatio–temporal statistics of the environment later in life.

Biological results show that some neurons in V1 area adapt their input weights and preferred orientations to the visual environment especially in the critical period of their sensory development. In an early study [12], some newborn kittens were raised in an environment displaying only vertical gratings. Six weeks after their birth, the orientation sensitivity for the population of cells in the primary visual cortex was found to largely concentrate around the vertical orientation. That means the preferred features of neurons in the visual cortex are affected largely by the environment. It was reported that cells in the retina, the first processing area along the visual pathway, change their feature properties after a few seconds under a new type of stimuli [13]. The changes of the RF and weights are helpful to improve sensory predictive coding under the new image statistics. These results indicate that the pattern selectivity of the cells changes dynamically according to the postnatal visual environment. That is why the developmental models are necessary to describe the behavior of biological cells. The fixed filters of existing models cannot account for these biological facts, and as we will explain later in this paper, they are incapable of effectively and efficiently representing the spatio–temporal input signal space.

Do cells for different sensory modality (e.g., vision versus audition) require very different developmental rules? It was reported in a study that when the retinal projections of a ferret were redirected neonatally to its auditory thalamus [14], the rewired auditory cortex, after weeks of development, showed typical characteristics of visual cells. It appears that different neurons in the human sensory cortex may have similar developmental rules, and the genetically specified developmental rules and the stimuli from the external environment may jointly decide what spatio–temporal patterns the cells detect. Furthermore, the developmental rules are critical not only for modeling cortical processing for visual signals, but potentially for other types of signals as well. If we can find a neuronal model with simple learning rule that simulates human or animal vision, it will be easier to apply this model to computer vision.

Then what are the learning rules for mimicking the development of cells in the visual cortex? They are grouped into a few categories. Early on, there were global principal component analysis (PCA) models, which did not exhibit adequate spatial–opponent patterns in its principal components for natural images [18], [15], [16], [17]. This issue was mitigated with the introduction of local PCA models such as the competitive model [19] and lobe component analysis (LCA) [20], and the

Manuscript received March 25, 2007; revised November 22, 2007 and November 15, 2008; accepted November 19, 2008. First published May 19, 2009; current version published June 03, 2009. This work was supported by the National Science Foundation of China under Grant NSF60571052 and by the Key Subject Foundation of Shanghai under Grant B112.

D. Chen was with the Department of Electronic Engineering, Fudan University, Shanghai 200433, China. He is now with the College of Information Science and Engineering, Northeastern University, Shenyang 110004, China (e-mail: chendongyue@ise.neu.edu.cn).

L. Zhang is with the Department of Electronic Engineering, Fudan University, Shanghai 200433, China (e-mail: lmzhang@fudan.edu.cn).

J. Weng is with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824 USA (e-mail: 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/TNN.2009.2015082

locally extracted principal components produced features with more complete orientations. Self-organizing map (SOM) [23] and adaptive-subspace self-organizing map (ASSOM) [21], [22] models introduced topographic map that was fitted to the distribution of the input samples. Later, there was independent component analysis (ICA) proposed by Field [25], Olshausen and Field [26], and Bell and Sejnowski [27]. The models based on ICA produce spatial weights of RFs that are local if the inputs have been whitened. While ICA has the issue of complex calculation and high storage demands, some improved models based on ICA with local computation were proposed to address such issues [28]–[31]. The recent model multilayer in-place learning networks (MILNs) by Weng *et al.* linked LCA with the cortical anatomy and the cell-centered update in the laminar cortex, using a concept "in-place" that is more biologically precise than the looser concept "local learning," etc., [32] These above models focus on the adaptation of the spatial RF, not the temporal RF, and are unable to produce topographic maps. Recently, the effort to generate the spatio–temporal RF with ICA [35] has attracted attention, but the neuronal computation, learning, and requirement of prewhitening are still too complex to be accountable by the biological principles of in-place learning.

In this paper, we create an integrated spatio–temporal network and also employ the topographic map in its simple and unified adaptation scheme. It has not been achieved with ICA, which requires higher order correlation to create topographic maps [33], [34]. We name our network the neuronal cluster, and its purpose is to emulate the visual sensory development of the neurons in V1 area. In the neuronal cluster, many neurons with lateral connections share the same RF (although the same model can be directly extended to the case of local connections). A unified developmental rule is presented to enable the model to adapt both its spatial and temporal weights to its environment. By training the model through natural videos, the feedforward weights of the neuronal cluster display both topographic arrangement in space and reversed profiles in time, which are consistent with biological evidence reported in the literature. Using the same model and developmental rule, we obtain the chromatic antagonistic feedforward weights by training them in a color environment with random sinusoidal gratings. Furthermore, we formulate that a single neuron's temporal profile is caused from the slightly delayed lateral connections with its adjacent neurons. A mathematical proof is also proposed in this paper to support our temporal model and the unified learning rule.

We apply our model to computer vision and propose a visual attention selection system using a multilayer neural network composed of many neuronal clusters. In many existing systems for bottom-up and task-independent visual attention, several different fixed saliency filters are hand designed, each representing a different type of saliency measure [36]–[39]. The proposed model is different from them, for it uses only one neural network to extract all types of features (color, edge, motion, etc.), which is more biologically plausible since all the connected weights between neurons in the proposed model are generated by an unsupervised learning mechanism in a biologically inspired way instead of by statistically predefined fixed filters.

The remainder of this paper is organized as follows. Section II introduces the biological model of a single cell. In Section III, we present the structure of the neuronal cluster and its response function. The learning algorithms for the neuronal cluster are described in Section IV. Section V gives some simulation results of the neuronal cluster. An example of its application in visual attention selection is provided in Sections VI and VII. The conclusions and discussion are provided in Section VIII.

II. Spatio–Temporal Model of Single Neuron

As the basic unit of the neural system, a neuron triggers action potentials in response to stimuli and inputs from other neurons. The action potentials convey information throughout the biological neural networks. If we ignore the brief duration and shape of each action potential, an action potential sequence can be characterized as a spike train. Consequently, the response of a single neuron can be estimated by the time-dependent firing rate $r(t)$ at the temporal resolution that we concentrate on. Many computational models were proposed to simulate the response function of a single neuron [40]–[45]. Based on these works, we know that the response of a single neuron at time t typically depends on the input stimuli over a temporal window prior to t. So the simplest estimation $r_e(t)$ (e denotes "estimation") of the time-dependent firing rate $r(t)$ can be expressed as a function of the weighted sum of the stimuli s [43], [45]

$$r_e(t) = r_0 + F(L(t)) \tag{1}$$

with

$$L(t) = \int_0^\infty D(\tau)s(t-\tau)d\tau \tag{2}$$

where r_0 is the background firing rate. $L(t)$ represents the linear term (weighted sum of stimuli) in the neuronal response, and $F(x)$ is a nonlinear monotonic nondecreasing function which constrains the firing rate within a limited range. In (2), $s(t-\tau)$ is the stimulus at time $t-\tau$. For the single neuron illustrated in Fig. 1(a), the factor $D(\tau)$ represents the temporal weight at τ and characterizes the joint effect of the cytoplastic environment inside the neuron. Considering the integral of the stimuli over the spatial input area [Fig. 1(b)], $L(t)$ in (2) can be rewritten as

$$L(t) = \int_0^\infty \int D(x,y,\tau)s(x,y,t-\tau)dxdy \tag{3}$$

where $s(x,y,t-\tau)$ is the stimulus at time $t-\tau$ and at position (x,y). $D(x,y,\tau)$ is the spatio–temporal weight of the neuron. According to the biological experiments [8], [42], the spatio–temporal weights of some neurons can be accurately modeled as the product of the independent spatial weights and temporal weights that is shown as follows:

$$D(x,y,\tau) = D_{\mathrm{sp}}(x,y)D_{\mathrm{te}}(\tau) \tag{4}$$

where $D_{\mathrm{sp}}(x,y)$ and $D_{\mathrm{te}}(\tau)$ are the spatial weight and the temporal weight, respectively, and the subscript "sp" and "te" are abbreviation of "spatial" and "temporal," respectively. Such neurons are said to have separable space–time RFs, which are called as separable neurons. In this paper, we focus on the separable neurons exclusively. Some researches indicate that

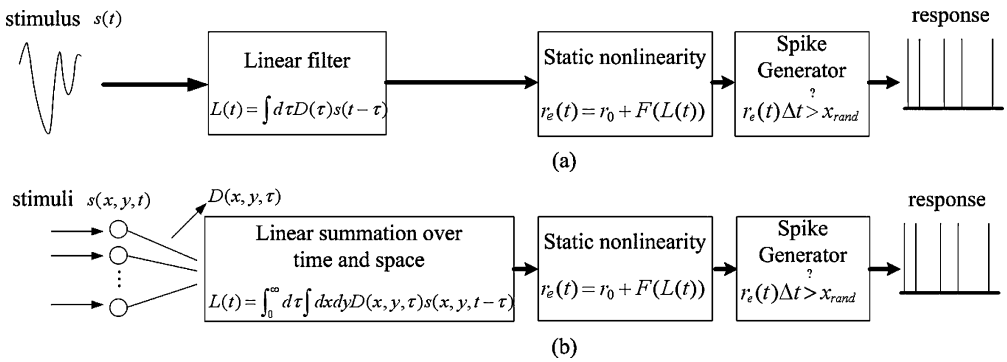

Fig. 1. Flow charts of generating spiking response by a single neuron in two different cases: (a) the input is a stimulus and the spatial integration of inputs is out of the consideration, and (b) the input is stimuli on the RF of the neuron, and the response is an account for the integration of stimuli on both space and time. $x_{\rm rand}$ is a threshold. A spike is generated at t when $r_e(t)\Delta t > x_{\rm rand}$.

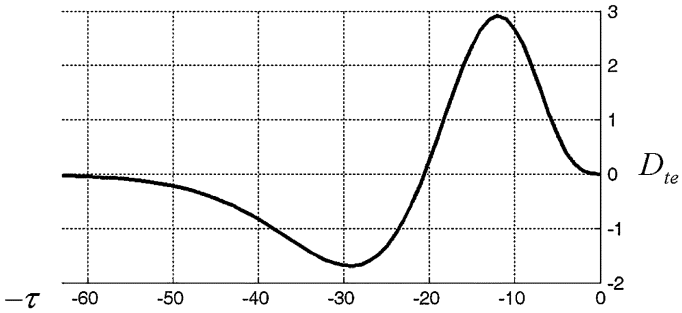

Fig. 2. Example of the profile of the temporal weights $D_{\rm te}(\tau)$ of a neuron where $\tau = 0$ is the time of firing. Plot is the result by Adelson and Bergen [9].

the typical patterns of $D_{\rm sp}$ in V1 area can be approximated by Gabor functions [3]–[5]. The Gabor-like RF indicates the orientation selectivity of the neurons [2]. On the other hand, some fixed functions [7]–[9] were presented to approximate the reversed temporal profile of the dynamic RF, e.g., the function shown in [7]

$$D_{\rm te}(\tau) = \begin{cases} \alpha \exp(-\alpha\tau)\left(\dfrac{(\alpha\tau)^5}{5!} - \dfrac{(\alpha\tau)^7}{7!}\right), & \tau > 0 \\ 0, & \tau < 0 \end{cases} \quad (5)$$

where α is a constant. An example of $D_{\rm te}(\tau)$ in (5) is plotted in Fig. 2. We can see that only signals that fall in the temporal window before the firing have an impact on the current firing. A temporal profile of $D_{\rm te}(\tau)$ is shown as a "reversed" function. The word "reversed" means a strong biphasic profile from positive (negative) to negative (positive). The reversed profile may be caused by the complexity of returns of inhibitory signals in the recurrent network around the neuron, which makes the neuron more sensitive to the changes of the inputs (see Section III-C in detail).

