International Journal of Computational Cognition (http://www.YangSky.com/yangijcc.htm)
Volume 1, Number 2, Pages 1–23, June 2003
Publisher Item Identifier S 1542-5908(03)10201-1/$20.00
Article electronically published on October 13, 2002 at http://www.YangSky.com/ijcc12.htm. Please cite this paper as: ⟨Tao Yang, "Multi-layer Cellular Neural Networks: Theory and Applications to Modeling Nitric Oxide Diffusion in Nervous Systems", *International Journal of Computational Cognition (*http://www.YangSky.com/yangijcc.htm*)*, Volume 1, Number 2, Pages 1–23, June 2003⟩.

MULTI-LAYER CELLULAR NEURAL NETWORKS: THEORY AND APPLICATIONS TO MODELING NITRIC OXIDE DIFFUSION IN NERVOUS SYSTEMS

TAO YANG

This paper is dedicated to machinself.

Abstract. In this paper, we study the dynamic range of multi-layer cellular neural networks (CNN's) by using Lyapunov functions. A theorem is presented to guarantee the existence of equilibrium point of multi-layer CNN. A theorem on globally stable equilibrium point of multi-layer CNN is given. Multi-layer CNN's are used to model some functions of nitric oxide (NO) in nervous systems. In a 2-layer CNN model, the first CNN layer implements synaptic events such as image processing tasks. During these synaptic events artificial NO sources are triggered by the outputs of the first CNN layer. In the 2nd CNN layer, a NO diffusion model is implemented. The output of the 2nd CNN layer functions as a feedback, which mimics the actions of NO to synaptic events, to the first CNN layer. This kind of feedback from the 2nd CNN layer introduces "plasticity" into the CNN synaptic law of the first CNN layer. We improve the NO diffusion model in the 2nd CNN layer by introducing the third CNN layer, which feedbacks the output of the first CNN layer to the NO diffusion model in the second CNN layer. As an application of CNN NO model, we use it to improve the performance of edge detection CNN. Copyright ©2002 Yang's Scientific Research Institute, LLC. All rights reserved.

1. Introduction

Cellular neural networks (CNN)[3, 2, 1, 6] are locally connected networks which can be easily implemented using state-of-the-art VLSI technology. Because of its immense computational power, CNNs are ideally suited for

Received by the editors August 17, 2002 / final version received September 14, 2002.
Key words and phrases. multi-layer cellular neural network, nitric oxide, edge detection, Lyapunov function, stability.
This work was completed with the support of Yang's Foundation of Initiative of Computational Cognition.

image processing and information technology applications where massive computational power is essential for real-time applications. So far, main researches of CNN are concentrated on single layer CNN's though multi-layer CNN's were presented at the same time when single layer CNN's were first presented in 1988[3]. In this paper, we give some basic theoretical results of multi-layer CNN's and present an application of multi-layer CNN's to modeling nitric oxide (NO) diffusion in nervous systems.

NO can be produced by many kinds of tissues in human body and a healthy brain is the richest source of nitric oxide synthase (NOS)([4], chap.10). Although only 2% neurons contain NOS, neurons are highly branched such that all neurons in central nervous system are within a few microns of an NO source. NO has the highest diffusion coefficient of any biological molecule, such as oxygen or carbon monoxide[4]. The estimated mean diffusion distance of NO is $540\mu m$ in normoxic buffer and $100\mu m$ in blood-free, isolated perfused heart in one biological half-life. These are large distances compared to the scale of a neuron in brain. Since NO is a diffusible intra- and inter-cellular messenger molecule, NO has many actions to alter both short-term and long-term synaptic events([4], chap.6). Because it is freely diffusible, NO may function to distribute plasticity across many synapses in a *local* domain([4],chap.7). An example of the effect of NO on synaptic plasticity is provided in Appendix.

