# Skull-closed Autonomous Development: WWN-7 Dealing with Scales

Xiaofeng Wu, Qian Guo and Juyang Weng

***Abstract*— The Where-What Networks (WWNs) consist of a series of embodiments of a general-purpose brain-inspired network called Developmental Network (DN). WWNs model the dorsal and ventral two-way streams that converge to, and also receive information from, specific motor areas in the frontal cortex. Both visual detection and visual recognition tasks were trained concurrently by such a single, highly integrated network, through autonomous development. By "autonomous development", we mean that not only that the internal (inside the "skull") self-organization is fully autonomous, but the developmental program that regulates the growth and adaptation of computational network is also task non-specific. This paper focused on the "skull-closed" WWN-7 in dealing with different object scales. By "skull-closed", we mean that the brain inside the skull, except the brain's sensory ends and motor ends, is off limit throughout development to all teachers in the external physical environment. The concurrent presence of multiple learned concepts from many object patches is an interesting issue for such developmental networks in dealing with objects of multiple scales. Moreover, we will show how the motor initiated expectations through top-down connections as temporal context assist the perception in a continuously changing physical world, with which the network interacts. The inputs to the network are drawn from continuous video taken from natural settings where, in general, everything is moving while the network is autonomously learning.**

## 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 [3]. 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. Put simply, the ventral stream (what) is sensitive to visual appearance and is largely responsible of object recognition. The dorsal (where and how) is sensitive to spatial locations and processes motion information.

With the advances of the studies on visual cortex in physiology and neuroscience, several cortex-like network models have been proposed. One Model is HMAX, introduced by Riesenhuber and Poggio [4], [5]. This model is a hierarchical system that closely follows the organization of visual cortex and builds an increasingly complex and invariant feature representation by alternating between a template matching and a maximum pooling operation. In the simplest form of the model, it contains four layers, which are $S_1$, $C_1$, $S_2$, $C_2$ from bottom to top. $S_1$ units corresponding to the classical simple cells in primary visual cortex (V1) [6] take the form of Gabor functions to detect the features with different orientations and scales, which have been shown to provide a good model of cortical simple cell receptive fields. $C_1$ units corresponding to cortical complex cells which show some tolerance to shift and size takes the maximum over a local spatial neighbourhood of the afferent $S_1$ units from the previous layer with the same orientation and scale band (each scale band contains two adjacent Gabor filter sizes). $S_2$ units measure the match between a stored prototype $\mathbf{P}_i$ and the input image at every position and scale using radial basis function (RBF). $C_2$ units takes a global maximum over each $S_2$ type (each prototype $\mathbf{P}_i$), i.e., only keep the value of the best match and discard the rest. Thus $C_2$ responses are shift- and scale-invariant, which are then passed to a simple linear classifier (e.g., SVM). In summary, HMAX is a feed-forward network using unsupervised learning, which only models the ventral pathway in primate vision system while the location information is lost, to implement the feature extraction and combination. And a classifier (e.g., SVM) is a must for the task of object recognition, which means the feature extraction and classification are not integrated in a single network.

Different from HMAX, WWNs introduced by Juyang Weng and his co-workers is a biologically plausible developmental model [7], [8], [9] 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. Moreover, multiple concepts (e.g., type, location, scale) can be learned concurrently in such a single network through autonomous development. That is to say, the feature representation and classification are highly integrated in a single network.

WWN has six versions. WWN-1 [10] can realize object recognition in complex backgrounds performing in two dif-

Qian Guo and Xiaofeng WU are with State Key Lab. of ASIC & System, Fudan University, Shanghai, 200433, China and Department of Electronic Engineering, Fudan University, Shanghai, 200433, China, (email: {10210720110, xiaofengwu} @fudan.edu.cn); Juyang Weng is with School of Computer Science, Fudan University, Shanghai, 200433, China and Department of Computer Science and Engineering,Michigan State University, East lansing, Michigan, 48824, USA, (email:weng@cse.msu.edu); 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.