III. Spatio–Temporal Model of Neuronal Cluster

In fact, the changes of both the spatial and temporal weights of a neuron are a part of the developmental process. We propose a computational model called the neuronal cluster that consists of several neighboring neurons which share the same input area as shown in Fig. 3. The input to each neuron includes the input from the previous layer and the input from the same layer. The former is called bottom-up excitation and the latter is called lateral inhibition. Top–down projections are also possible but for the scope of this work we do not consider them explicitly. In Fig. 3, $C_i, i = 1, 2, \ldots, n$ represent the neurons in the neuronal cluster. Let us denote by ψ the spatial input area which is composed of $l \times l$ input units. For the sake of consistency, we still use the same symbols as we used in Section II to represent the neuronal responses and the spatio–temporal weights; however, superscripts are used to denote the indexes of neurons.

A. Bottom-Up Response Function

In Fig. 3, each neuron C_i receives inputs from a given spatial input area ψ in the previous layer. For convenience, we assume that $F(x) = x$ and $r_0 = 0$ in (1) (the result can be readily extended to the general case). Then, the output $r_e^i(t)$ of the ith neuron is equal to the linear term $L^i(t)$. The stimulus at position (x, y) and at time t is denoted by $s(x, y, t)$. All the stimuli over the spatial input area ψ can be written as a vector $\mathbf{S}(t) \in R^{l^2}$. The spatial weights $D_{\rm sp}(x, y)$ of neuron C_i can be rewritten as a vector $\mathbf{D}_{\rm sp}^i \in R^{l^2}$. The current stimuli on ψ give a contribution to the response of neuron C_i, which is denoted as $L_{\rm sp}^i(t)$

$$L_{\rm sp}^i(t) = \frac{\mathbf{D}_{\rm sp}^{i\ T}\mathbf{S}(t)}{\left\|\mathbf{D}_{\rm sp}^i\right\|} \quad (6)$$

where the spatial weight vector $\mathbf{D}_{\rm sp}^i$ is normalized by $\|\mathbf{D}_{\rm sp}^i\|$.

Let N denote the maximal latency of the temporal response of all the neuron. That is, the contribution of the stimuli beyond N is neglected for the current neuron's firing rate. The temporal weights of neuron C_i, as the discrete version of the temporal profile, can be written as a vector $\mathbf{D}_{\rm te}^i = [D_{\rm te}^i(0), D_{\rm te}^i(1), \ldots, D_{\rm te}^i(N-1)]^T$. The accumulation of $L_{\rm sp}^i(t)$ over the time domain, with the constraint of a normalized temporal weight vector, can be computed by

$$L_{\rm te}^i(t) = \mathbf{D}_{\rm te}^{i\ T} \cdot \left[L_{\rm sp}^i(t), L_{\rm sp}^i(t-1), \ldots, L_{\rm sp}^i(t-(N-1))\right]^T / \left\|\mathbf{D}_{\rm te}^i\right\|, \quad i = 1, 2, \ldots, n. \quad (7)$$

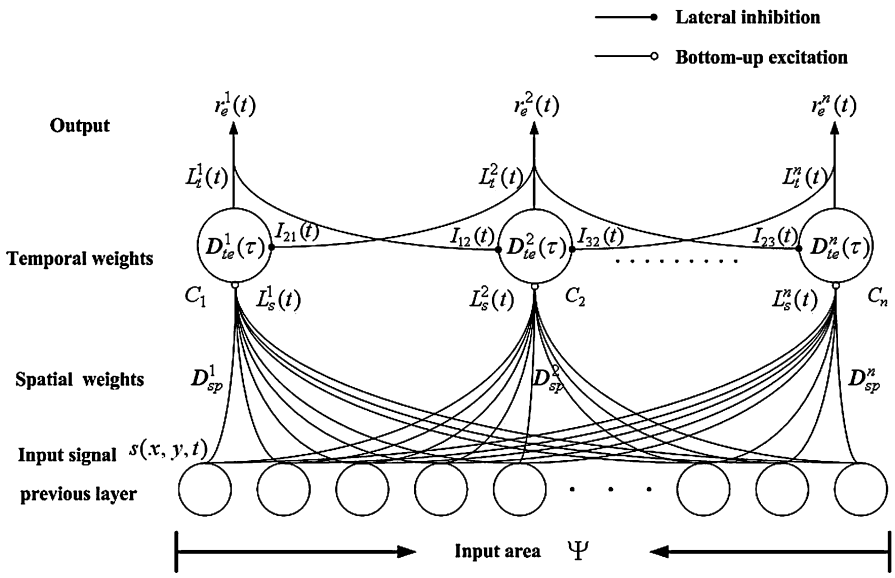

Fig. 3. Architecture of a neuronal cluster model. The hollow circles at the bottom are the cells that receive the input signal and transfer it to neurons in the next layer. The lines with small hollow circles in their terminal are the bottom-up spatial weights. The lines with small solid circles are the lateral inhibitory weights.

Substituting (6) for the $L^i_{\rm sp}(t)$ in (7), the response $r^i_e(t)$ (subscript "e" means $r^i_e(t)$ is an evaluation of the real time-dependent firing rate) of neuron C_i can be written as

$$r^i_e(t) = L^i_{\rm te}(t) = \sum_{\tau=0}^{N-1}\sum_{y\in\psi}\frac{D^i_{\rm sp}(x,y)}{\|\mathbf{D}^i_{\rm sp}\|}\frac{D^i_{\rm te}(\tau)}{\|\mathbf{D}^i_{\rm te}\|}s(x,y,t-\tau). \quad (8)$$

Equation (8) is the response function of a separable neuron whose spatio–temporal weight can be written as the product of the spatial weight $D^i_{\rm sp}(x,y)$ and the temporal weight $D^i_{\rm te}(\tau)$. Equations (6)–(8) give a functional description for the response of a single neuron, which is supported by some biological studies [8], [42]. However, there is no evidence that a real neuron has multiple internal passages with different time delays that form temporal weight profile $D^i_{\rm te}(\tau)$, given in (8). We hypothesize that the compounding effect of the delayed inhibitory signals between a neuron and its recurrently connected neurons plays a major role for the temporal profile of the neuron. We will discuss it in Section III-C.

B. Lateral Connections

In Section III-A, we only take into account the bottom-up signals. In fact, there are lateral inhibitory connections between neurons in the proposed neuronal cluster model (Fig. 3), which enable the neighboring neurons in the same layer to suppress each other and make the network more stable and efficient [46]. Neurons with high responses are more effective to suppress their neighboring neurons, which, in turn, reduce the suppression from these neurons. Thus, the responses of the neurons with lateral connections can be described by following:

$$\begin{aligned} r^{i'}_e(t) &= r^i_e(t) + I_i(t) \\ &= \sum_{\tau=0}^{N-1}\sum_{y\in\psi}\frac{D^i_{\rm sp}(x,y)}{\|\mathbf{D}^i_{\rm sp}\|}\frac{D^i_{\rm te}(\tau)}{\|\mathbf{D}^i_{\rm te}\|}s(x,y,t-\tau) + I_i(t) \end{aligned} \quad (9)$$

where $r^{i'}_e(t)$ is the response of neuron C_i when it receives lateral inhibitory signals from its neighboring neurons, and $I_i(t)$ is the total effect of the lateral inhibitory signals at time t from other neurons to neuron C_i. In general, it takes on a nonpositive value. To avoid the expensive iterative computation in such a recurrent network, we approximate the global effect of lateral inhibition by two noniterative steps: 1) ranking, and 2) scaling according to rank.

The"ranking" step computes the bottom-up response $r^i_e(t)$ of all neurons using (8) and selects the strongest neuron as the winner that is denoted by C_c. C_c is called the top neuron at time t. The subscript c is defined as

$$c = \arg\max_i \left(r^i_e(t)\right), \qquad i = 1, 2, \ldots, n. \quad (10)$$

The step of "scaling according to rank" is to relatively boost the winning neuron by suppressing other runner-ups. Thus, the suppressed response $r^{i'}_e(t)$ can be approximated by following:

$$r^{i'}_e(t) = \alpha_i(t) r^i_e(t) \quad (11)$$

where $\alpha_i(t)$ is the scaling factor for the neuron C_i, which is defined as

$$\alpha_i(t) = \exp\left(-\frac{d^2_{ci}(t)}{\sigma^2_{\rm sp}(t)}\right) \quad (12)$$

where $d_{ci}(t)$ is the distance between neurons C_i and C_c, and $\sigma_{\rm sp}(t)$ controls the range of the inhibitory region. Therefore, all other neurons except top neuron C_c will be inhibited in varying degrees. Only the top neuron keeps its original response. Here the parameter $\sigma_{\rm sp}(t)$ changes with time

$$\sigma_{\rm sp}(t) = \sigma_{\rm sp}(0)\exp\left(-\frac{t}{\lambda}\right) \quad (13)$$

where $\sigma_{\rm sp}(0)$ is the initial size of the inhibitory area, and λ is a time constant. Generally, $\sigma_{\rm sp}(0)$ is not to be smaller than the radius of the area of the neuronal cluster. If the maximal training

time is $T_{\max}$, λ should be initialized to make sure $\sigma_{\rm sp}(T_{\max}) \geq d_{\rm unit}$, where $d_{\rm unit}$ is the distance between two neighboring neurons. In our experiment, the value of λ is initialized empirically. In general, λ is proportional to the number of training images.

C. An Interpretation of Temporal RFs

As mentioned in Section III-B, the temporal response is the weighted summation of the stimuli over a sliding temporal window called the temporal RF. The storage effect of various chemical transmitters inside the cell body may partially explain the effect of the temporal profile of a neuron. However, the temporal profile is unlikely to be fully implemented by a single biological neuron, as the concentration of chemical transmitters in the body of the cell alone cannot fully explain the "reversed effect" of the temporal profile. There has been no biological evidence that each neuron has a multiunit temporal storage inside itself either. Here we give a possible interpretation that the temporal window should be considered as the approximate effect accounting for the dynamic effects of neighboring connected neurons while they interact through input and lateral connections. In other words, the suppressed response $r_e^{i'}(t)$ of neuron C_i is decided by both the bottom-up signals directly from its previous layer and the delayed lateral signals from the neighboring neurons in the same layer.

Here we provide a mathematical account for the effect of lateral connections within a neuronal cluster. Without loss of generality, we consider a simple case of the neuronal cluster that includes only two neurons C_1 and C_2. In the neuronal cluster, neurons C_1 and C_2 share the same RF but have different spatial weight vectors $D_{\rm sp}^1$ and $D_{\rm sp}^2$, respectively. The lateral inhibitory weights are denoted by H_{12} and H_{21} (they take on negative values), respectively. The input is $S(t)$, and the corresponding responses of C_1 and C_2 at time t are $r_e^{1'}(t)$ and $r_e^{2'}(t)$, respectively. If the lateral signal from one neuron to the other is always delayed for one time unit, the neuron's response at time t can be computed by

$$r_e^{i'}(t) = L_{\rm sp}^i(t) + H_{ji} r_e^{j'}(t-1), \qquad i = 1, 2, \qquad i \neq j \quad (14)$$

where $L_{\rm sp}^i(t)$ can be computed by (6) that considers only the spatial summation at time t. The second term $H_{ji} r_e^{j'}(t-1)$ on the right-hand side of (14) represents the lateral inhibitory signal from its neighboring neuron, which is different from $I_i(t)$ in (9) because it is a delayed inhibitory signal. Equation (14) is a recursive formula for computing the response of the neuron. Substituting (14) for $r_e^{j'}(t-1)$ in (14) recursively, we have (15), shown at the bottom of the page.

