Brain-Like Learning Directly from Dynamic Cluttered Natural Video

Yuekai Wang*†, Xiaofeng Wu*†, Juyang Weng‡§
*State Key Lab. of ASIC & System, Fudan University, Shanghai, 200433 China
†Department of Electronic Engineering, Fudan University, Shanghai, 200433 China
‡School of Computer Science, Fudan University, Shanghai, 200433 China
§Department of Computer Science and Engineering, Cognitive Science Program, Neuroscience Program,
Michigan State University, East Lansing, Michigan, 48824 USA
Email: 10210720110@fudan.edu.cn, xiaofengwu@fudan.edu.cn, weng@cse.msu.edu

Abstract—**It is mysterious how the brain of a baby figures out which part of a cluttered scene to attend to in the dynamic world. On one hand, the various backgrounds, where object may appear at different locations, make it difficult to find the object of interest. On the other hand, with the numbers of locations, types and variations in each type (e.g., rotation) increasing, conventional model-based search schemes start to break down. It is also unclear how a baby acquires concepts, such as locations and types. Inspired by brain anatomy, the work here is a computational synthesis from rich neurophysiological and behavioral data. Our hypothesis is that motor signals pay a critical role for the neurons in the brain to select the motor-correlated pattern on the retina to respond. This work introduces a new biologically inspired mechanism – synapse maintenance in tight integration with Hebbian mechanisms to realize object detection and recognition from cluttered natural video while the motor manipulates (or correlate with) object of interest. Synapse maintenance is meant to automatically decide which synapse should be active during the firing of the post-synaptic neuron. With the synapse maintenance, each neuron automatically wires itself with the other parts of the brain-like network even when a dynamic object of interest, specified by the supervised motor, takes up only a small part of the retina in the presence of complex dynamic backgrounds.**

I. Introduction

In the recent years, much effort has been spent on the field of artificial intelligence (AI) [1]. As the field of AI is inspired by human intelligence, more and more artificial intelligent models proposed are inspired by the brain to different degrees [2]. General objects recognition and attention is one of the important issues among the field of AI. And since human vision systems can accomplish such tasks quickly, mimicking the human vision systems is thought as one possible approach to address this open yet important vision problem.

In the primate vision system, two major streams have been identified. The ventral stream involving V1, V2, V4 and the inferior temporal cortex is responsible for the cognition of shape and color of objects. The dorsal stream involving V1, V2, MT and the posterior parietal cortex takes charge of spatial and motion cognition. Several cortex-like network models have been proposed. One Model is HMAX, introduced by Riesenhuber and Poggio [3]. It is based on hierarchical feed forward architecture similar to the organization of visual cortex. It analyzes the input image via Gabor function and builds an increasingly complex and invariant feature representation by maximum pooling operation. HMAX mainly solves the visual recognition problem which only simulates the ventral pathway in primate vision system. The location information is lost [4]. Another model for general attention and recognition is Where-What Networks (WWNs) introduced by Juyang Weng and his co-workers. This is a biologically plausible developmental model [5], [6] designed to integrate the object recognition and attention namely, what and where information in the ventral stream and dorsal stream respectively. It uses both feedforward (bottom-up) and feedback (top-down) connections.

WWN has six versions. WWN-1 [7] can realize object recognition in complex backgrounds performing in two different selective attention modes: the top-down position-based mode finds a particular object given the location information; the top-down object-based mode finds the location of the object given the type. But only 5 locations were tested. WWN-2 [8] can additionally perform in the mode of free-viewing, realizing the visual attention and object recognition without the type or location information and all the pixel locations were tested. WWN-3 [9] can deal with multiple objects in natural backgrounds using arbitrary foreground object contours, not the square contours in WWN-1. WWN-4 used and analyzed multiple internal areas [10]. WWN-5 is capable of detecting and recognizing the objects with different scale in the complex environments [11]. WWN-6 improves the architecture and mechanisms of the network according to the concept “skull closed” [12]. The pre-programmed “pulvinar ”, which suppresses neurons far from the foreground location, is not needed by WWN-6.

However, for the above versions of WWN, various backgrounds are a serious problem which also exists in other approaches. In real applications, the object contours are arbitrary while the receptive fields are usually regular (e.g., square) in the image scanning. Thus, the leak of pixels of backgrounds into the receptive field is hard to be avoided which may

This work was supported by Fund of State Key Lab. of ASIC & System (11M-S008) and the Fundamental Research Funds for the Central Universities to XW, Changjiang Scholar Fund of China to JW; The authors would like to thank Matthew Luciw at Michigan State University for providing his source program of WWN-3 to conduct the work here.