Since the highly reactivity of NO, it has a short half-life ($\approx$ 5s). Thus the diffusion of NO in the brain is *local*. This is an essence that NO shares with cellular neural networks (CNN's)[3, 2, 6]. In this paper, we present a two-layer CNN to implement a NO diffusion model for distributing plasticity among cells in the vicinity of NO sources. The basic configuration of the CNN NO model is intuitively simple and similar to its biological counterpart. For a given task, the cell in the first layer will have different levels of "activities" due to different inputs. The activity levels in turn determine different amount of NO generated in the first CNN layer. After NO generated in the first CNN layer, it is diffused in the second layer and contributes plasticity to the template of the first layer by feedback mechanics. The 2-layer CNN NO diffusion model is structurally simple but very sensitive to different inputs. To overcome this problem, we use a third CNN layer to make a feedback from the first CNN layer to the 2nd CNN layer. This 3-layer close-loop configuration is more robust than the 2-layer open-loop one. Simulation results of edge detection using both configurations are provided in this paper.

The organization of this paper is as follows. In Section 2, the theory for multi-layer CNN is presented. In Section 3, a 2-layer CNN and a 3-layer CNN are used to model the NO diffusion model in nervous systems and used

to improve the performance of edge detection. In Section 4, some concluding remarks are given.

2. Theory of Multi-layer Cellular Neural Networks

Since the "sphere of influence" is an essential concept for all cellular neural networks, we give its definition as follows.

Definition 1: *Sphere of influence*[1]
Each cell C_{ij} in a cellular neural network is, by definition, coupled locally only to those neighbor cells which lie inside a prescribed *sphere of influence* $S_{ij}(r)$ of *radius* r, where

$$S_{ij}(r) = \{C_{kl} | \max(|k-i|, |l-j|) \leq r,\ 1 \leq k \leq M,\ 1 \leq l \leq N\}.$$

We usually delete r from $S_{ij}(r)$ and simply write S_{ij} to avoid clutter.

The state equation of a single layer $M \times N$ CNN is given by[1]

$$\begin{aligned} \dot{x}_{ij} &= -x_{ij} + \sum_{C_{kl} \in S_{ij}} A(i,j;k,l) y_{kl} + \sum_{C_{kl} \in S_{ij}} B(i,j;k,l) u_{kl} + z_{ij}, \\ & 1 \leq i \leq M; 1 \leq j \leq N, \end{aligned} \tag{1}$$

where y_{kl} is the output of cell C_{kl}, which is given by

$$y_{kl} = f(x_{kl}). \tag{2}$$

Since each cell in the above CNN is governed by a first-order ODE, we call it the first-order CNN. If cells are high-order dynamic systems, the above single-layer structure can be generally rewritten as

$$\begin{aligned} &\mathcal{F}\left(\frac{d^n x_{ij}}{dt^n}, \left\{ \frac{d^{n-1} x_{kl}}{dt^{n-1}} \right\}_{C_{kl} \in S_{ij}}, \left\{ \frac{d^{n-2} x_{kl}}{dt^{n-2}} \right\}_{C_{kl} \in S_{ij}}, ..., \right. \\ &\left. \left\{ \frac{dx_{kl}}{dt} \right\}_{C_{kl} \in S_{ij}}, \{x_{kl}\}_{C_{kl} \in S_{ij}}, \{\mathbf{u}_{kl}\}_{C_{kl} \in S_{ij}}, z_{ij} \right) = 0, \\ &1 \leq i \leq M; 1 \leq j \leq N, \end{aligned} \tag{3}$$

where $\mathbf{u}_{kl} \in \mathbb{R}^m$ is an m-dimensional input vector for cell C_{kl}. In Eq.(3), each cell is an nth-order continuous-time dynamic system. Thus we call this CNN the nth-order CNN. Also, as in the first-order CNN, each cell in an nth-order CNN only coupled to cells in the sphere of influence. Since the nth-order CNN in Eq.(3) is very general, we normally handle some of its special cases. One widely used form of nth-order CNN is called multi-layer CNN as presented in [3]. A K-layer CNN presented in [3] is in fact a special case of Kth-order CNN. In a K-layer CNN, we denote the cell at site (i, j)

of the pth layer as $C_{ij}^{[p]}$. The state equation of a standard cell $C_{ij}^{[p]}$ in a K-layer $M \times N$ CNN is given by