This equation states that the response of a neuron at time t with lateral connections is a weighted summation of previous responses from the laterally connected neurons. This is a basis for our computational model of the biological "reversed" temporal profile.

For the convenience of further discussion, we suppose that the input sequence $S(t)$ is stationary so that $L_{\rm sp}^j(t-\tau) \approx \kappa L_{\rm sp}^i(t-\tau), \tau = 1, 2, \ldots, t-1$ for a constant κ. Substituting this expression into (15), we have

$$r_e^{i'}(t) = \sum_{\tau=0}^{t-1} D_{\rm te}^i(\tau)^T L_{\rm sp}^i(t-\tau) \quad (16)$$

where

$$D_{\rm te}^i(\tau) = \begin{cases} H_{ji}^{\tau/2} H_{ij}^{\tau/2} & \tau = 0, 2, 4, \ldots \quad \tau \leq t \\ \kappa H_{ji}^{(\tau+1)/2} H_{ij}^{(\tau-1)/2} & \tau = 1, 3, 5, \ldots \quad \tau \leq t. \end{cases} \quad (17)$$

The expression of (17) shows that the temporal weight $D_{\rm te}^i$ will decrease exponentially as τ increases. That inspires us to approximate the neuronal response by taking into account only the first N terms in (16) and neglecting other smaller terms. Thus, the normalization of (16) approximates essentially to the spatio–temporal response function in (7).

Since the H_{ij} and H_{ji} are the inhibitory weights, $H_{ij} < 0$ when $i \neq j$ and $H_{ij} = 0$ when $i = j$. In the case of multiple neurons, the inhibitory weights H_{ij} change smoothly as the distance d_{ij} increases. So it is possible that the synthetic effect of the lateral inhibitory connections displays a smooth reversed profile of the temporal weights.

Above deductions indicate that the nature of the temporal RF is possibly the compounding effect of lateral inhibitory connections in a recurrent network. Since the temporal weights are caused by lateral inhibitory weights, it is possible that the Hebbian rule can also be used for training the temporal weights. This consideration motivates us to use a unified learning rule to adapt both the spatial and temporal weights, which opens up a new way to explore the intrinsic developmental rule of the visual cortex.

In summary, our model is designed with the following novel characteristics:

1) both spatial weights and temporal weights of a neuron are unified into one adaptation scheme;
2) give a mathematical demonstration for the possibility that the nature of the temporal weights is the compounding effect of the delayed lateral inhibition in the recurrent network.

IV. Developmental Algorithm for Spatio–Temporal Weights

In our model, the competition and lateral inhibition between neurons in a neuronal cluster follow the deduction in (10)–(13). Both the spatial and temporal weights of each neuron adapt to its environment, and the adaptive rule is controlled by the cell-centered developmental mechanism of neurons. The cell-centered property means that each neuron is responsible for the development of itself while it interacts with its environment and there

$$r_e^{i'}(t) = \begin{cases} L_{\rm sp}^i(t) + H_{ji} L_{\rm sp}^j(t-1) + H_{ji} H_{ij} L_{\rm sp}^i(t-2) + \cdots + H_{ji}^{t/2} H_{ij}^{(t-2)/2} L_{\rm sp}^j(1), & t \text{ is even} \\ L_{\rm sp}^i(t) + H_{ji} L_{\rm sp}^j(t-1) + H_{ji} H_{ij} L_{\rm sp}^i(t-2) + \cdots + H_{ji}^{(t-1)/2} H_{ij}^{(t-1)/2} L_{\rm sp}^j(1), & t \text{ is odd.} \end{cases} \quad (15)$$

TABLE I
PARAMETERS FOR THE NEURONAL CLUSTER IN SIMULATION 1

The number of neurons (n)	14×10	Temporal window width (N)	64
Topological neighborhood	Hexagonal	$\sigma_{sp}(0)$ in (13)	$10 \times d_{unit}$
Input area (ψ)	15×15(pixel)	λ in (13)	100000
Input stimuli	Intensities	δ_{te} in (22)	13

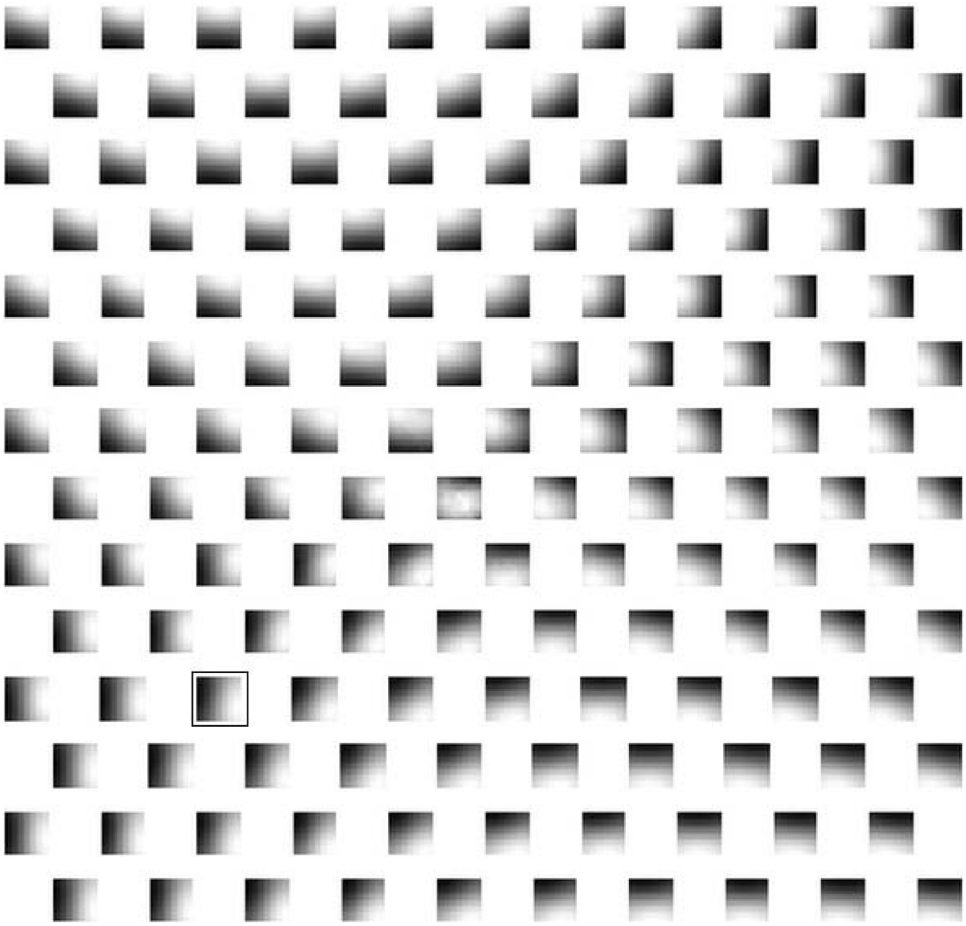

Fig. 4. Spatial weight vectors of 14×10 neurons in a neuronal cluster, $D^i_{\text{sp}}(t), i = 1, 2, \ldots, 140$, trained by our developmental algorithm. Each weight vector is rearranged as a 15×15 square.

is no centralized controller that controls the development of any neuronal cluster, which is supported by some biological theories [47], [48]. Thus, we need a cell-centered computational mechanism that models the development of both the spatial and temporal weights.

In the proposed model, we use the Hebbian rule [24] to adapt the spatio–temporal weights to the input vectors. According to the Hebbian rule, the connecting weight between two neurons, denoted by w, is modified with the increment $\Delta w = \eta r^1 r^2$, where η is the learning rate; r^1 and r^2 are the responses of the two neurons, respectively. However, exactly what learning rate is appropriate for the development of the neuron through its entire lifetime is still an open question when the input is an unknown nonstationary process. However, if the input process of a neuron changes slowly (e.g., when the cortex is nearly mature), we can evaluate the learning rate by minimizing the mean square error between the estimated weight vector and the target lobe component vector, as Weng and Zhang reported in [20].

A. Development of Spatial Weights

In Fig. 3, the inputs of neuron C_i are the responses of neurons in the previous layer, which are specified by vector $\mathbf{S}(t)$. The response of neuron C_i with the lateral inhibition is denoted by $r^{i\prime}_e(t)$, and can be computed by (9) and (11). The connecting weights between neuron C_i and its input area in the previous layer are characterized as a spatial weight vector $\mathbf{D}^i_{\text{sp}}$. According to the Hebbian rule and the optimal learning rate pro-

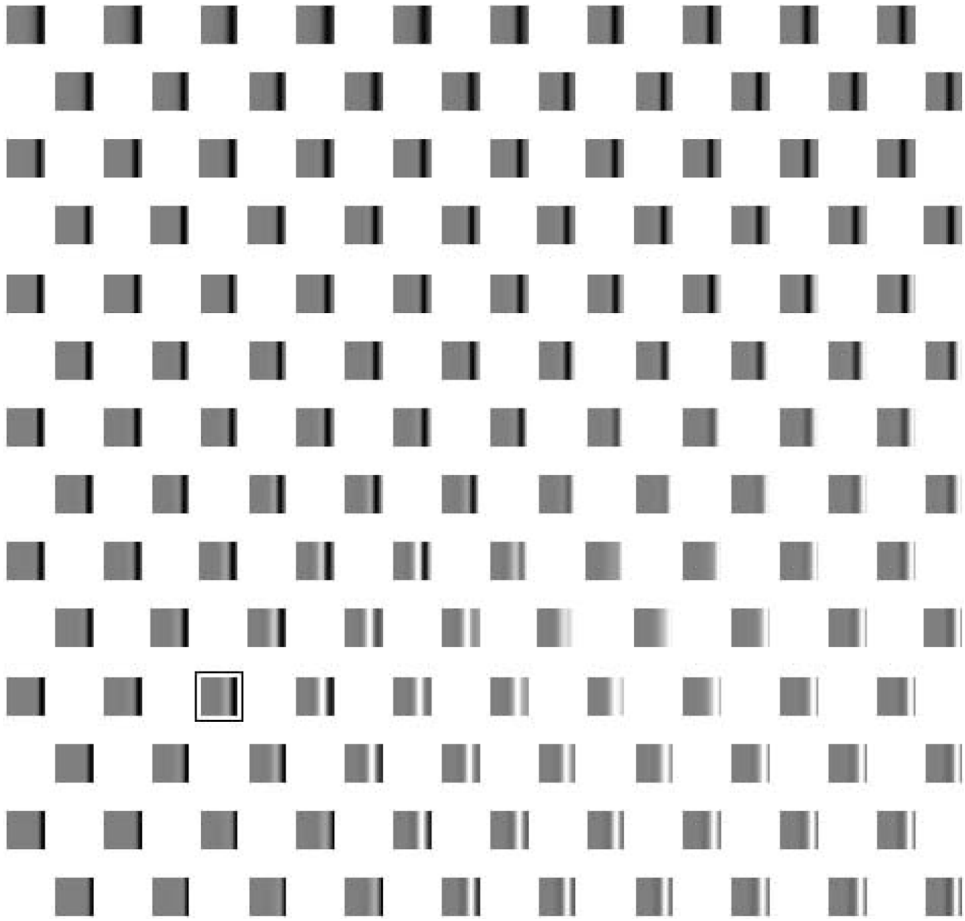

Fig. 5. Visualization of temporal weight vectors of 140 neurons in the trained neuronal cluster. Each temporal weight profile is shown as an image with the same intensity along columns.

posed by Weng and Zhang [20], the update of the spatial weight vector can be written as

$$\mathbf{D}^i_{\rm sp}(t+1) = w_1\mathbf{D}^i_{\rm sp}(t) + w_2\mathbf{S}(t+1)r_e^{i'}(t+1), \quad i = 1,2,\ldots,n. \tag{18}$$

The update for $\mathbf{D}^i_{\rm sp}$ in (18) is the Hebbian increment above, but the minimization further gives a closed-form solution for a series of optimal step sizes at different neuronal ages, each of which is represented as a pair of retention rate w_1 and learning rate w_2 that sum to one. For speeding up the convergence of the spatial weight vector $\mathbf{D}^i_{\rm sp}(t)$, w_1 and w_2 are designed as follows [20]:

$$w_1 = \frac{t-\mu(t+1)}{t+1} \quad w_2 = \frac{1+\mu(t+1)}{t+1}. \tag{19}$$

When $\mu(t) \equiv 0$, the update in (18) gives an incremental temporal average of the "observations" $S(t)r_e^{i'}(t), i = 1,2,\ldots,n$

$$\mathbf{D}^i_{\rm sp}(t+1) = \frac{1}{t+1}\sum_{t'=1}^{t+1}\mathbf{S}(t')r_e^{i'}(t'). \tag{20}$$

When $\mu(t)$ is positive, more weight is given for the new observations, which keeps sufficient plasticity of the spatial weight throughout its entire life. A typical value $\mu(t) = 2$ can be chosen for sufficiently large t (for more discussion of the amnesic weight, see [20]).