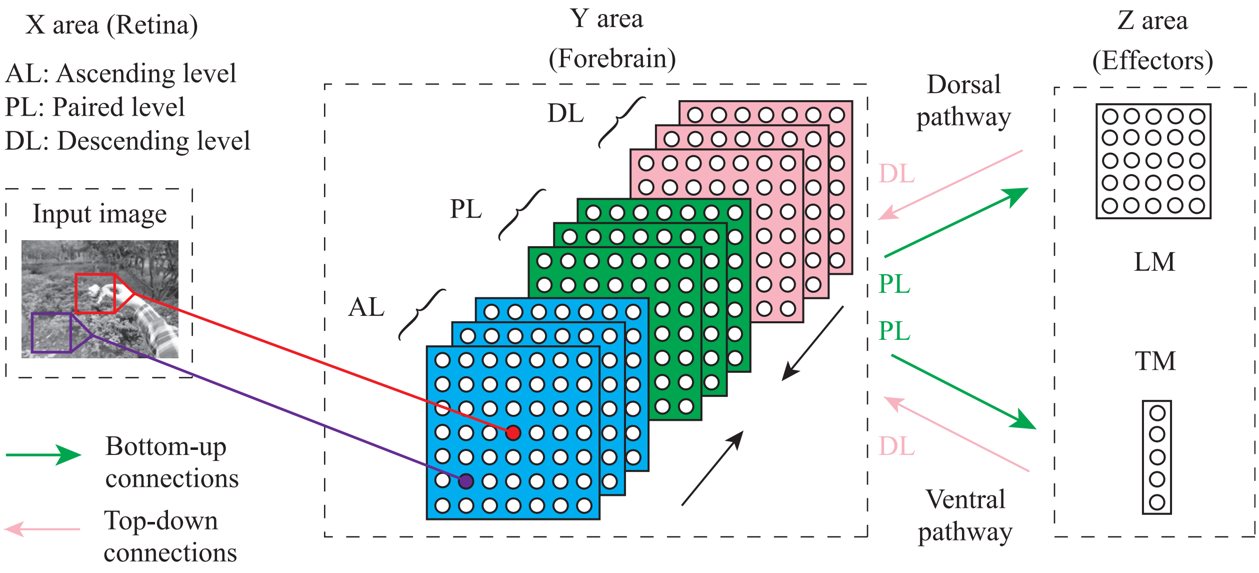


Fig. 1. The structure of WWN-6

produce distracter-like patterns.

We note that, during the competitive self-organization among a limited number of neurons who share a roughly the same receptive field, the patterns from foreground objects appear relatively more often than patterns of backgrounds. Furthermore, neurons whose bottom-up weights match well a foreground object often receive top-down attention boost from a motor area to be more likely a winner. Although their default receptive fields do not match the contour of a foreground object perfectly, among the cases during which each neuron fires, the standard deviation of pixels from a foreground object should be smaller than that of pixels in backgrounds. In this paper, we introduce a new biologically inspired mechanism based on this statistics in nature — synapse maintenance — for each neuron to find precise input fields based on statistics, with handcrafting neither what feature each neuron detects nor its precise input scope.

In the remainder of the paper, the overview of the latest version of WWN is described in Section II. Concepts and the detail algorithms in WWN are presented in Section III. Experiments and results are provided in Section IV. Section V gives the concluding remarks.

II. Network Overview

In this section, the network structure and the overall scheme of the network learning are described.

A. Network Structure

The network (WWN-6) is shown as Fig. 1 which consists of three areas, X area (sensory ends/sensors), Y area (internal brain inside the skull) and Z area (motor ends/effectors). X acts as the retina, which perceives the inputs and sends signals to internal brain Y. The motor area Z serves both input and output. When the environment supervises Z, Z is the input to the network. Otherwise, Z gives an output vector to drive effectors which act on the real world. Z is used as the hub for emergent concepts (e.g., goal, location and type), abstraction (many forms mapped to one equivalent state), and reasoning (as goal-dependant emergent action). In our paradigm, two categories of concepts emerge in Z supervised by the external teacher, the location of the foreground object in the background and the type of this foreground object, corresponding to Location Motor (LM) and Type Motor (TM).

Internal brain Y is like a limited-resource "bridge" connecting with other areas X and Z as its two "banks" through 2-way connections (ascending and descending). Y is inside the closed skull, which is off limit to the teachers in the external environments. Using a prescreening area for each source in Y area, before integration, results in three laminar levels: the ascending level (AL) that prescreenings the bottom-up input, the descending level (DL) that prescreenings the top-down input and paired level (PL) that combines the outputs of AL and DL. In this model, there exist two pathways and two connections. Dorsal pathway refers to the stream $X \rightleftharpoons Y \rightleftharpoons$ LM, while ventral pathway refers to $X \rightleftharpoons Y \rightleftharpoons$ TM, where $\rightleftharpoons$ indicates that each of the two directions has separate connections. That is to say, X provides bottom-up input to AL, Z gives top-down input to DL, and then PL combines these two inputs.

B. General Processing Flow of the Network

The general processing flow of the Network is as follows. The dimension and representation of X and Y areas are hand designed based on the sensors and effectors of the robotic agent or biologically regulated by the genome. Y is skull-closed inside the brain, not directly accessible by the external world after the birth.

1) At time $t = 0$, for each area A in $\{X, Y, Z\}$, initialize its adaptive part $N = (V, G)$ and the response vector $\mathbf{r}$,

where V contains all the synaptic weight vectors and G stores all the neuronal ages.

2) At time $t = 1, 2, ...$, for each A in $\{X, Y, Z\}$ repeat:

   a) Every area A computes its area function f, described below,

$$(\mathbf{r}', N') = f(\mathbf{b}, \mathbf{t}, N)$$

   where $\mathbf{r}'$ is the new response vector of A.

   b) For every area A in $\{X, Y, Z\}$, A replaces: $N \leftarrow N'$ and $\mathbf{r} \leftarrow \mathbf{r}'$. If this replacement operation is not applied, the network will not do learning anymore.

In the remaining discussion, $\mathbf{x} \in X$ is always supervised. The $\mathbf{z} \in Z$ is supervised only when the teacher chooses. Otherwise, $\mathbf{z}$ gives (predicts) effector output.

According to the above processing procedure (described in details in section III), an artificial Developmental Program (DP) is handcrafted by a human to short cut extremely expensive evolution. The DP is task-nonspecific as suggested for the brain in [13], [14] (e.g., not concept-specific or problem-specific).

III. Concepts and Algorithm details

A. Foreground and Background

"Foreground" here refers to the objects to be learned whose contours are arbitrary and "background" refers to the other part in the whole image. Considering the default receptive field of a neuron in AL (square or octagon), we can find two kinds of pixels: foreground pixels and background pixels. The pre-response of the neuron is contributed from both the foreground match and the background match. Although the contribution from foreground pixels can provide a high pre-action value when the foreground object is detected correctly for both type and location, the contribution from background pixels usually gives a somewhat random value. Thus, there is no guarantee that the winner neuron always leads to the correct type in TM and a precise location in LM. In order to deal with the problem, a biologically-inspired mechanism, called synapse maintenance, is introduced. This idea was tried for improving the performance of WWN-3 network and achieved satisfying results for the artificial synthesis images [15]. In this paper, we improve this mechanism (see section III-E as below) and apply it in WWN-6 for natural videos.