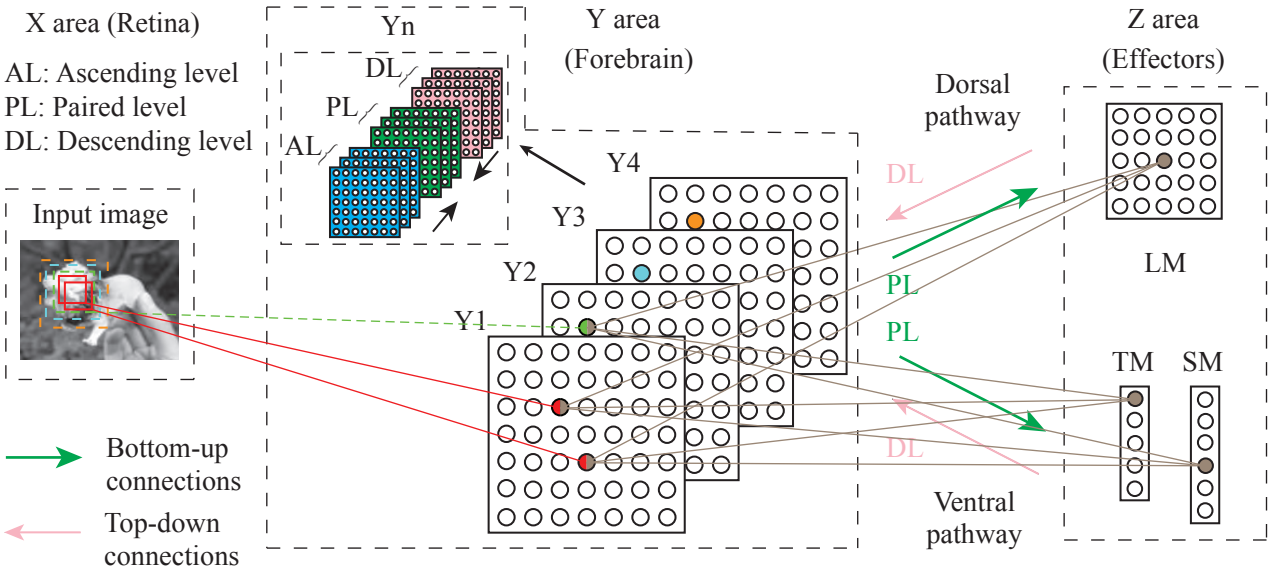


Fig. 1: The structure of WWN-7. The squares in the input image represent the receptive fields perceived by the neurons in the different $Y$ areas. The red solid square corresponds to $Y_1$, the green dashed square with the smallest interval corresponds to $Y_2$, the blue and orange one with larger interval corresponds to $Y_3$ and $Y_4$, respectively. Three linked neurons are firing, activated by the stimuli.

ferent 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 [11] 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 [12] 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 [13]. WWN-5 is capable of detecting and recognizing the objects with different scale in the complex environments [14]. WWN-6 [15] has implemented truly autonomous skull-closed [16], which means that the "brain" inside the skull is not allowed to supervised directly by the external teacher during training and the internal connections are capable of self-organizing autonomously and dynamically (including on and off), meaning more closer to the mechanisms in the brain.

In this paper, a new version of WWN, named WWN-7, is proposed. Compared with the prior versions, especially recent WWN-5 and WWN-6 [17], WWN-7 have at least three innovations described below:

- WWN-7 is skull-closed like WWN-6, but it can deal with multiple object scales.
- WWN-7 is capable of dealing with multiple object scales like WWN-5, but it is truly skull-closed.
- WWN-7 has the capability of temporal processing, and uses the temporal context to guide visual tasks.

In the remainder of the paper, Section II overviews the architecture and operation of WWN-7. Section III presents some important concepts and algorithms in the network. Experimental results are reported 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). The neurons in each area are arranged in a grid on a 2D plane, with equal distance between any two adjacent (non-diagonal) neurons.

$X$ acts as the retina, which perceives the inputs and sends signals to internal brain $Y$. The motor area $Z$ serves as 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, scale and type), abstraction (many forms mapped to one equivalent state), and reasoning (as goal-dependant emergent action). In our paradigm, three categories of concepts emerge in $Z$ supervised by the external teacher, the location of the foreground object in the background, the type and the scale of this foreground object, corresponding to Location Motor (LM), Type Motor (TM) and Scale Motor (SM).

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. In WWN-7, there are multiple $Y$ areas with different receptive fields, shown as $Y_1$, $Y_2$, $Y_3$,$Y_4$... in Fig. 1. Thus the neurons in different $Y$ areas can represent the object features of multiple scales. Using a pre-screening