B. Development of Temporal Weights

Based on the Hebbian rule, the imaginary temporal weights of a neuron also adapts to the input signals. Considering the temporal weights that simulate the delays of signals propagating through the neighboring connected neurons, the input vector for the temporal weights can be written as

$$\mathbf{S}^i_{\rm te}(t) = [\rho_0 L^i_{\rm sp}(t), \rho_1 L^i_{\rm sp}(t-1), \ldots, \rho_{N-1} L^i_{\rm sp}(t-(N-1)),] \tag{21}$$

where $\rho_\tau, \tau = 0,1,\ldots,N-1$ are the attenuation factors of the delayed signals. We define ρ_τ as a Gaussian-like function as follows:

$$\rho_\tau = \exp\left(-\frac{\tau^2}{2\delta^2_{\rm te}}\right), \qquad \tau = 0,1,\ldots,N-1 \tag{22}$$

where $\delta_{\rm te}$ is a parameter that controls the attenuation speed of the lateral inhibitory signals. So the temporal weight vector of neuron C_i can be updated by the following with the Hebbian rule:

$$\mathbf{D}^i_{\rm te}(t+1) = \frac{t-\mu(t+1)}{t+1}\mathbf{D}^i_{\rm te}(t) + \frac{1+\mu(t+1)}{t+1}\mathbf{S}^i_{\rm te}(t+1)r_e^{i'}(t+1),$$

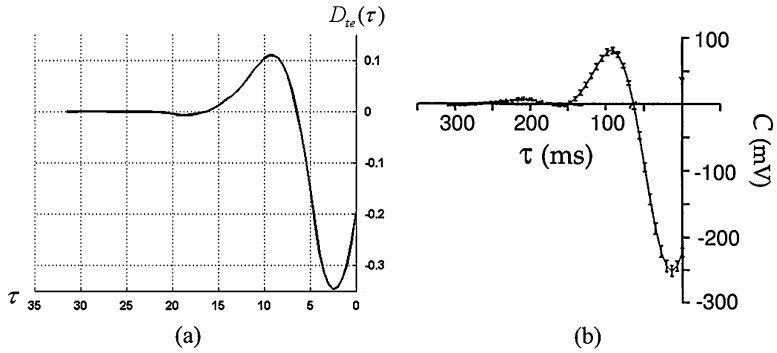

Fig. 6. (a) Temporal profile of a neuron trained by our developmental algorithm (the neuron enclosed by the black hollow square in Fig. 5). For convenience of comparison, we cut the tail (τ from 32 to 63) of the simulating temporal profile off and only plot a piece of this temporal profile (τ from 0 to 32). (b) Temporal profile of a real neuron measured from an electric fish. (Plot (b) is adapted from [50].)

TABLE II
PARAMETERS OF THE NEURONAL CLUSTER FOR SIMULATING CANS

The number of neurons (n)	16×16	Temporal window width (N)	64
Topological neighborhood	rectangular	$\sigma_{sp}(0)$ in (13)	$10 \times d_{unit}$
Input area (ψ)	15×15(pixel)	λ in (13)	5000
Input stimuli	chromatic	δ_{te} in (22)	13

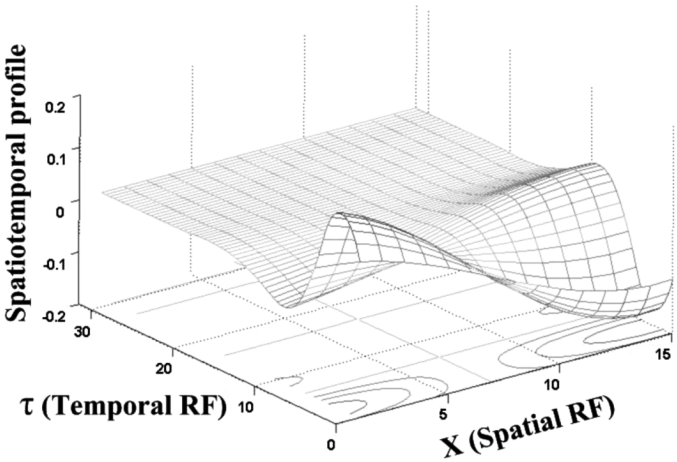

Fig. 7. Profile of the spatio–temporal weights of a neuron trained by our developmental algorithm (the neuron enclosed by the black hollow square in Figs. 4 and 5). It shows a reversed profile over both space and time, and indicates that the neuron has the preference for changes of stimuli over space and time.

$$i = 1, 2, \ldots, n. \quad (23)$$

Actually, (23) and (18) are almost the same in mathematical form. The only difference between them is the input signals [$\mathbf{S}(t+1)$ for (18); $\mathbf{S}^i_{\mathrm{te}}(t+1)$ for (23)]. This unification of the learning rules is based on a biologically plausible speculation: different cells in the visual cortex have almost the same learning rule, and the variety of features detected by the neurons is caused by the diversity of environments.

We have mentioned that the development of cells is probably controlled by the cell-centered mechanisms. That means the learning rule is supposed to be "*in-place*." By "*in-place*" learning, we mean that each neuron deals with its own development through its internal biological mechanisms and the interactions with the outside environment and its neighboring neurons. "*In-place*" learning is a better choice for developmental systems because of its simplicity, lower space and time complexities, and biological plausibility. It is clear that our learning rule is totally "*in-place*" learning. Besides, the competition between neurons enables them to detect different features. In addition, the lateral inhibitory connections also assist our model to generate topographic maps of RF that resemble the ordered arrangement of the orientation columns in V1 area. The "*in-place*" learning and the topographic map enable our model to be easily applied to computer vision [49].

V. SIMULATIONS FOR DEVELOPMENT OF SPATIO–TEMPORAL WEIGHTS

In this section, we present some simulation results of the neuronal cluster model with the developmental algorithm given in Section IV. In the first simulation, we train our model with gray scale natural videos to produce topographic maps of the orientation selective spatial weights and the reversed temporal weights. In the second simulation, the proposed model is situated in a special visual environment that contains only vertical gratings. In the last simulation, we try to train the proposed model in an environment of random chromatic sinusoidal gratings by using the same developmental rule and produce color-opponent patterns that approximate the RFs of chromatic antagonistic neurons. This result is related to applications in our attention selection system.

A. Simulation 1: Training Results in a Natural Video Environment

1) Initialization of the Neuronal Cluster: Some parameters of the neuronal cluster in Fig. 3 are provided in Table I, where

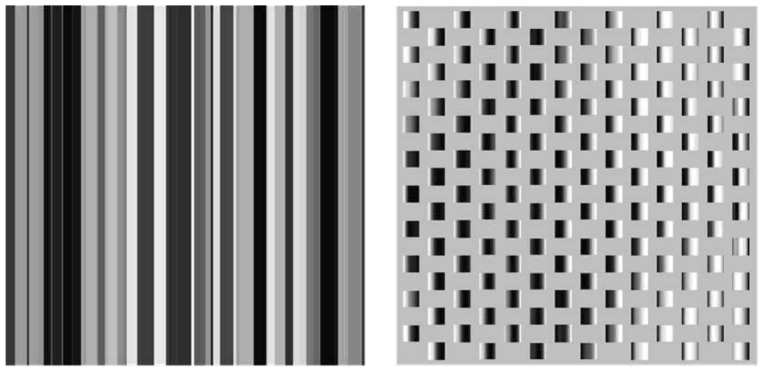

Fig. 8. Special visual environment in which only vertical gratings appear and the corresponding developmental result of spatial weight vectors for 140 neurons. (a) A large image with only vertical gratings for training. (b) The developmental results of spatial weight vectors when all training images are taken from the large picture shown in (a).

we randomly initialize the 64-dimensional temporal weight vectors. In order to speed up the convergence, in this simulation, we simply choose the first several samples to initialize the 225-dimensional spatial weight vectors, which guarantee that the initial spatial weight vectors are within the manifold of our training set.

2) Training Data: The training database is derived from eight gray scale natural video sections taken from the database provided by Hateren and Ruderman [35]. For each 8-b 128×128 pixels video section consisting of 9600 (50 f/s $\times$ 192 s) frames, we place onto it ten 15×15 pixels windows to get a video that consists of 96 000 frames. We then string the eight resulting videos into one long 15×15 pixels video consisting of 768 000 frames and use it to train the neuronal cluster model.

3) Results of the Developmental RFs: Fig. 4 shows the resulting spatial weight vectors for the 140 neurons in the trained neuronal cluster. Each subsquare image represents the 2-D visualization of the spatial weights of one neuron, which looks like a low-frequency Gabor function. It can be seen from Fig. 4 that all the neurons in the trained model have their respective preferred orientations. They are orderly arranged to form an almost complete set of spatial orientations, which is reasonable since the input video consists of nature images that cover all possible orientations. The developmental spatial weight vectors always adapt themselves to extract effective features despite the changing visual environment. The neighborhood relation between neurons makes these weights arranging in order.

Fig. 5 shows the temporal weights of these neurons. Each subimage in Fig. 5 is composed of a series of vertical gratings with the same intensity along columns. They are generated to display the 1-D data of the temporal weights in 2-D plane. The x-axis of each subimage refers to the time delay τ from 0 to $N-1$ (from right to left). The intensity of a vertical grating in a subimage corresponds to the value of the temporal weight at the given τ. The curve of temporal profile of the neuron circled by the black square in Fig. 5 is plotted in Fig. 6(a). Fig. 6(b) shows the profile of the real temporal weights measured from a visual neuron of an electric fish [50]. Comparing the simulation result with the real temporal weights, we can see that they have very similar profiles.

To display the spatio–temporal weight profile $D(x, y, \tau)$ in a space–time plot rather than the independent spatial profile and temporal profile, we ignore the weight's change along the y-axis and plot only the projection of the spatio–temporal weights on the $x - \tau$ plane. Fig. 7 shows the x-τ plot of the spatio–temporal weights of a selected neuron (the neuron circled by the black hollow square in Figs. 4 and 5). This neuron is supposed to be separable. As the value of τ increases, the pattern of the spatio–temporal weights turns from ON/OFF to OFF/ON and then flatten out. The reversed spatio–temporal profile indicates a strong preference for the changes of the stimuli. For example, in a case when a group of vertical gratings is moving horizontally across this neuron's RF, as they move in, the stimuli on this neuron's RF may have a pattern that the left is dark and the right is light (left-OFF/right-ON); however, as the vertical gratings move out, the stimuli may show a totally reversed pattern that the left is light and the right is dark (left-ON/right-OFF). Considering the reversed spatio–temporal profile in Fig. 7, it can be deduced that the neuron with such a weight configuration will have a strong response to moving edges in the x-direction. Furthermore, the result in Fig. 7 is also consistent with some biological experimental results [8].