B. Inputs and Outputs of Internal Brain Y

As mentioned in section II-A, the inputs to Y consist of two parts, one from X (bottom-up) and the other from Z (top-down).

The neurons in AL have the local receptive fields from X area (input image) shown as Fig. 2. Suppose the receptive field is $a \times a$, the neuron (i, j) in AL perceives the region $R(x, y)$ in the input image ($i \le x \le (i + a - 1)$, $j \le y \le (j + a - 1)$), where the coordinate (i, j) represents the location of the neuron on the two-dimensional plane shown as Fig. 1 and similarly the coordinate (x, y) denotes the location of the pixel in the input image.

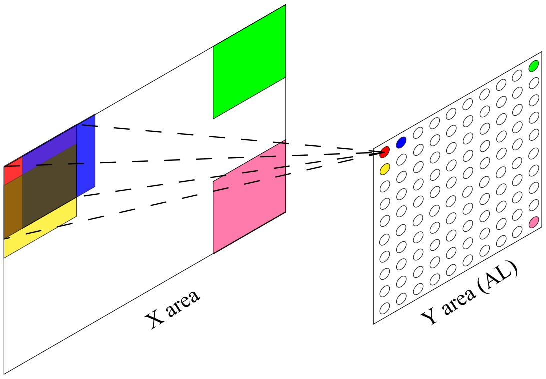


Fig. 2. The illustration of the receptive fields of neurons

Likewise, the neurons in DL have the global receptive fields from Z area including TM and LM. It is important to note that in Fig. 1, each Y neuron has a limited input field in X but a global input field in Z.

Finally, PL combines the outputs of the above two levels, AL and DL, and output the signals to motor area Z.

C. Pre-response of the Neurons

It is desirable that each brain area uses the same area function f, which can develop area specific representation and generate area specific responses. Each area A has a weight vector $\mathbf{v} = (\mathbf{v}_b, \mathbf{v}_t)$. Its pre-response value is:

$$r(\mathbf{v}_b, \mathbf{b}, \mathbf{v}_t, \mathbf{t}) = \dot{\mathbf{v}} \cdot \dot{\mathbf{p}}$$

where $\dot{\mathbf{v}}$ is the unit vector of the normalized synaptic vector $\mathbf{v} = (\dot{\mathbf{v}}_b, \dot{\mathbf{v}}_t)$, and $\dot{\mathbf{p}}$ is the unit vector of the normalized input vector $\mathbf{p} = (\dot{\mathbf{b}}, \dot{\mathbf{t}})$. The inner product measures the degree of match between these two directions of $\dot{\mathbf{v}}$ and $\dot{\mathbf{p}}$, because $r(\mathbf{v}_b, \mathbf{b}, \mathbf{v}_t, \mathbf{t}) = \cos(\theta)$ where θ is the angle between two unit vectors $\dot{\mathbf{v}}$ and $\dot{\mathbf{p}}$. This enables a match between two vectors of different magnitudes. The pre-response value ranges in $[-1, 1]$.

In other words, if regarding the synaptic weight vector as the object feature stored in the neuron, the pre-response measures the similarity between the input signal and the object feature.

D. Two Types of the Neurons

Considering that the learning rate in Hebbian learning (described below) is 100% while the retention rate is 0% when the neuron age is 1, we need to enable each neuron to autonomously search in the input space $\{\dot{\mathbf{p}}\}$ but keep its age (still at 1) until its pre-response value is sufficiently large to indicate that current learned feature vector is meaningful (instead of garbage-like). A garbage-like vector cannot converge to a desirable target based on Hebbian learning.

Therefore, there exist two types of neurons in the Y area (brain) according to their states, initial state neurons (ISN) and learning state neurons (LSN). After the initialization of the network, all the neurons are in the initial state. During the training of the network, neurons may be transformed from initial state into learning state, which is determined by the pre-response of the neurons. In our network, a parameter ϵ_1

is defined. If the pre-response is over $1-\epsilon_1$, the neuron is transformed into learning state, otherwise, the neuron keeps the current state.

E. Synapse Maintenance

With the arbitrary foreground object contours in real environments, the various backgrounds in the receptive fields of Y neurons will influence the recognition results as described in section III-B. An idea is naturally generated: if the network can distinguish the foreground and the background or outline the object contours automatically, the irrelevant components (backgrounds) in the receptive fields of the Y neurons can be removed to reduce the backgrounds interference in the process of object recognition. The synapse maintenance is exactly designed to realize such idea (i.e. removing the irrelevant components, while minimizing removing those relevant components) by calculating the standard deviation of each pixel in different images.

Suppose that the input to a neuron is $\mathbf{p} = (p_1, p_2, ..., p_d)$ and its synaptic weight vector is $\mathbf{v} = (v_1, v_2, ..., v_d)$. The standard deviation of match between v_i and p_i is a measure of expected uncertainty for each synapse i:

$$\sigma_i = E[|v_i - p_i|].$$

Mathematically, σ_i is the expected standard deviation of the match by the synapse i.

Each neuron should dynamically determine which synapse should keep active and which synapse should be retracted depending the goodness of match.We would like to retract synapse i if $\sigma_i(n)$ is large. However, we do not want a fixed threshold to do it because it may cause a synapse to be retracted and extracted repeatedly. Therefore, we introduce a smooth synaptogenic factor $f(\sigma_i)$ defined based on the theory of Mahanobis distance, which is a distance measure introduced by P. C. Mahalanobis in 1936. It is based on correlations between variables by which different patterns can be identified and analyzed. It differs from Euclidean distance in that it takes into account the correlations of the data set and is scale-invariant. In other words, it is a multivariate effect size.