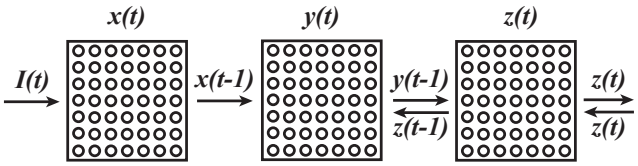


Fig. 2: Architecture diagram of a three-layer network. $\mathbf{I}(t)$ is an image from a discrete video sequence at time $t$. $\mathbf{x}(t)$, $\mathbf{y}(t)$ and $\mathbf{z}(t)$ is the response of the area $X$, $Y$ and $Z$ at time $t$, respectively. The update of each area is asynchronous, which means that at time $t$, $\mathbf{x}(t)$ is the response corresponding to $\mathbf{I}(t)$ (suppose no time delay, and in our experiment, $\mathbf{x}(t) = \mathbf{I}(t)$), $\mathbf{y}(t)$ is the response with the input $\mathbf{x}(t-1)$ and $\mathbf{z}(t-1)$, and similarly, $\mathbf{z}(t)$ is the response with the input $y(t-1)$. Based on this analysis, $\mathbf{z}(t)$ is corresponding to the input image frame $\mathbf{I}(t-2)$, i.e., two-frame delay.

area for each source in each $Y$ area, before integration, results in three laminar levels: the ascending level (AL) that pre-screenings the bottom-up input, the descending level (DL) that pre-screenings 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 and SM, 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.

The dimension and representation of $X$ and $Z$ 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.

### B. General Processing Flow of the Network

For explaining the general processing flow of the Network, Fig. 1 is simplified into a three-layer network shown as Fig. 2, representing $X$, $Y$ and $Z$ respectively.

Suppose that the network operates at discrete times $t = 1, 2....$ This series of discrete time can represent any network update frequency. Denote the sensory input at time $t$ to be $\mathbf{I}_t$, $t = 1, 2, ...$, which can be considered as an image from a discrete video sequence. At time $t = 1, 2, ...$, for each $A$ in $\{X, Y, Z\}$ repeat:

1) 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 $A$, $\mathbf{b}$ and $\mathbf{t}$ is the bottom-up and top-down input respectively.
2) 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.

The update of each area described above is asynchronous [18] shown as the table, which means for each area $A$ in $\{X, Y, Z\}$ at time $t$, the input is the response of the corresponding area at time $t-1$. For example, the bottom-up

| Time t | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\mathbf{z}(t)$: su | B | B | $\alpha$ | * | $\alpha$ | $\beta$ | * | $\beta$ | $\alpha$ | * | * |
| $\mathbf{z}(t)$: em | - | - | ? | $\alpha$ | ? | ? | $\beta$ | ? | ? | $\alpha$ | $\alpha$ |
| $\mathbf{y}(t)$: $\mathbf{z}$ | - | B | B | $\alpha$ | $\alpha$ | $\alpha$ | $\beta$ | $\beta$ | $\beta$ | $\alpha$ | $\alpha$ |
| $\mathbf{y}(t)$: $\mathbf{x}$ | - | $\alpha$ | $\alpha$ | $\alpha$ | $\beta$ | $\beta$ | $\beta$ | $\alpha$ | $\alpha$ | $\alpha$ | $\beta$ |
| $\mathbf{x}(t)$ | $\alpha$ | $\alpha$ | $\alpha$ | $\beta$ | $\beta$ | $\beta$ | $\alpha$ | $\alpha$ | $\alpha$ | $\beta$ | $\beta$ |

TABLE I: Time sequence for an example: the teacher wants to teach a network to recognize two foreground objects $\alpha$ and $\beta$. "B" represents the concept of no interested foreground objects in the image(i.e., neither $\alpha$ nor $\beta$). "em": emergent if not supervised; "su": supervised by the teacher. "*" means free. "-" means not applicable.

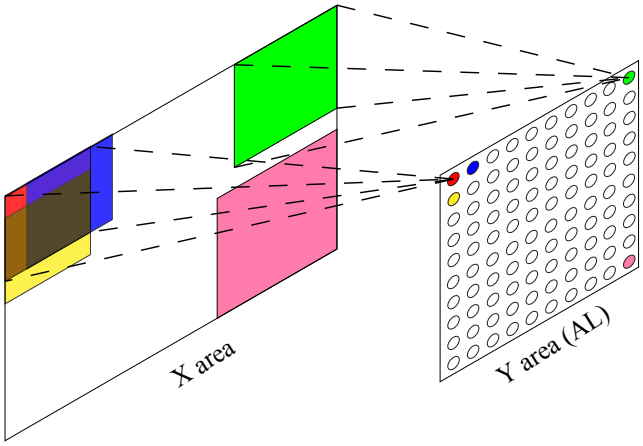


Fig. 3: The illustration of the receptive fields of neurons

and top-down input to $Y$ area at time $t$ is the response of $X$ and $Z$ area at time $t-1$ respectively. Based on such an analysis, the response of $Z$ at time $t$ is the result of the both $\mathbf{x}(t-2)$ and $\mathbf{z}(t-2)$. This mechanism of asynchronous update is different from the synchronous update in WWN-6, where the time of computation of each area was not considered.