B. Simulation 2: Training Results in a Special Visual Environment

In this simulation, the model is trained with some subimages taken from a large image which displays exclusively vertical gratings, as shown in Fig. 8(a). All parameters of the model are initialized as we did in Section V-A. The simulation results display only vertical orientation with different phases, as shown in Fig. 8(b). It means that the model always adapts to extract popular features and gives a better representation for the given environment. The simulation results are also consistent with the work by Blakemore and Cooper [12].

C. Simulation 3: Training Results With Chromatic Sinusoidal Grating Images

For proving the validity of the neuronal cluster model in color environment, we design this simulation to train the proposed model with images of chromatic sinusoidal gratings and try to

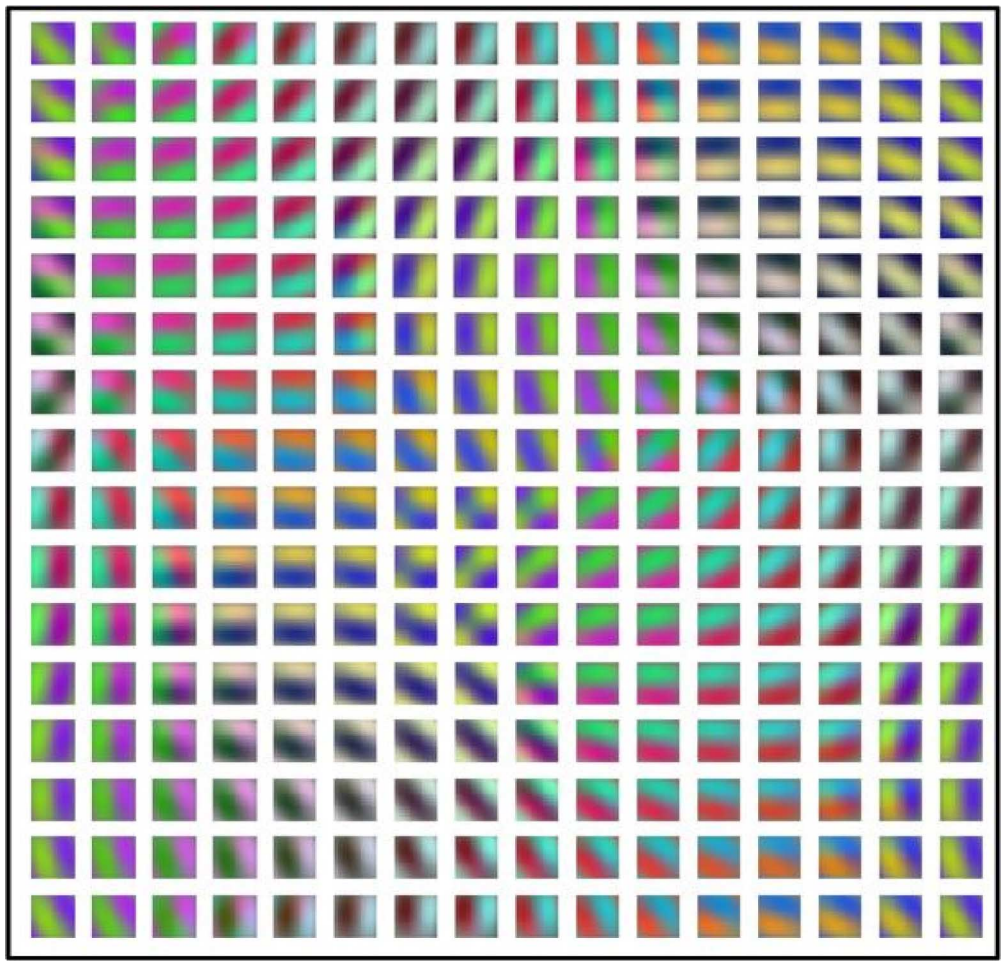

Fig. 9. Spatial weight vectors of 256 CANs trained by chromatic sinusoidal grating images.

develop color-opponent sensitive RFs that can be applied to our attention selection system in Sections VI and VII.

1) Initialization of the Neuronal Cluster: It has been discovered that three kinds of cone receptor cells in human retina are able to extract three different colors [red, green, and blue (RGB)] from light stimuli. So we use RGB primary colors as the inputs, then each input vector over a $l \times l$ pixels RF is $\mathbf{S}(t)$, where $\mathbf{S}(t) \in R^{3l^2}$, since each pixel contains three color components. The spatial weight vector is $\mathbf{D}^i_{\rm sp}(t) = [(\mathbf{D}^i_{\rm spr}(t))^T, (\mathbf{D}^i_{\rm spg}(t))^T, (\mathbf{D}^i_{\rm spb}(t))^T]^T \in R^{3l^2}$ accordingly, where $\mathbf{D}^i_{\rm spr}(t), \mathbf{D}^i_{\rm spg}(t), \mathbf{D}^i_{\rm spb}(t) \in R^{l^2}$ are the spatial weight vectors for the three different color components. Thus, the entire spatial weight vector $\mathbf{D}^i_{\rm sp}$ and the temporal weight vector $\mathbf{D}^i_{\rm te}$ of neuron C_i can be initialized using the same way as in Section V-A. The other parameters of this neuronal cluster are given in Table II.

2) Training Color Images: In natural scenes, only a very small proportion of the data contains the edges between different colors. Some preprocessing should be considered in color channels. For simulating the preprocessed natural images, 50 000 frames of 15×15 pixels color images (RGB) of sinusoidal gratings are generated to build a training database which is abundant in colors, spatial orientations, and phases.

3) Simulation Results: We use the same method presented in Sections III and IV to simulate the development of chromatic antagonistic neurons (CANs). The training results of the spatial weights of CANs are shown in Fig. 9. Although the training images have different colors, random orientations, and phases, the resulting patterns are similar to the color-opponent RFs (red/green and yellow/blue antagonistic weights). These RFs have a topographic arrangement, which indicates that the resulting networks of CANs can be applied as good feature detectors to visual attention selection.

VI. Visual Attention Selection System

Visual attention selection provides an effective way to detect the region of interest from a complex visual scene, which is critical for later visual processing to focus on only the stimuli that relate to the interesting objects. When we browse the real world without any given task, we are generally attracted by salient regions such as color or intensity contrasts (e.g., edge) and distinct features that differ from their surroundings [36], [38], [39], [51], [52]. We call this mechanism "task-independent attention selection." It has been shown from the simulations in Sections V-A and V-C that the trained neurons in the neuronal cluster have their respective preferred orientations and

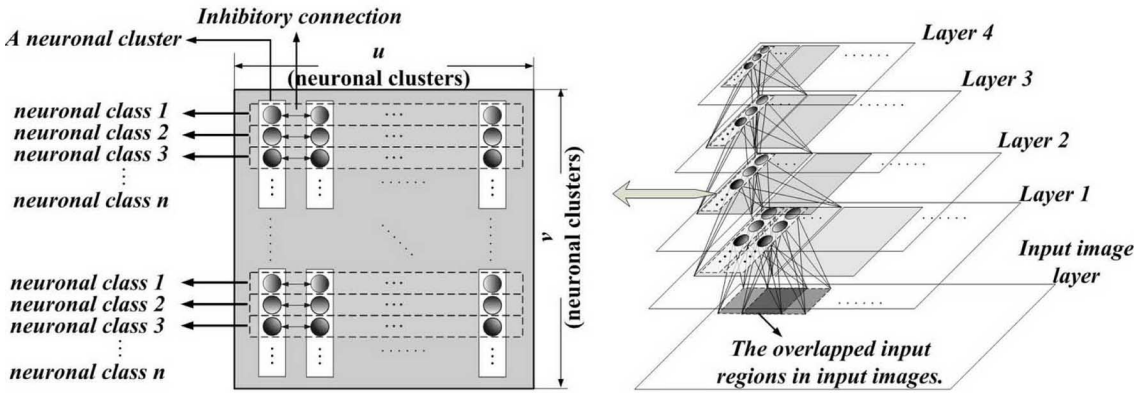

Fig. 10. Structure of MBSN. This is a four-layer network composed of many neuronal clusters as shown on the right-hand side. The left-hand side is an amplified picture of layer 2 that includes V × U neuronal clusters arranged in 2-D plane, where each white vertical rectangle (plotted by solid lines) with many circles inside it represents one neuronal cluster. Each circle in a neuronal cluster is a neuron that detects one type of features. Each neuronal class is a set of neurons enclosed by the horizontal rectangle (plotted by dashed lines). Neurons in the same neuronal class may be located in different neuronal clusters, but detect the same feature.

reversed temporal profiles, which form the topographic maps. That means each neuron has a preference for changes of the stimuli over both space and time, namely, they are capable of detecting edges and motion. In this section, we use many neuronal clusters to construct a multilayer network that simulates attention selection of human vision.

A. Structure of Multilayer Bottom-Up Sensory Network

Using the neuronal cluster presented in this paper, a multilayer bottom-up sensory network (MBSN) is constructed to simulate visual attention selection. MBSN has four layers as shown on the right-hand side of Fig. 10. Every layer consists of many neuronal clusters. Each neuronal cluster has n neurons that share a common input area in the previous layer, as shown in Figs. 3 and 10. It can be seen from the simulation results in Figs. 4, 5, and 9 that n neurons in a trained neuronal cluster can detect n different features, such as various oriented edges, color-opponent pattern, and motion. The topographic arrangement enables their adjacent neurons to detect similar features.

To simplify the training stage, we hypothesize that neuronal clusters in the same layer of the MBSN are the same as each other. In consequence, two neurons in different neuronal clusters may detect the same feature. Thus, we define all the neurons which have the same pattern of spatio–temporal weights and detect the same feature as a neuronal class, as shown on the left-hand side of Fig. 10. If there are n neurons in each neuronal cluster, then there is a total of n neuronal classes. The neuron that belongs to the qth class of the pth cluster in the uth layer is denoted as $C_{p,q,u}$.

The input to each neuronal cluster of the first layer is an $l \times l$ pixels area of the input image, which is exactly the neuronal cluster's RF. RFs of the neighboring neuronal clusters in the first layer overlap each other, as shown on the right-hand side of Fig. 10. However, the input areas of the neighboring neuronal clusters in other layers except the first layer do not overlap each other, and so the upper layers have larger RFs on the original input image, which essentially provides each layer with a scaling effect on the input image. This structure enables the MBSN to extract features in various scales.