Here, Mahanobis distance between the normalized input vector $\mathbf{p}$ and the synaptic vector $\mathbf{v}$ is used instead of the Euclidian distance. So each component should be weighted by the inverse of the standard deviation σ_i. We have derived the following optimal time-dependent synaptogenic factor for the i-th component of input $\mathbf{p}$ vector:

$$f_i(t) = \frac{1}{\eta(\sigma_i(t) + \epsilon)}$$

where $\epsilon = \delta/\sqrt{12}$ and $\delta = 1/256$ as an estimate of 8-bit resolution of neuronal response in the range $[0, 1]$ and to avoid a too large $f_i(t)$, and $\eta = \sum_{i=1}^{d}(\sigma_i(t) + \epsilon)^{-1}$ so that $\sum_{i=1}^{d} f_i(t) \equiv 1$.

Trimming can be considered the maintenance of spine-synapse combination. We would like to define the trimming of $\mathbf{v} = (v_1, v_2, ..., v_d)$ to be

$$v_i' \leftarrow f_i v_i, \qquad (1)$$

$i = 1, 2, ..., d$. Similarly, trim the input vector $\mathbf{p} = (p_1, p_2, ..., p_d)$ where $\mathbf{p} = (\mathbf{b}, \mathbf{t})$.

After trimmed, the pre-response of every neuron using inner products is calculated as:

$$r(\mathbf{b}, \mathbf{t}, \mathbf{v}_b, \mathbf{v}_t) = \alpha(\frac{\mathbf{b}'}{\|\mathbf{b}'\|} \cdot \frac{\mathbf{v}_b'}{\|\mathbf{v}_b'\|}) + \beta(\frac{\mathbf{t}'}{\|\mathbf{t}'\|} \cdot \frac{\mathbf{v}_t'}{\|\mathbf{v}_t'\|})$$

where $\alpha > 0$, $\beta > 0$ with $\alpha + \beta = 1$. The default values for α, β are $\alpha = \beta == 1/2$. $\mathbf{b}'$, $\mathbf{t}'$, $\mathbf{v}_b'$ and $\mathbf{v}_t'$ are the trimmed input vector and synaptic weight vector of each neuron (trimmed as equation 1). In the current version, synapse maintenance is only applied in the bottom-up connections between X and Y.

F. Competition among the Neurons

Top-k competition takes place among the neurons in the same level in Y area, imitating the lateral inhibition which effectively suppresses the weakly matched neurons (measured by the pre-responses). Top-k competition guarantees that different neurons detect different features. The response $r'(t)$ after top-k competition is

$$r'(t) = \begin{cases} r(t)(r_q - r_{k+1})/(r_1 - r_{k+1}) & \text{if } 1 \le q \le k \\ 0 & \text{otherwise} \end{cases}$$

where r_1, r_q and r_{k+1} denote the first, qth and $(k+1)$th neuron's pre-response respectively after being sorted in descending order. This means that only the top-k responding neurons can fire while all the other neurons are set to zero.

G. Hebbian-like Learning

Hebbian-like learning of our WWN-6 network includes both synapse weight vector update and standard deviation vector update in synapse maintenance.

The update of the synapse weight vector is described as:

$$\mathbf{v}_j(n) = w_1(n)\mathbf{v}_j(n-1) + w_2(n)r'(t)\mathbf{p}_j(t)$$

where $r'(t)$ is the response of the neuron after top-k competition, $\mathbf{v}_j(n)$ is the synapse weight vector of the neuron with age of n and $\mathbf{p}_j(t)$ is the input of the neuron.

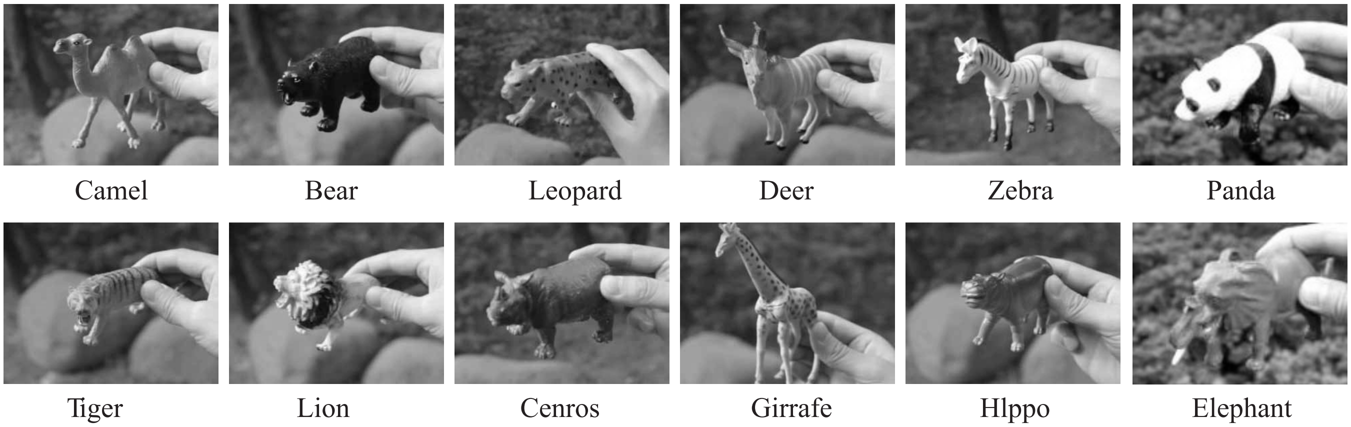
The update of the standard deviation in synapse maintenance is described as:

$$\sigma_i(n) = \begin{cases} 1/\sqrt{12} & \text{if } n \le n_0 \\ w_1(n)\sigma_i(n-1) + w_2(n)|v_i - p_i| & \text{otherwise} \end{cases}$$

where the latency for the synapse maintenance $n_0 = 4$ is set to wait synapse weights (the first order statistics) to get good estimates first through the first n_0 updates before the standard deviation σ_i (the second order statistics) can have reasonable observations. The default estimate for σ_i, $1/\sqrt{12}$, is needed at early ages.