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 [19], [20] (e.g., not concept-specific or problem-specific).

## III. Concepts and Algorithms

### A. 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. 3. Suppose the receptive field is $a \times a$, the neuron $(i, j)$ in AL perceives the region $R(x, y)$ in the input image ($i \leq x \leq (i + a - 1)$, $j \leq y \leq (j+a-1)$), 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.

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 $Z$.

### B. Release of neurons

After the initialization of the network, all the $Y$ neurons are in the initial state. With the network learning, more and more neurons which are allowed to be turned into the learning state will be released gradually via this biologically plausible mechanism. Whether a neuron is released depends on the status of its neighbor neurons. As long as the release proportion of the region with the neuron at the center is over $p_0$, this neuron will be released. In our experiments, the region is $3 \times 3 \times d$ ($d$ denotes the depth of $Y$ area) and $p_0 = 5\%$.

### 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:

$$r(\mathbf{v}_b, \mathbf{b}, \mathbf{v}_t, \mathbf{t}) = \dot{\mathbf{v}} \cdot \dot{\mathbf{p}} \tag{1}$$

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 $\theta$ 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.

The firing of a neuron is determined by the response intensity measured by the pre-response (shown as Equation 1). That is to say, If a neuron becomes a winner through the top-k competition of response intensity, this neuron will fire while all the other neurons are set to zero. In the network training, both motors' firing is imposed by the external teacher. In testing, the network operates in the free-viewing mode if neither is imposed, and in the location-goal mode if LM's firing is imposed, and in the type-goal mode if TM's is imposed. The firing of $Y$ (internal brain) neurons is always autonomous, which is determined only by the competition among them.

### D. Tow types of neurons

Considering that the learning rate in Hebbian learning (introduced 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 $\epsilon_1$ 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. Top-k Competition

Top-k competition takes place among the neurons in the same 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 $r'_q$ after top-k competition is

$$r'_q = \begin{cases} (r_q - r_{k+1})/(r_1 - r_{k+1}) & \text{if } 1 \le q \le k \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $r_1$, $r_q$ and $r_{k+1}$ denote the first, $q$th 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.

In $Y$ area, due to the two different states of neurons, top-k competition needs to be modified. There exist two kinds of cases:

- If the neuron is ISN and the pre-response is over $1-\epsilon_1$, it will fire and be transformed into the learning state, otherwise keep the current state (i.e., initial state).
- If the neuron is LSN and the pre-response is over $1-\epsilon_2$, it will fire.

So the modified top-k competition is described as:

$$r''_q = \begin{cases} r'_q & \text{if } r_q > \epsilon \\ 0 & \text{otherwise} \end{cases}$$

$$\epsilon = \begin{cases} 1 - \epsilon_1 & \text{if neuron is ISN} \\ 1 - \epsilon_2 & \text{if neuron is LSN} \end{cases}$$

where $r'_q$ is the response defined in Equation 2.

### F. Hebbian-like Learning

The concept of neuronal age will be described before introducing Hebbian-like learning. Neuronal age represents the firing times of a neuron, i.e., the age of a neuron increases by one every time it fires. Once a neuron fires, it will implement hebbian-like learning and then update its synaptic

weights and age. There exist a close relation between the neuronal age and the learning rate. Put simply, a neuron with lower age has higher learning rate and lower retention rate. Just like human, people usually lose some memory capacity as they get older. At the "birth" time, the age of all the neurons is initialized to 1, indicating $100\%$ learning rate and $0\%$ retention rate.

Hebbian-like learning is described as:

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

where $r'_j(t)$ is the response of the neuron $j$ after top-k competition, $n$ is the age of the neuron (related to the firing times of the neuron), $\mathbf{v}_j(n)$ is the synaptic weights vector of the neuron, $\dot{\mathbf{p}}_j(t)$ is the input patch perceived by the neuron, $w_1$ and $w_2$ are two parameters representing retention rate and learning rate with $w_1 + w_2 \equiv 1$. These two parameters are defined as following:

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

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 $t_1 = 20, t_2 = 200, c = 2, r = 10000$ [21].

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. The age of the neurons in learning state and initial state is updates as

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

Generally, a neuron with lower age has higher 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. So, the resources utilization (RU) in $Y$ area can be calculates as

$$\mathrm{RU} = \frac{\mathrm{N_{LSN}}}{\mathrm{N_{LSN}} + \mathrm{N_{ISN}}} \times 100\%$$

where RU represents the resources utilization, $N_{\mathrm{LSN}}$ and $N_{\mathrm{ISN}}$ are the number of LSN and ISN.