B. Signal Processing in MBSN

In Section III, we have discussed the response function (9)–(11) of a neuron inside a single neuronal cluster. In (9), only the lateral inhibition inside the neuronal cluster is considered. In the MBSN, however, we have to take into account the lateral inhibition between neighboring neuronal clusters. To simplify the complex recurrent computation in the MBSN, we hypothesize that bottom-up stimuli and inhibitory signals from neighboring neuronal clusters to a neuron in layers except the first layer come from only the neurons belonging to the same class. Let us denote by $r_e^{p,q,u''}(t)$ the response of the neuron $C_{p,q,u}$ in the MBSN, which can be rewritten as

$$r_e^{p,q,u''}(t) = f\left(r_e^{p,q,u'}(t) - \sum_{C_{p,q',u} \in A_{p,q,u}} h^{p,q,u}(i,p,k) r_e^{p,q',u'}(t)\right) \tag{24}$$

where $A_{p,q,u}$ is the neighboring area of $C_{p,q,u}$. $r_e^{p,q,u'}(t)$ [computed by (11)] is the response of neuron $C_{p,q,u}$ with only the lateral inhibition inside the neuronal cluster. $r_e^{p,q,u''}(t)$ is the response of $C_{p,q,u}$ when the interactive inhibition between the neighboring neuronal clusters is considered. The second term of the function $f(\cdot)$ in (24) is the summation of all the inhibitory signals that $C_{p,q,u}$ receives from its neighboring clusters (only these neurons of the same class are taken into account). Let us define the function $f(\cdot)$ in (24) as $f(x) = \max(x, 0)$, then the responses of all neurons in the MBSN always take on nonnegative values. If x is negative, the neuron will be suppressed completely by its adjacent neurons and stop firing. $h^{p,q,u}(p,v,u)$ denotes the inhibitory weight from the neuron $C_{p,q',u}$ to neuron $C_{p,q,u}$, which can be described by a standard difference of Gaussian (DOG) function [45]

$$h^{p,q,u}(p,q',u) = \alpha_e \exp\left(-\frac{(d^{p,q,u}(p,q',u))^2}{2(\sigma_e^u)^2}\right) - \alpha_h \exp\left(-\frac{(d^{p,q,u}(p,q',u))^2}{2(\sigma_h^u)^2}\right) \tag{25}$$

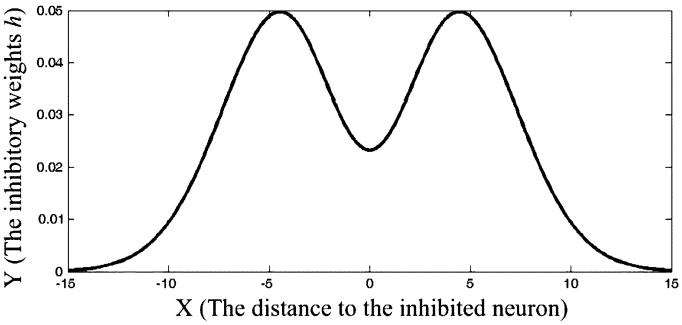

Fig. 11. Profile of the lateral inhibitory weights that models the biological measurement by Hartline [53].

where $d^{p,q,u}(p,q',u)$ is the distance between neuron $C_{p,q',u}$ and $C_{p,q,u}$; $\alpha_e, \alpha_h, \sigma_e^u$ and σ_h^u are the system parameters which determine the shape of the DOG function. The profile of $h^{p,q,u}(p,q',u)$ is plotted in Fig. 11 with $\alpha_e = 0.125, \alpha_h = 0.114, \sigma_e^u = 4, \sigma_h^u = 3.17$. The x-axis denotes the distance $d^{p,q,u}(p,q',u)$, and the y-axis denotes the value of the inhibitory weight $h^{p,q,u}(p,q',u)$. The profile of the lateral inhibitory weights given in Fig. 11 models the biological measurement in [53].

Here we can use the proposed developmental rules to learn all the spatio–temporal weights of neuronal clusters on all layers. To simplify the computation in the practical application, the feedforward connections of higher layers (in this case, the higher layers are all the layers except the first layer) are designed to make the neuron's response proportional to the average amplitude of the outputs from the previous layer, but the lateral inhibition is still preserved. So, the response function of neurons in higher layers can be written as

$$r_e^{p,q,u\prime}(t) = \beta^u \left\langle r_e^{p,q^\star,u-1\prime\prime}(t) \right\rangle, \quad \text{for all } C_{p,q^\star,u-1} \in \Psi^{p,q,u} \tag{26}$$

where $\Psi^{p,q,u}$ is the input spatial area of the neuron $C_{p,q,u}$, and β^u is a parameter whose typical value is 1. $\langle\rangle$ is an operator to compute the mean value of $r_e^{p,q^\star,u-1\prime\prime}(t)$ for all $C_{p,q^\star,u-1} \in \Psi^{p,q,u}$.

C. Focus of Attention

For many existing task-independent attention systems [36]–[39], a strategy called "winner-take-all" (WTA) is adopted, which means the focus of attention (FOA) is always located in the region with the strongest response to the input scene. By the WTA rule, the FOA will be always located in the RF of the current winning neuron in the MBSN.

In the MBSN, however, only the bottom-up signals and the lateral inhibitory signals are considered. If the testing input is a still image, the FOA will always stay in the region of the first winning neuron's RF. To prevent the FOA from getting stuck on one region, a top–down inhibitory signal from the higher level cortex is necessary, which simulates the habituation process of associative learning modeled in psychology. We may assume that the value of the top–down inhibitory signal is decided by the "novelty" of an interesting region.

We define the neuronal cluster that contains the winning neuron as the "top cluster." If the winning neuron at time t is $C_{p,q,u}$, the responses of all neurons in current top cluster can be characterized as an n-dimensional vector $\mathbf{R}_{\text{top}}(t) = [r_e^{1,q,u\prime\prime}(t), r_e^{2,q,u\prime\prime}(t), \ldots, r_e^{n,q,u\prime\prime}(t)]$, where n is the amount of neurons in this top cluster. The novelty θ can be computed by

$$\theta = 1 - \frac{\mathbf{R}_{\text{top}}(t)^T \cdot \mathbf{R}_{\text{top}}(t-1)}{\|\mathbf{R}_{\text{top}}(t)^T\| \cdot \|\mathbf{R}_{\text{top}}(t-1)\|}. \tag{27}$$

In (27), if $\mathbf{R}_{\text{top}}(t)$ and $\mathbf{R}_{\text{top}}(t-1)$ are almost the same, the novelty θ is close to zero. That means the novelty θ, in essence, is the difference between current interesting region and previous interesting region. A small θ indicates that the current interesting region is an object that the system had seen before. So at time $(t+1)$, the winning neuron $C_{p,q,u}$ will receive a top–down inhibitory signal $I_{\text{td}}^{p,q,u}(t+1)$ that makes the neuron $C_{p,q,u}$ ineligible to compete for the current winner. According to the WTA rule, the FOA will jump onto another neuron with the strongest response.

Here the top–down inhibitory signal $I_{\text{td}}^{p,q,u}$ is a decreasing function of the novelty θ. That means the system tends to focus on novel objects and spend less time on familiar things. To prevent the neuron $C_{p,q,u}$ from attracting the FOA back immediately, the top–down inhibitory signal $I_{\text{td}}^{p,q,u}$ should last for a while. In this period, $I_{\text{td}}^{p,q,u}$ decreases slowly as time increases. The eligibility for winning returns to neuron $C_{p,q,u}$ when $I_{\text{td}}^{p,q,u}$ approaches zero, which means the system no longer recalls that it saw the corresponding region of $C_{p,q,u}$ before.

VII. Simulation of Attention Selection

Using the MBSN, we can realize the attention selection for both static and dynamic scenes. In this section, three simulations are designed to test the performance of the MBSN. The first simulation contains simple video and images that test the MBSN's ability to detect salient features. The second simulation tests attention selection in a complex natural image, and the last simulation tests attention selection in a video of both static and dynamic scenes.

A. Detection of Salient Regions

For human vision, the FOA generally tends to locate in regions with edges, motion, and other salient features [54]–[56]. By salient feature, we mean the feature that differs from its surroundings. For example, consider one red flower and many yellow flowers in a green lawn. In general, the red flower draws more attentions of human vision. The similar effect can be realized by the MBSN. Fig. 11 shows four testing inputs and their corresponding responses of the MBSN. The system parameters are $\alpha_e = 0.125, \alpha_h = 0.114, \sigma_e^u = 4, \sigma_h^u = 3.17$, and $\beta^u = 1.5$. The intensity of each location in the response maps is proportional to the largest response of the neuron in the corresponding neuronal cluster.

Fig. 12(a) is a video frame in a static scene and Fig. 12(b) is a video frame in a dynamic scene (moving ball and little toy in hand). Both the frame sizes of Fig. 12(a) and (b) are 320×240. The input area of each neuronal cluster in the first layer covers 15×15 pixels. The shift of input regions

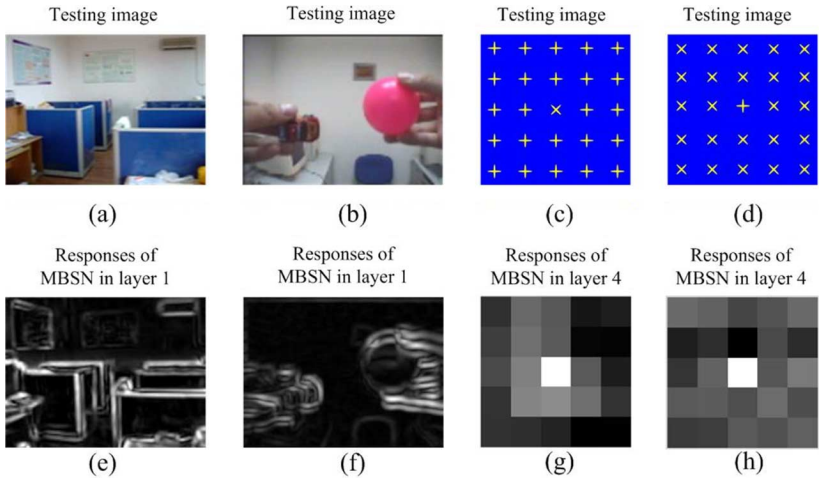

Fig. 12. Four testing inputs and the corresponding response maps of the MBSN. (a) A video frame in a static scene (320 × 240). (b) A video frame in a dynamic scenes (320 × 240), in which the ball and the little toy are moved by the hands. (c) A testing image with the salient feature "×" surrounded by the common features "+." (d) A testing image with the salient feature "+" surrounded by "×." (e)–(h) are the corresponding responses of the MBSN to the testing images (a)–(d), respectively.

Fig. 13. The 600 × 800 color image for testing attention selection using the MBSN.

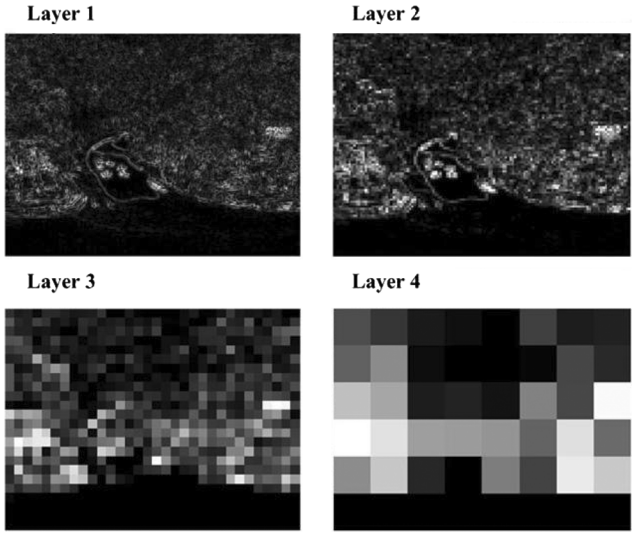

Fig. 14. Response maps of four layers in the MBSN to the testing image in Fig. 13. Each pixel corresponds to a neuronal cluster. The intensity of each pixel is proportional to the response amplitude of the neuron that has the strongest response in this neuronal cluster.

between any two neighboring neuronal clusters is one pixel (along the x-axis or the y-axis or both of them). To extract features from the border regions of the input image, all border regions are extended symmetrically. Consequently, the first layer of the corresponding MBSN also contains 320 × 240 neuronal clusters. Fig. 12(e) and (f) gives the corresponding responses of the neurons in the first layer when the testing inputs are Fig. 12(a) and (b), respectively. According to Fig. 7, the reversed shape of the spatio–temporal weights indicates that the neuron has special preference for the contrast of stimuli over time and space. Fig. 12(e) and (f) shows that the MBSN generates stronger responses in regions with edges and motion.