In the above two equations, w_1 and w_2 are the two parameters representing retention rate and learning rate with $w_1 + w_2 \equiv 1$. These two parameters are defined as following:

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



Camel Bear Leopard Deer Zebra Panda

Tiger Lion Cenros Girrafe Hlppo Elephant

Fig. 3. 12 objects to be learned in the experiment

where $u(n)$ is the amnesic function:

$$u(n) = \begin{cases} 0 & \text{if } n \le t_1 \\ c(n - t_1)/(t_2 - t_1) & \text{if } t_1 < n \le t_2 \\ c + (n - t_2)/r & \text{if } t_2 < n \end{cases}$$

where n is the firing age of the neuron, $t_1 = 20, t_2 = 200, c = 2, r = 10000$ [16].

Only the firing neurons (firing neurons are in learning state definitely) and all the neurons in initial state will implement Hebbian-like learning, updating the synaptic weights according to the above formulas. In Y area, if the neuron in learning state is one of the top-k winners and its pre-response is over $1 - \epsilon_2$, the neuron will be fired and implement Hebbian-like learning. The firing age of the neurons in learning state and initial state is updates as

$$n(t+1) = \begin{cases} n(t) & \text{if the neuron is ISN} \\ n(t) + 1 & \text{if the neuron is top-k LSN} \end{cases} .$$

To a neuron, the lower the firing age the higher the learning rate. That is to say, ISN is more capable to learn new concepts than LSN. If the neurons are regarded as resources, ISNs are the idle resources while LSNs are the developed resources.

H. How each Y neuron matches its two input fields

All Y neurons compete for firing via the above top-k mechanisms. The initial weight vector of each Y neuron is randomly self-assigned, as discussed below. We would like to have all Y neurons to find good vectors in the input space $\{\dot{\mathbf{p}}\}$. A neuron will fire and update only when its match between $\dot{\mathbf{v}}$ and $\dot{\mathbf{p}}$ is among the top, which means that the match for the bottom-up part $\dot{\mathbf{v}}_b \cdot \dot{\mathbf{b}}$ and the match for the top-down part $\dot{\mathbf{b}}_t \cdot \dot{\mathbf{t}}$ must be both top. Such top matches must be sufficiently often in order for the neuron to mature.

This gives an interesting but extremely important property for attention — relatively very few Y neurons will learn background, since a background patch does not highly correlated with an action in Z.

> Whether a sensory feature belongs to a foreground or background is defined by whether there is an action that often co-occurs with it.

IV. Experiments and Results

A. Sample Frames Preparation from Natural Videos

In our experiment, 12 objects shown in Fig.3 have been learned. The raw video clips of each object to be learned were completely taken in the real natural environments. During video capture, the object held by the teacher's hand was required to move slowly so that the agent could pay attention to the object. Fig. 4 shows the example frames extracted from a continuous video clip as an illustration which needs to be preprocessed before input to the network. The pre-processing described below is automatically or semi-automatically via handcrafted programs.

1) Resize the image extracted from the video clip to normalize the size of foreground object in different frames as big as the receptive field size of each Y neuron.
2) In our experiments, the teacher provided the correct information of the samples, including the type and the location of the object in any natural backgrounds. Thus, such information needs to be recorded as the standard of test and the supervision in Z area.

B. Verifying a Network

The training set consisted of even frames extracted from 12 different video clips, with one type of foreground object per video. We trained every possible object to be learned at every possible location (pixel-specific) for each epoch, and we trained over many epochs. So, there are 12 classes × 2 (iterations) training instances × 23 × 23 locations = 12696 different training foreground configurations. The test set consisted of odd frames. After every epoch, we tested every possible foreground at every possible location. There are $12 \times 23 \times 23 = 6348$ different test foreground configurations.

Considering both the foreground and background are different in every video frame, the network is nearly 100% short of resource to memorize all the foreground configurations. For example, if one video contains 500 frames, as there are only 6 neurons at each location, but 500 training foreground configurations, the resource shortage is $(500 - 6)/500 = 98.8\%$.