### G. 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.

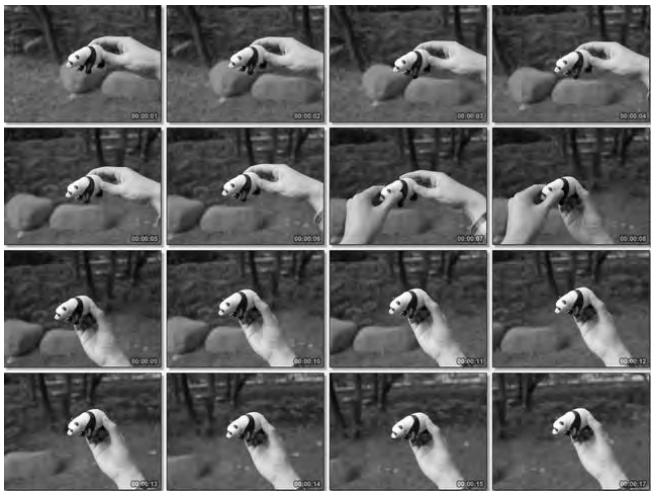

Fig. 5: Frames extracted from a continuous video clip and used in the training and testing of the network

## IV. Experiments and Results

### A. Sample Frames Preparation from Natural Videos

In our experiment, 10 objects shown in Fig.4 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 it. Fig. 5 shows the example frames extracted from a continuous video clip as an illustration which needs to be preprocessed before fed into the network. The pre-processing described below is automatically or semi-automatically via hand-crafted programs.

1) Resize the image extracted from the video clip to fit the required scales demanded in the network training and testing.
2) Provid the correct information including the type, scale and location of the sample in each extracted image with natural backgrounds as the standard of test and the supervision in $Z$ area, just like what the teacher does.

### B. Experiment Design

In our experiment, the size of each input image is set to $32 \times 32$ for $X$ area. For sub-areas $Y_1$, $Y_2$, $Y_3$ and $Y_4$ with individual receptive fields $7 \times 7$, $11 \times 11$, $15 \times 15$ and $19 \times 19$ are adopted in $Y$ area. And totally 10 different types of objects (i.e., TM has 10 neurons) with 11 different scales (from $16 \times 16$ to $26 \times 26$, i.e., SM has 11 neurons) are used in $Z$ area. For each scale of objects, the possible locations is $(32 - S + 1) \times (32 - S + 1)$ ($S = 16, 17, ...26$), i.e., LM has $17 \times 17$ neurons considering that objects with different