In the tests for Fig. 12(c) and (d), the resolution of the input images is 320 × 320 pixels, and the input area of each neuron in layers except layer 1 covers 4 × 4 neuronal clusters in its previous layer. Thus, layer 4 of the MBSN consists of 5 × 5 neuronal clusters $(320/(4^3) = 5)$. The centers of these two testing images have the salient features which are different from their surroundings. The responses of layer 4 of the MBSN to these two testing images are shown in Fig. 12(g) and (h). We can see that the salient features cause stronger responses, which results from lateral inhibitory signals between the same neuronal classes of neurons in the MBSN. This phenomenon can be explained by the example in Fig. 12(d), where the neuron of class A detecting the feature "+" in the center of layer 4 is not suppressed by its neighbors because there are no responses from neurons of class A in its surroundings, and all the neurons of class B detecting features "×" over the input image are suppressed by each other because features "×" are uniformly arranged in the testing image. Thus, the neuron of class A in the center of layer 4 has the strongest response. The same reason holds for testing image Fig. 12(c) and its response map Fig. 12(g).

Fig. 15. Trajectory of the FOA for Fig. 13. For distinguishing the trajectory from the background, the lines are plotted in white and black. The hollow squares record the RF region of the neuronal cluster where the FOA is located.

B. Movement of FOA for Complex Color Image

In this experiment, the testing image is a large static natural color image as shown in Fig. 13. All the system parameters of the MBSN are chosen to be the same as in Section VII-A. The responses of four layers in the MBSN are shown in Fig. 14. We can see that the model extracts features in various scales, such as the strong contrast of color and intensities.

In Fig. 15, the black and white line segments depict the trajectory of the FOA through 55 located points. In this experiment, all the winning neurons chosen by the MBSN are found in layers 2 and 3. The reason is that there are no salient objects in the scene that match the RF's size of the neurons in layer 1 and layer 4. Thus, the neurons in layers 1 and 4 are not strong enough to be the winner. So only two types of marks (the smaller hollow square is for layer 2 and the larger hollow square is for layer 3) are shown in Fig. 15. The black and white numbers denote the order of the FOA's saccades. It appears that the FOA tends to be located in the salient regions such as the boats, the beach umbrella, the rocks, and the blue board in the mountain. This indicates that the MBSN is capable of detecting salient features.

C. Movement of FOA for Dynamic Video

The reversed temporal profile enables the neuron to detect motion. Therefore, the proposed MBSN should be able to realize visual attention for not only static scenes but also dynamic scenes. Fig. 16 shows the trajectories of the FOA in several sample frames of the video for a moving toy. The scenes before frame 93 are static where the FOA moves to the salient regions, such as the tea cup, the characters on the books, the keys, and the ink bottle. After the humanoid toy moves in, the system attends to the moving toy (frame 101–241). When the toy moves into a region (in front of the black book) in which the background has the same color (black) as the toy, the motion of the toy is not sufficiently salient. Thus, the FOA jumps from the toy to other regions (the characters on the book and the tea cup in frames 181 and 191). As the toy moves out of the scene, the video becomes static again and the FOA start again to shift between those static salient regions of the scene (frames 251–291). Fig. 16 shows that the MBSN can perform intuitively for both static and dynamic scenes using the same developed system.

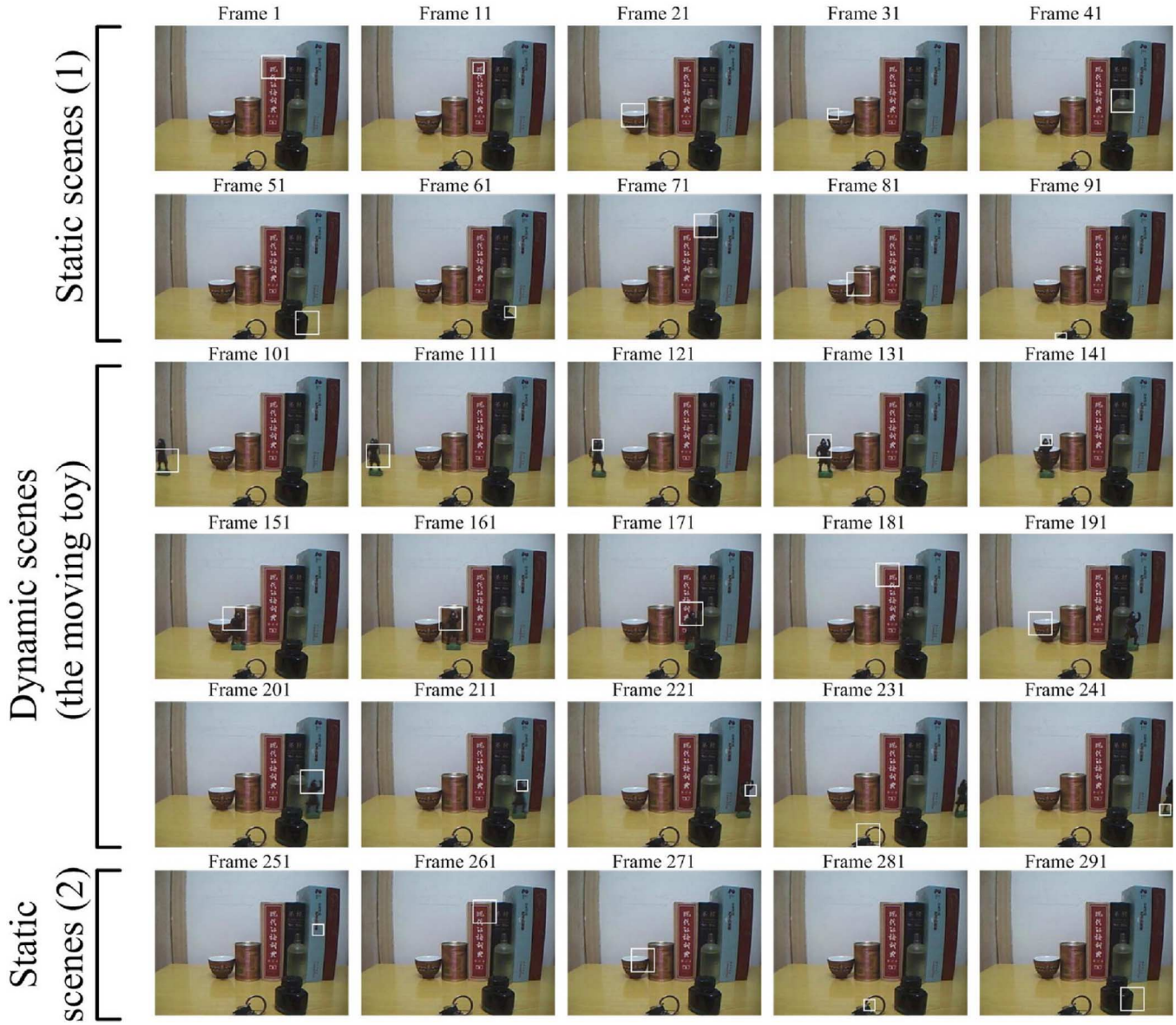

Fig. 16. Trajectory of the FOA for a testing color video. In the static scenes (1) and (2), the system is focused on the objects with salient features. When a toy moves in, the FOA is attracted by the moving object.

VIII. Conclusion and Discussion

In the human brain, neurons at different cortical locations have different spatial and temporal properties, and these properties change with time according to experience. However, the formation and adaptation of these properties are guided by developmental rules and experience. The work presented here proposes a model called the neuronal cluster to simulate the development of weights for RFs of visual neurons in V1 area, which can process both spatial and temporal information. The presented algorithm indicates that a simple and unified in-place learning rule is possible for both adaptations of spatial weights, temporal weights, and chromatic weights. Motivated by the biological in-place learning principle, the proposed learning rule is based on the Hebbian rule and lateral inhibition, and is without need for extra storage and computations of the covariance matrix (or other forms of moment matrix of second order or higher) of the neuronal inputs or the partial derivatives with respect to all neuronal weights.

We further hypothesize that the temporal weights of the neurons in the cortical area are enabled by delayed lateral inhibition among neighboring neurons. We also give a mathematical deduction for describing the relationship between the temporal weights and the lateral inhibitory weights. That gives a strong support to unifying the adaptations of both spatial weights and temporal weights using the same learning rule.

Using the proposed neuronal cluster and the developmental rule, we simulate the development of spatio–temporal weights using a natural video. The typical spatial orientation selective neurons with topographic arrangement and temporal reversed profile are generated, and some typical chromatic antagonistic neurons are also generated by training the model in the environment of chromatic sinusoidal gratings. These results are supported by biological facts.

To demonstrate the application of such developed spatio–temporal network, we use many neuronal clusters to construct a MBSN, which can implement visual attention selection for both static and dynamic scenes. The MBSN model is different from other existing systems, because it uses a unified in-place developmental rule to generate the weights that can detect all types of features (color, edge, motion, etc.).

Although the proposed model has the above merits, for biologically plausible development, we still need to do more research. A limitation is that our model cannot produce multiple ripples and sparse RFs as ICA-type models do. The effort to find a multilayer model with a simple in-place learning rule that models both spatial RF and temporal RF development and to make the spatio–temporal weights more similar to visual coding that can extract both sparse, localized topographic, and complete features is our further work. Another limitation is that in this paper we only use the artificial input as training set to develop chromatic antagonistic weights for the convenience of applications. In the future, we need to build a preprocessing filter in our model for natural color images and video streams, and apply the improved model to perform more tasks in computer vision, such as object detection and recognition. Also, eye movement and attention modulation studies may be conducted using the proposed model. Recently, top–down attention mechanisms, both in position or object type, called where–what network, have been proposed and experimented with by Weng *et al.* [57].

ACKNOWLEDGMENT

The authors would like to thank J. H. van Hateren and D. L. Ruderman for the database of dynamic videos. They would also like to thank P. Bian for his help in proofreading and editing.

REFERENCES

[1] H. K. Hartline, "The response of single optic nerve fibers of the vertebrate eye to illumination of the retina," *Amer. J. Physiol.*, vol. 121, no. 22, pp. 400–415, 1938.

[2] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *J. Physiol.*, vol. 160, no. 1, pp. 106–154, 1962.

[3] S. Marcelja, "Mathematical description of the response of simple cortical cells," *J. Opt. Soc. Amer.*, vol. 70, no. 11, pp. 1297–1300, 1980.

[4] J. G. Daugman, "Tow-dimensional spectral analysis of cortical receptive field profiles," *Vis. Res.*, vol. 20, no. 5, pp. 847–856, 1980.

[5] D. J. Heeger, "Modeling simple-cell direction selectivity with normalized, half-squared, linear operators," *J. Neurophysiol.*, vol. 70, no. 5, pp. 1885–1898, 1993.

[6] C. Rasche, "Neuromorphic excitable maps for visual processing," *IEEE Trans. Neural Netw.*, vol. 18, no. 2, pp. 520–529, Mar. 2007.

[7] J. K. Stevens and G. L. Gerstein, "Spatiotemporal organization of cat lateral geniculate receptive fields," *J. Neurophysiol.*, vol. 39, no. 22, pp. 213–238, 1976.

[8] G. C. DeAngelis, I. Ohzawa, and R. D. Freeman, "Receptive field dynamics in the central visual pathway," *J. Neurophysiol.*, vol. 18, no. 10, pp. 451–458, 1995.