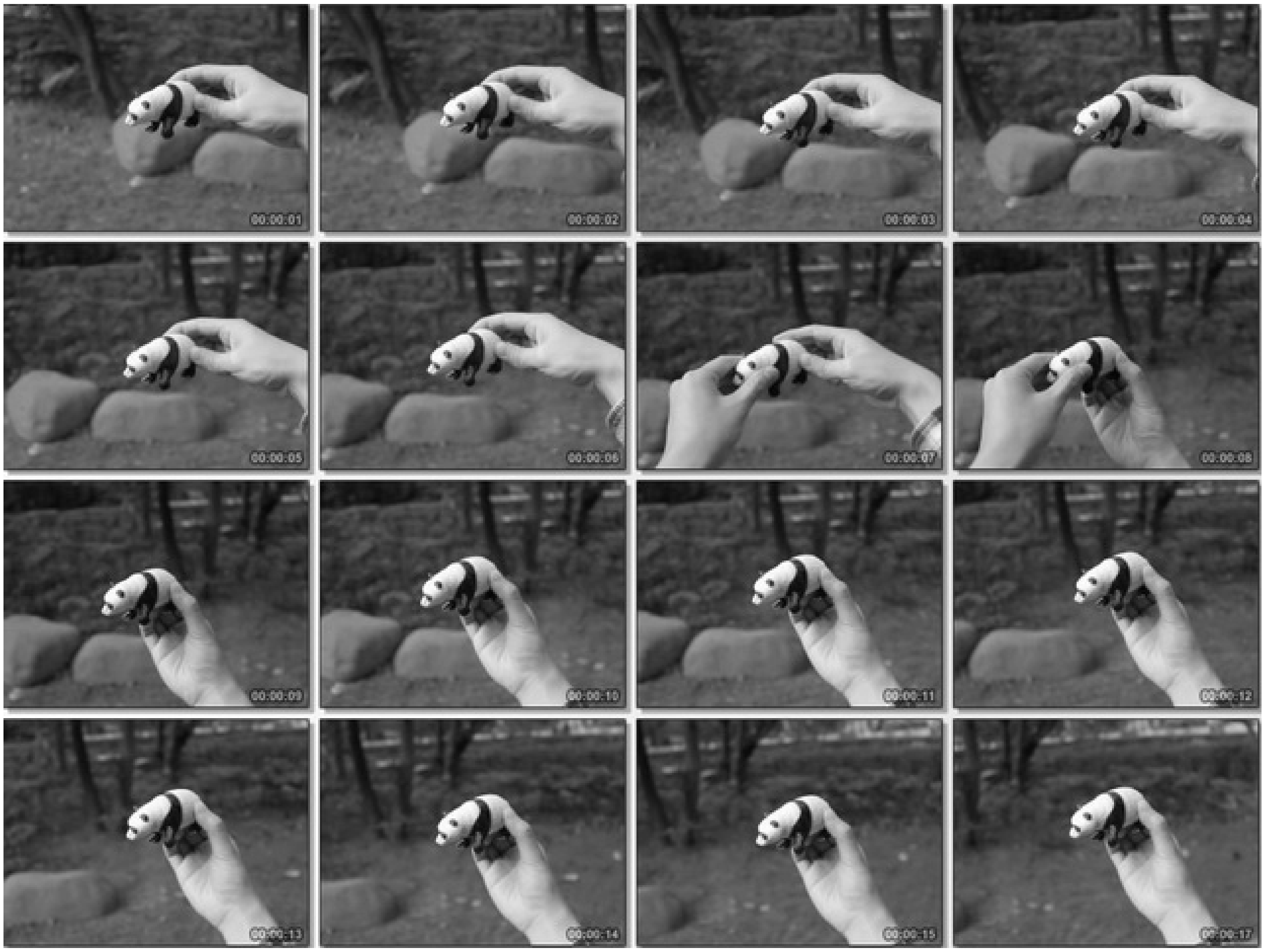
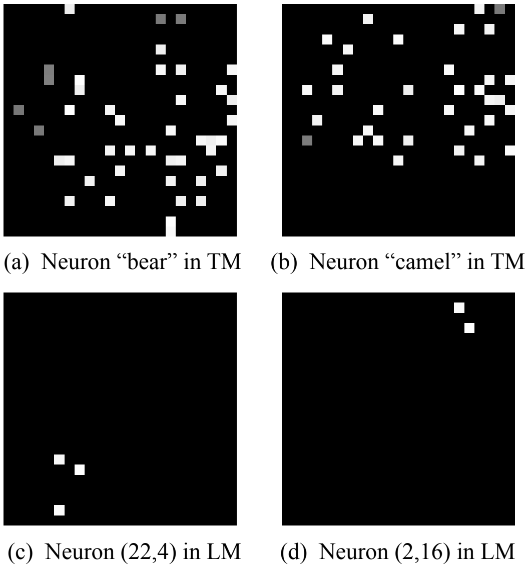


Fig. 4. Frames extracted from a continuous video clip and used in the training and testing of the network

C. Network Performances

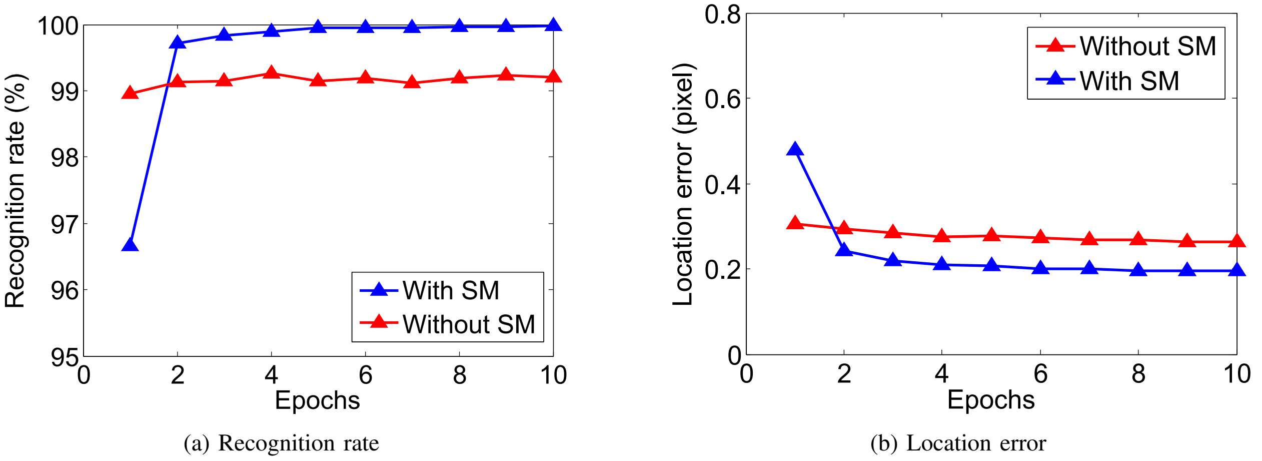
To see how the synapse maintenance influences the network performances, we tested the network with/without synapse maintenance in free-viewing mode (no top-down attention). As shown in Fig. 5, the synapse maintenance improved the performances of the network including recognition rate and localization precision, though the performances became a little worse in the first epoch. After 5 epochs of training, the network with synapse maintenance reached a correct disjoint classification rate nearly 100%.

In order to investigate the detailed effects of synapse maintenance mechanism, the standard deviation (σ_i) and synaptogenic factor (f_i) after 10 epochs are visualized as Fig. 6. The removal of the background pixels is not as effective as the results in artificial synthesis images. In our experiment, the input images for the network training were extracted from the natural video clips, which indicates that the variation of the foreground object is also considerable (for example, affected by illumination or a little different viewangle) compared to that of backgrounds. Thus, there does not exist significant difference between the standard deviation of the pixels in foreground and background.