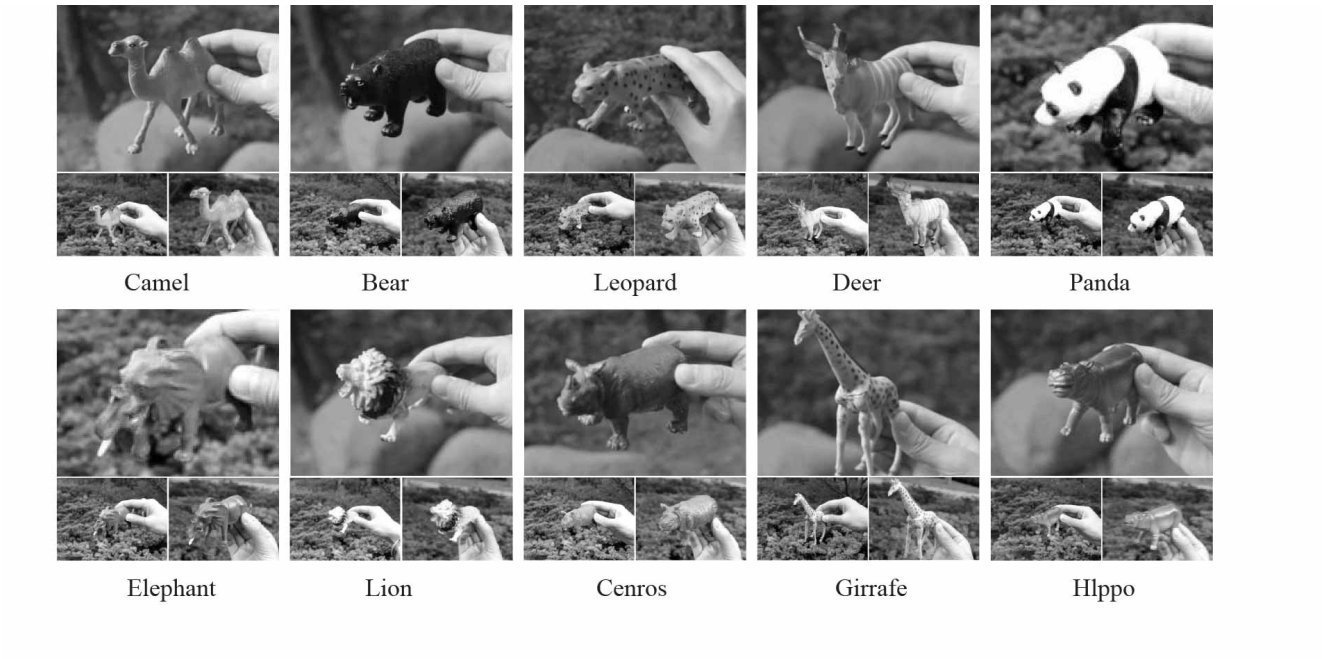


Fig. 4: The pictures on the top visualize 10 objects to be learned in the experiment. The lower-left and the lower-right pictures show the smallest and the largest scale of the objects, respectively (the size of the pictures carries no particular meaning).

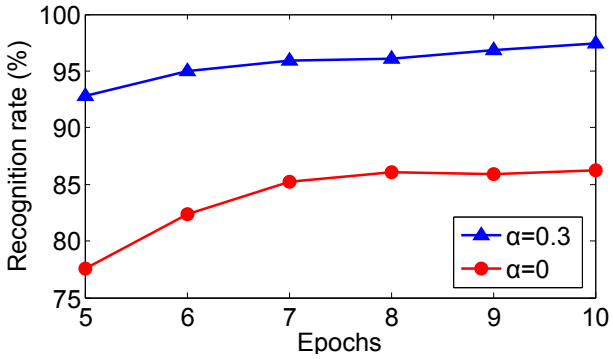


Fig. 6: Recognition rate variation within 6 epochs (from epoch 5th to 10th) under $\alpha = 0$ and $\alpha = 0.3$.

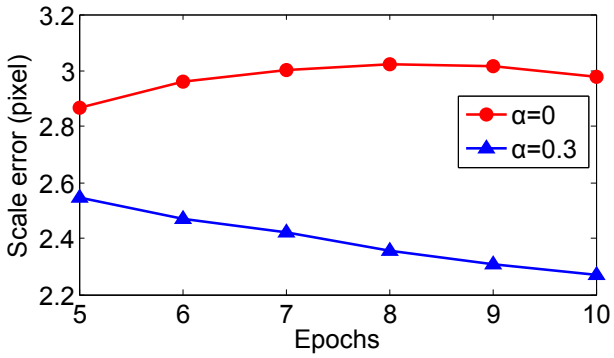


Fig. 7: Scale error variation within 6 epochs (from epoch 5th to 10th) under $\alpha = 0$ and $\alpha = 0.3$.

scales can have the same location. In additional, if the depth of each $Y$ area is 3, the total number of $Y$ neurons is $26 \times 26 \times 3 + 22 \times 22 \times 3 + 18 \times 218 \times 3 + 14 \times 14 \times 3 = 5040$, which can be regarded as the resources of network.

The training set consisted of even frames of 10 different video clips, with one type of foreground object per video. For each training epoch, every object with every possible scale is learned at every possible location (pixel-specific). So, there are 10 classes $\times(17 \times 17 + 16 \times 16 + 15 \times 15 + 14 \times 14 + 13 \times 13 + 12 \times 12 + 11 \times 11 + 10 \times 10 + 9 \times 9 + 8 \times 8 + 7 \times 7)$ locations = 16940 different training cases and the network is about $1 - 5040/16940 = 70.2\%$ short of resources to memorize all these cases. The test set consisted of odd frames of 10 video clips to guarantee the difference of both foreground and background in the network training phase and testing phase. Multiple epochs are applied to observe the performance modification of the network by testing every foreground object at every possible location after each epoch.

During the network training, each type of foreground object with every scale sweeps smoothly across different locations against a fixed or moving random complex background. A different background is used with a different sweeping trajectory. For example, an object of the same scale and of the same appearance sweeps through a natural background. This simulates a baby who holds a toy and moves it across his field of view.