[9] E. H. Adelson and J. R. Bergen, "Spatiotemporal energy models for the perception of motion," *J. Opt. Soc. Amer.*, vol. A2, no. 2, pp. 284–299, 1985.

[10] Y. J. Wang, X. L. Qi, J. Xing, and D. S. Yu, "Extended Gabor function model and simulation of some characteristic curves of receptive fields," *Scientia Sinica Seriess B*, vol. 31, no. 1, pp. 1185–1994, 1988.

[11] Y. J. Wang, X. L. Qi, J. Xing, and Y. Z. Chen, "Simulation of RF dynamics," *Trends Neurosci.*, vol. 19, no. 1, pp. 385–386, 1996.

[12] C. Blakemore and G. Cooper, "Development of the brain depends on the visual environment," *Nature*, vol. 228, no. 5270, pp. 471–478, 1970.

[13] T. Hosoya, S. A. Baccus, and M. Meister, "Dynamic predictive coding by the retina," *Nature*, vol. 436, no. 7047, pp. 71–77, 2005.

[14] S. L. Pallas, L. von Melchner, and M. sur, "Visual behavior mediated by retinal projections directed to the auditory pathway," *Nature*, vol. 404, no. 6780, pp. 871–876, 2000.

[15] T. D. Sanger, "Optimal unsupervised learning in a single-layer linear feedforward neural network," *Neural Netw.*, vol. 2, no. 6, pp. 459–473, 1989.

[16] R. Linsker, "From basic network principles to neural architecture: Emergence of spatial-opponent cells," *Proc. Nat. Acad. Sci. USA*, vol. 83, no. 19, pp. 7508–7512, 1986.

[17] P. Foldiak, "Adaptive network for optimal linear feature extraction," in *Proc. IEEE/INNS Int. Joint Conf. Neural Netw.*, 1989, pp. 401–405.

[18] E. Oja, "A simplified neuron model as a principal component analyze," *J. Math. Biol.*, vol. 15, no. 1, pp. 267–273, 1982.

[19] H. G. Barrow, "Learning receptive fields," in *Proc. IEEE 1st Int. Conf. Neural Netw.*, 1987, vol. 4, no. 1, pp. 115–121.

[20] J. Weng and N. Zhang, "Optimal in-place learning and lobe component analysis," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 16–21, 2006, pp. 3887–3894, 2006.

[21] T. Kohonen, "The adaptive-subspace SOM (ASSOM) and its use for the implementation of invariant feature detection," in *Proc. Int. Conf. Artif. Neural Netw.*, Paris, France, 1995, pp. 3–10.

[22] H. C. Zheng, G. Lefebvre, and C. Laurent, "Fast-learning adaptive-subspace self-organizing map: An application to saliency-based invariant image feature construction," *IEEE Trans. Neural Netw.*, vol. 19, no. 5, pp. 746–757, Sep. 2008.

[23] T. Kohonen, *Self-Organizing Map*, 2nd ed. New York: Wiley, 2001.

[24] D. O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*, 1st ed. New York: Wiley, 1949.

[25] D. J. Field, "What is the goal of sensory coding?," *Neural Comput.*, vol. 6, no. 1, pp. 559–601, 1994.

[26] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for nature images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[27] A. J. Bell and T. J. Sejnowski, "The "independent components" of natural scenes are edge filters," *Vis. Res.*, vol. 37, no. 23, pp. 3327–3338, 1997.

[28] R. Linsker, "A local learning rule that enables information maximization for arbitrary input distributions," *Neural Comput.*, vol. 9, no. 8, pp. 1661–1665, 1997.

[29] M. D. Plumbley and E. Oja, "The "nonnegative PCA" algorithm for independent component analysis," *IEEE Trans. Neural Netw.*, vol. 15, no. 1, pp. 66–76, Jan. 2004.

[30] M. S. Falconbridge, R. L. Stamps, and D. R. Badcock, "A simple hebbian/anti-Hebbian network learns the sparse independent components of natural images," *Neural Comput.*, vol. 18, no. 1, pp. 415–429, 2006.

[31] J. Karhunen and S. Malaroiu, "Locally linear independent component analysis," in *Proc. Int. Joint Conf. Neural Netw.*, 1999, pp. 882–887.

[32] 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, no. 1, pp. 150–159, 2008.

[33] A. Hyvärinen, P. O. Hoyer, and M. Inki, "Topographic independent component analysis," *Neural Comput.*, vol. 13, no. 7, pp. 1527–1558, 2001.

[34] L. Ma and L. Zhang, "A hierarchical generative model for overcomplete topographic representations in natural images," in *Proc. Int. Joint Conf. Neural Netw.*, Orlando, FL, Aug. 12–17, 2007, pp. 1198–1203, 2007.

[35] J. H. Hateren and D. L. Ruderman, "Independent component analysisi of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex," *Proc. R. Soc. Lond. B.*, pp. 2315–2320, 1998.

[36] C. Koch and S. Ullman, "Shifts in selective visual attention: Towards the underlying neural circuitry," *Human Neurobiol.*, vol. 4, no. 4, pp. 219–227, 1985.

[37] J. K. Tsotsos, S. M. Culhane, W. Y. Kei Wai, Y. Lai, N. Davis, and F. Nuflo, "Modeling visual attention via selective tuning," *Artif. Intell.*, vol. 78, no. 1, pp. 507–545, 1995.

[38] L. Itti and C. Koch, "A saliency-based search mechanism for overt and covert shifts of visual attention," *Vis. Res.*, vol. 40, no. 10, pp. 1489–1506, 2000.

[39] L. Itti, "Quantitative modeling of perceptual salience at human eye position," *Vis. Cogn.*, vol. 14, no. 4, pp. 959–984, 2006.

[40] W. S. McCulloch and W. H. Pitts, "A logical calculus of ideas immanent in neural nets," *Bull Math Biophys.*, vol. 5, no. 1, pp. 115–133, 1943.

[41] C. Enroth-Cugell and L. Pinto, "Algebraic summation of centre and surround inputs to retinal ganglion cells of the cat," *Nature*, vol. 226, no. 5244, pp. 458–459, 1970.
[42] J. A. Movshon, I. D. Thompson, and D. J. Tolhurst, "Spatial summation in the receptive fields of simple cells in the cat's striate cortex," *J. Neurophysiol.*, vol. 283, no. 1, pp. 53–57, 1978.
[43] P. Z. Marmarelis and V. Z. Marmarelis, *Analysis of Physiological Systems: The White-Noise Approach*, 2nd ed. New York: Plenum, 1978.
[44] J. J. Hopfield and D. W. Tank, "Computing with neural circuits: A model," *Science*, vol. 233, no. 4764, pp. 625–633, 1986.
[45] P. Dayan and L. F. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*, 1st ed. Cambridge, MA: MIT Press, 2001.
[46] Z. H. Mao and S. G. Massaquoi, "Dynamics of winner-take-all competition in recurrent neural networks with lateral inhibition," *IEEE Trans. Neural Netw.*, vol. 18, no. 1, pp. 55–69, Jan. 2007.
[47] T. Yamada, "Control of tissue specificity: The pattern of cellular synthetic activities in tissue transformation," *Amer. Zoologist*, vol. 6, no. 1, pp. 21–31, 1966.
[48] T. S. Okada, *Transdifferentiation*, 1st ed. New York: Oxford Univ. Press, 1991.
[49] Z. Li, "A neural model of contour integration in the primary visual cortex," *Neural Comput.*, vol. 19, no. 4, pp. 480–498, 1999.
[50] F. Gabbini, W. Metzner, R. Wessel, and C. Koch, "From stimulus encoding to feature extraction in weakly electric fish," *Nature*, vol. 384, no. 6609, pp. 564–567, 1996.
[51] Y. Sun and R. Fisher, "Object-based visual attention for computer vision," *Artif. Intell.*, vol. 146, no. 1, pp. 77–123, 2003.
[52] S. Frintrop, *Vocus: A Visual Attention System for Object Detection and Goal-Directed Search*, ser. Lecture Notes in Computer Science, 1st ed. Berlin, Germany: Springer-Verlag, 2006, vol. 3889.
[53] H. K. Hartline, H. G. Wagner, and F. Ratliff, "Inhibition in the eye of limulus," *J. Gen. Physiol.*, vol. 39, no. 5, pp. 651–673, 1956.
[54] A. M. Treisman and S. Gormican, "Feature analysis in early vision: Evidence from search asymmetries," *Psychol. Rev.*, vol. 95, no. 1, pp. 15–48, 1988.
[55] H. E. Egeth and S. Yantis, "Visual attention: Control, representation, and time course," *Annu. Rev. Psychol.*, vol. 48, no. 1, pp. 269–297, 1997.
[56] J. Theeuwes, "Top-down search strategies cannot override attentional capture," *Psychonom. Bull. Rev.*, vol. 11, no. 1, pp. 65–70, 2004.
[57] Z. Ji, J. Weng, and D. Prokhorov, "Where-what network 1: "Where" and "what" assist each other through top-down connections," in *Proc. IEEE Int. Conf. Develop. Learn.*, Monterey, CA, Aug. 9–12, 2008, pp. 61–66, 2008.

Dongyue Chen received the B.S. and Ph.D. degrees from the Department of Electronic Engineering, Fudan University, Shanghai, China, in 2002 and 2007, respectively.

Currently, he is an Associate Professor at the School of Information Science and Engineering, Northeastern University, Shenyang, China. His research interests include biologically motivated visual model, computer vision, artificial intelligence, and pattern recognition.

Liming Zhang (M'88–SM'02) received the undergraduate degree in physics from Fudan University, Shanghai, China, in 1965.

From 1986 to 1988, she was a Visiting Scholar at the Electrical Engineering Department, University of Notre Dame, South Bend, IN. In 1996, she was a Senior Visiting Scholar at Munich Technology University, Munich, Germany. Currently, she is a full Professor and Doctoral Advisor at the Department of Electronic Engineering, Fudan University, and a Leader of Image and Intelligence Laboratory. Since 1986, she has been engaged in artificial neural network, machine learning, feature selection and pattern recognition of image and objects, including face recognition, brain-like robot, etc. Her group has accomplished more than ten projects supported by climbing program, national key project, natural sciences foundation, Shanghai Science and Technology Committee, etc. She has published more 120 papers in important national and international journals and conference proceedings concentrated on pattern recognition, machine learning, and neural networks.

Juyang (John) Weng (S'85–M'88–SM'05–F'09) received the B.S. degree from Fudan University, Shanghai, China, in 1982 and the M.S. and Ph.D. degrees from University of Illinois at Urbana-Champaign, Urbana, in 1985 and 1989, respectively, all in computer science.

Currently, he is 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. After a few years of research on structure and motion from image sequences, 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, abstract reasoning, and thinking 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 is the Editor-in-Chief of the *International Journal of Humanoid Robotics*, a member of Board of Governors of the International Neural Network Society, and an Associate Editor of the new IEEE TRANSACTIONS ON AUTONOMOUS MENTAL DEVELOPMENT. He was a Program Chairman of the NSF/DARPA-funded Workshop on Development and Learning 2000 (1st ICDL), the Chairman of the Governing Board of the International Conferences on Development and Learning (ICDL) (2005–2007), a General Chairman of 7th ICDL, Chairman of the Autonomous Mental Development Technical Committee of the IEEE Computational Intelligence Society (2004–2005), and an Associate Editor of the IEEE TRANSACTIONS ON PATTERN RECOGNITION AND MACHINE INTELLIGENCE and the IEEE TRANSACTIONS ON IMAGE PROCESSING.