(a) Neuron "bear" in TM (b) Neuron "camel" in TM

(c) Neuron (22,4) in LM (d) Neuron (2,16) in LM

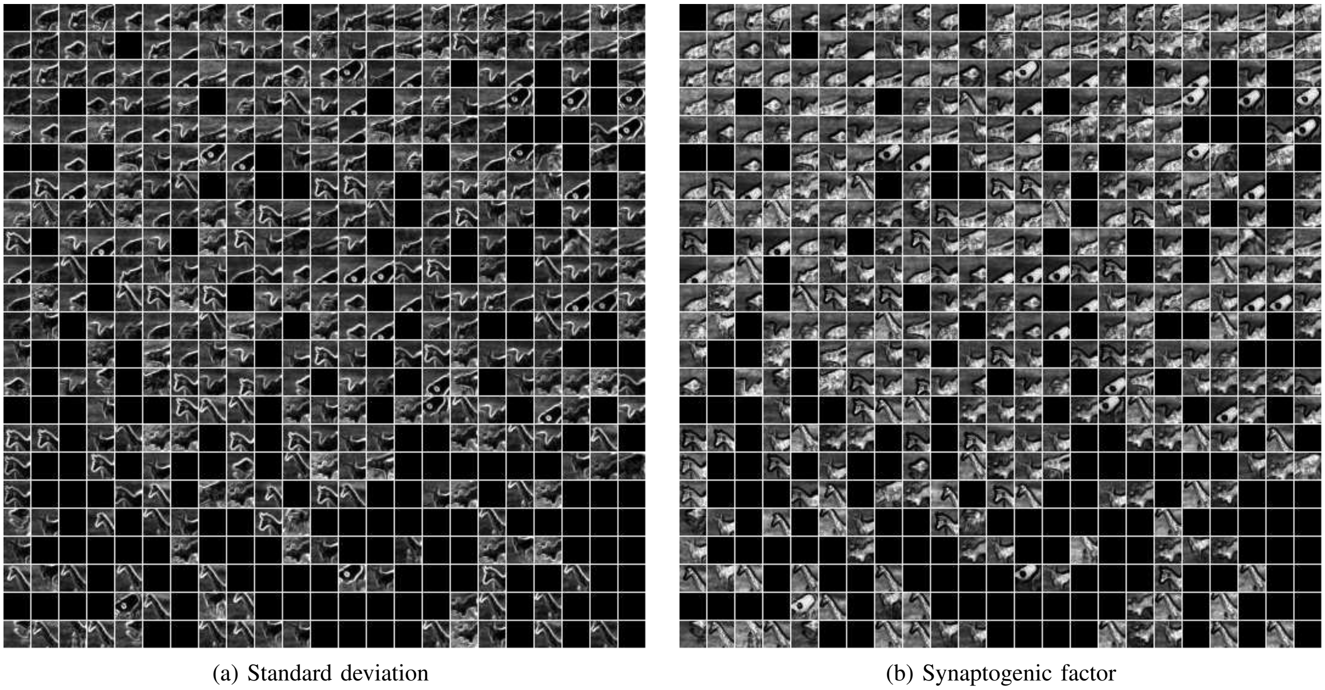
Fig. 7. Bottom-up weights visualization of two neurons in TM and LM. (row, column) represents the neuron position. Here only the weights of Y neurons in first depth are visualized. The size of TM (give the object type) is 12×1 and the size of LM (give the object location) is 23×23.



(a) Recognition rate

(b) Location error

Fig. 5. Network performance variation within 10 epochs with/without synapse maintenance (SM).



(a) Standard deviation

(b) Synaptogenic factor

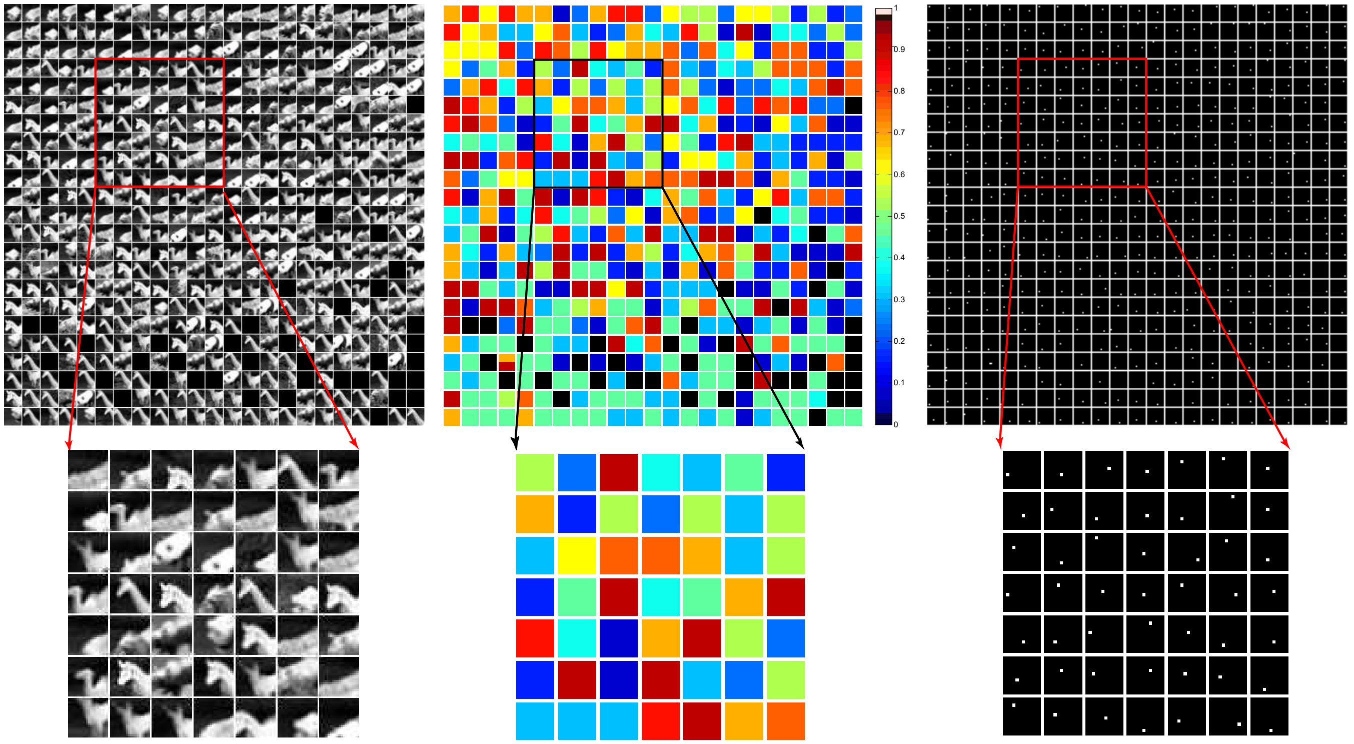
Fig. 6. Visualization of the variables in synapse maintenance. The black patch refers to the neurons which has not done synapse maintenance.