### C. Network Performances

The pre-response of $Y$ neurons is calculated as

$$\mathbf{r}(\mathbf{v}_b, \mathbf{b}, \mathbf{v}_t, \mathbf{t}) = (1 - \alpha)\mathbf{r}^b(\mathbf{v}_b, \mathbf{b}) + \alpha\mathbf{r}^t(\mathbf{v}_t, \mathbf{t}) \quad (3)$$

where $\mathbf{r}_b$ is the bottom-up response and $\mathbf{r}_t$ is the top-down response. Parameter $\alpha$ is applied to adjust the coupling ratio of top-down part to bottom-up part in order to control the

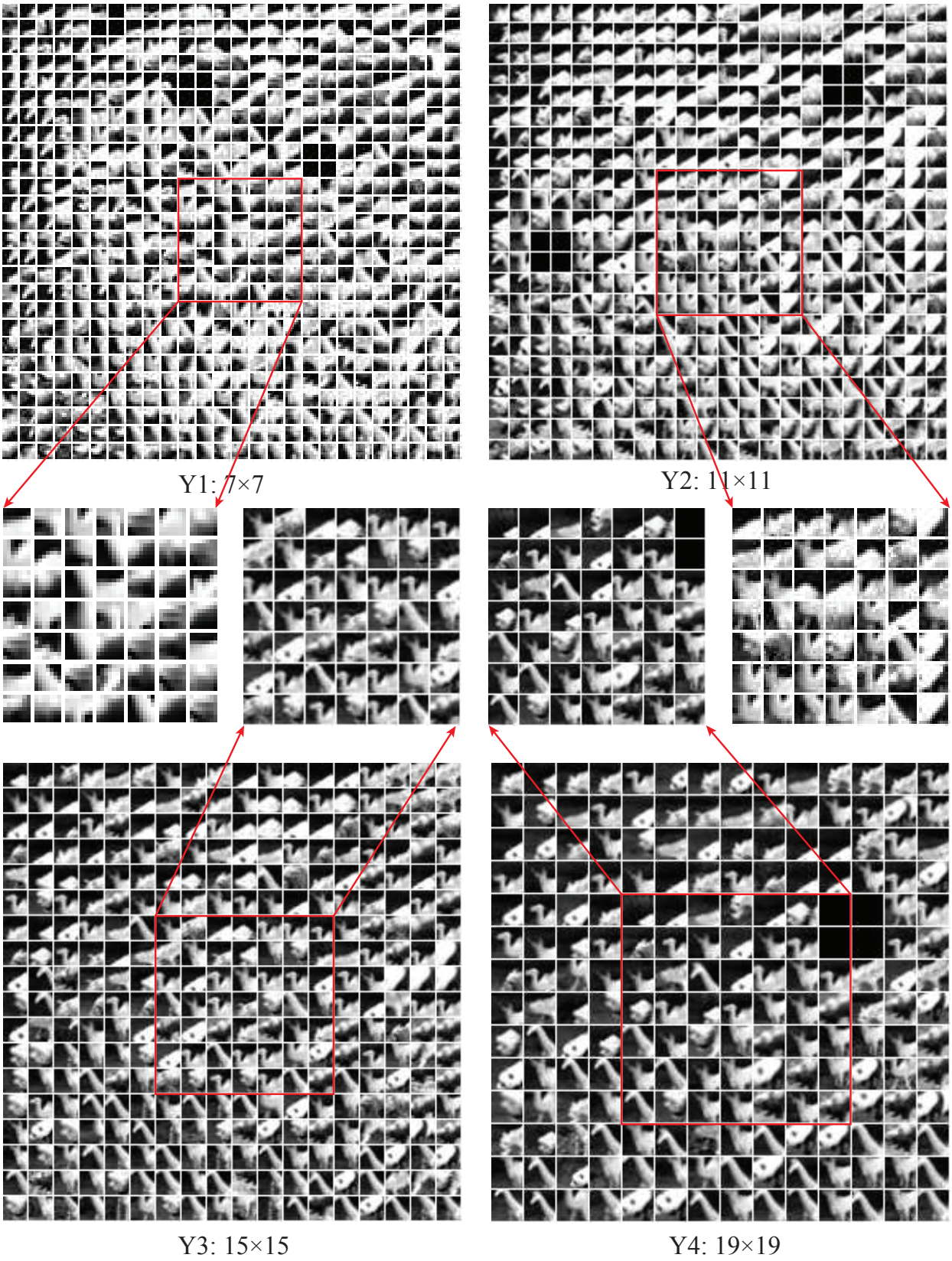


Fig. 9: Visualization of the bottom-up weights of the neurons in the first depth of each $Y$ area. Each small square patch visualized a neuron's bottom-up weights vector, whose size represents the receptive field. The black image patch indicates the corresponding neuron is in the initial state.

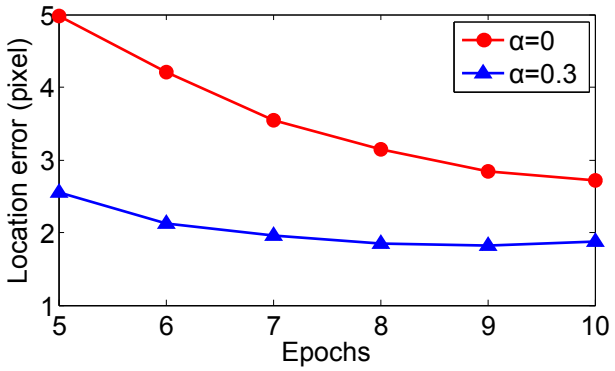


Fig. 8: Location error variation within 6 epochs (from epoch 5th to 10th) under $\alpha = 0$ and $\alpha = 0.3$.

influence on $Y$ neurons from these two parts. This bottom-up, top-down coupling is not new. The novelty is twofold: first, the top-down activation originates from the previous time step $(t-1)$ and second, non-zero top-down parameter ($\alpha > 0$) is used in the testing phase. These simple modifications create a temporally sensitive network. In formula 3, top-down response $\mathbf{r}^t$ consists of three parts from TM, SM and LM respectively. In our experiments, the percentage of energy for each section is the same, i.e., $1/3$.

The high responding $Z$ neurons (including TM,SM and LM) will boost the pre-response of the $Y$ neurons correlated with those neurons more than the other $Y$ neurons mainly correlated with other classes, scales and locations. This can be regarded as top-down biases. These $Y$ neurons' firing leads to a stronger chance of firing of certain $Z$ neurons without taking into account the actual next image (if $\alpha = 1$). This top-down signal is thus generated regarded as an expectation of the next frame's output. The actual next image also stimulates the corresponding neurons (feature neurons) to fire from the bottom-up. The combined effect of these two parts is controlled by the parameter $\alpha$. When $\alpha = 1$, the network state ignores subsequent image frames entirely. When $\alpha = 0$, the network operates in a frame-independent way (i.e., free-viewing, not influenced by top-down signal).

The performance of the network, including type recognition rate, scale error and location error, is shown as Fig 6, 7 and 8. In each figure, two performance curves, which corresponds to two conditions, $\alpha = 0$ and $\alpha = 0.3$, are drawn. As discussed above, parameter $\alpha$ controls the ratio of top-down versus the bottom-up part. The higher $\alpha$ is, the stronger the expectations triggered by the top-down signal is. These three figures indicate that the motor initiated expectations through top-down connections have improved the network performance to a certain extent.

In order to investigate the internal representations of WWN-7 after learning the specific objects in the natural video frames, the bottom-up synaptic weights of the neurons in four $Y$ areas with different receptive fields are visualized in Fig 9. Multiple scales of object features are detected by the neurons in different $Y$ areas shown as the figure.

## V. Conclusion

In this paper, based on the prior work, a new biologically-inspired developmental network WWN-7 has been proposed to deal with general recognition of multiple objects with multiple scales. From the results of experiments, WWN-7 showed its capability of learning multiple concepts (i.e., type, scale and location) concurrently from continuous video taken from natural environments. Besides, in WWN-7, temporal context is used as motor initiated expectation through top-down connections, which has improved the network performances shown in our experiments.

In the future work, more objects with different scales and views will be used in experiment to further verify the performance of WWN-7. And an ongoing work is to study the influence of the parameter $\alpha$ on the network performance and try to implement the autonomous and dynamical adjustment of the percentage of energy for each section (i.e., bottom-up, TM, SM and LM).

## 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] L. G. Ungerleider and M. Mishkin, "Two cortical visual systems," in *Analysis of visual behavior*, D. J. Ingel, Ed. Cambridge, MA: MIT Press, 1982, pp. 549–586.

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

[5] 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.

[6] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *Journal of Physiology*, vol. 160, no. 1, pp. 107–155, 1962.

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

[8] 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.

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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] Y. Wang, X. Wu, and J. Weng, "Skull-closed autonomous development: Wwn-6 using natural video," in *Proc. IEEE International Joint Conference on Neural Networks*, Brisbane, QLD, 10-15 June 2012, pp. 1–8.

[16] 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.

[17] 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.

[18] J. Weng, "Three theorems: Brain-like networks logically reason and optimally generalize," in *Proc. Int'l Joint Conference on Neural Networks*, San Jose, CA, July 31 - August 5 2011, pp. +1–8.

[19] 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.

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

[21] 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.