Furthermore, the synaptic weights of neurons in Y area and Z area (TM and LM) are visualized in Fig. 8 and Fig. 7 to study the details of WWN-6 learning effect from the natural video frames. It shows that any Y neuron in any depth can only detect a specific object ("what") feature (shown as Fig. 8 (b)) in a specific position ("where" shown as Fig. 8 (c)) except it is in the initial state whose synaptic weights are visualized as black square patch in Fig. 8.

The bottom-up weights of Z neurons shown in Fig. 7 represent the connection strength from Y to Z, normalized to the range from 0 (black) to 255 (white). The distribution of the nonzero weights, shown in Fig. 7 (a) and (b), should be scattered as shown since a particular object type (e.g., bear) appeared at all the possible image locations detected by Y neurons (at depth 1) each tuned to that type and a location. Likewise, the distribution of the nonzero weights, shown in Fig. 7 (c) and (d), should be localized as shown since objects at a particular location (e.g., (row, column) = (22, 4)) appeared in the vicinity of a single location detected by Y neurons (at depth 1) tuned to an object type and that location. The bottom-up weights from other Y depths to the Z area are similar.

V. Conclusion and Future Work

In this paper, a new mechanism for the biologically-inspired developmental network WWN-6, synapse maintenance has been proposed to automatically determine and adapt the receptive field of a neuron. The default shape of the adaptive field does not necessarily conform to the actual contour of an object, since the object may have different variations in its different



(a) Bottom-up weights

(b) Top-down weights (TM)

(c) Top-down weights (LM)

Fig. 8. Weight Visualization of the neurons in one depth (23×23) in Y area (6 depths), which have three types: bottom-up weights (connected from X area), top-down weights (connected from TM) and top-down weights (connected from LM). For each neuron, the dimensions of the above weights are 22×22, 12×1 and 23×23 respectively. Block color in (b) represents the object type, and all the 12 objects are mapped into a color bar ranged from 0 to 1. The black square patches in (a), (b) and (c) correspond to the initial state neuron.

parts. The adaptive receptive field intends to find a subset of synapses that provide a better majority of matches. Synapse maintenance achieved impressive results for natural videos, as shown in experiments, under a large resource shortage nearly 100%. This indicates that synapse maintenance has great practical potential in real application.

An ongoing work is to handle different scales of the same object. Other variations are also possible for the WWN to deal with in principle, but future experiments are needed. We believe that synapse maintenance is a necessary mechanism for the brain to learn and to achieve a satisfactory performance in the presence of natural backgrounds.

References

[1] J. E. Laird, A. Newell, and P. S. Rosenbloom, "Soar: An architecture for general intelligence," *Artificial Intelligence*, vol. 33, pp. 1–64, 1987.

[2] J.Weng, "Symbolic models and emergent models: A review," *IEEE Trans. Autonomous Mental Development*, vol. 3, pp. +1–26, 2012, accepted and to appear.

[3] M. Riesenhuber and T. Poggio, "Hierachical models of object recognition in cortex," *Nature Neuroscience*, vol. 2, no. 11, pp. 1019–1025, Nov. 1999.

[4] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 411–426, 2007.

[5] J. Weng, "On developmental mental architectures," *Neurocomputing*, vol. 70, no. 13-15, pp. 2303–2323, 2007.

[6] J.Weng, "A theory of developmental architecture," in *Proc. 3rd Int'l Conf. on Development and Learning (ICDL 2004)*, La Jolla, California, Oct. 20-22 2004.

[7] 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'l Conference on Development and Learning*, Monterey, CA, Aug. 9-12 2008, pp. 61–66.

[8] Z. Ji and J. Weng, "WWN-2: A biologically inspired neural network for concurrent visual attention and recognition," in *Proc. IEEE Int'l Joint Conference on Neural Networks*, Barcelona, Spain, July 18-23 2010, pp. +1–8.

[9] M. Luciw and J. Weng, "Where-what network 3: Developmental top-down attention for multiple foregrounds and complex backgrounds," in *Proc. IEEE International Joint Conference on Neural Networks*, Barcelona, Spain, July 18-23 2010, pp. 1–8.

[10] M.Luciw and J.Weng, "Where-what network-4: The effect of multiple internal areas," in *Proc. IEEE International Joint Conference on Neural Networks*, Ann Arbor, MI, Aug 18-21 2010, pp. 311–316.

[11] X. Song, W. Zhang, and J. Weng, "Where-what network-5: Dealing with scales for objects in complex backgrounds," in *Proc. IEEE International Joint Conference on Neural Networks*, San Jose, California, July 31-Aug 5 2011, pp. 2795–2802.

[12] Y. Wang, X. Wu, and J. Weng, "Skull-closed autonomous development," in *18th International Conference on Neural Information Processing, ICONIP 2011*, Shanghai, China, 2011, pp. 209–216.

[13] 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.

[14] J. Weng, "Why have we passed "neural networks do not abstract well"?" *Natural Intelligence*, vol. 1, no. 1, pp. 13–23, 2011.

[15] Y. Wang, X. Wu, and J. Weng, "Synapse maintenance in the where-what network," in *Proc. IEEE International Joint Conference on Neural Networks*, San Jose, California, July 31-Aug 5 2011, pp. 2822–2829.

[16] J. Weng and M. Luciw, "Dually optimal neuronal layers: Lobe component analysis," *IEEE Trans. Autonomous Mental Development*, vol. 1, no. 1, pp. 68–85, 2009.