$$\begin{aligned} \dot{x}_{ij}^{[p]} &= -x_{ij}^{[p]} + \sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} A_{pq}(i,j;k,l) y_{kl}^{[q]} \\ &\quad + \sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} B_{pq}(i,j;k,l) u_{kl}^{[q]} + z_{ij}^{[p]}, \\ &\quad 1 \le p \le K; 1 \le i \le M; 1 \le j \le N, \end{aligned} \tag{4}$$

where $x_{ij}^{[p]}$ and $u_{ij}^{[p]}$ are state variable and input of $C_{ij}^{[p]}$, respectively. $z_{ij}^{[p]}$ is the bias (threshold) of $C_{ij}^{[p]}$. $z_{ij}^{[p]}$ is supposed to be space-varying in this paper. $A_{pq}(i,j;k,l)$ and $B_{pq}(i,j;k,l)$ are feedback and feedforward weights, respectively. We suppose that each layer has an independent input for generality. The output of $C_{ij}^{[p]}$, $y_{ij}^{[p]}$, is usually given by the following standard output equation[3]:

$$y_{ij}^{[p]} = f(x_{ij}^{[p]}) = \frac{1}{2}(|x_{ij}^{[p]} + 1| - |x_{ij}^{[p]} + 1|). \tag{5}$$

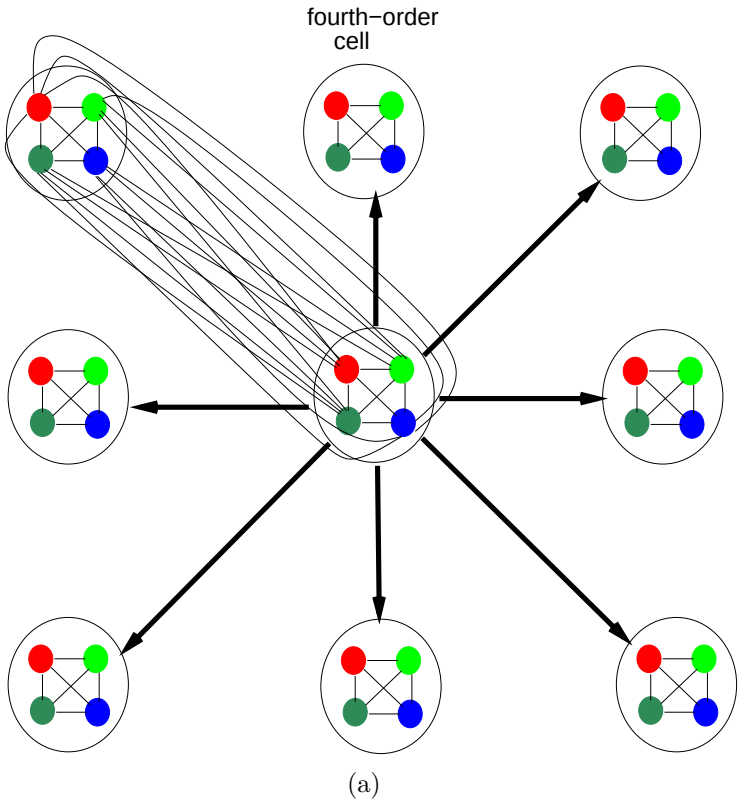
It is also convenient for theoretical analysis to suppose that the input of each cell $u_{ij}^{[p]}$ satisfies

$$|u_{ij}^{[p]}| \le 1, \ 1 \le p \le K; 1 \le i \le M; 1 \le j \le N. \tag{6}$$

Although the K-layer CNN in Eq.(4) is a special case of Kth-order CNN, the very clear and regular coupling structure presented in Eq.(4) has many advantages for designing a high-order CNN for many applications. This is because in many cases it is easier to understand the functions of multi-layer coupled first-order CNN than single-layer multi-order CNN.

An example of how to transfer a fourth-order CNN into a four-layer CNN is shown in Fig.1. Figure 1(a) shows the fourth-order CNN with each cell described by a forth-order dynamical system. In this figure, each spot denotes a state variable. Note that in a VLSI implementation, each spot denotes a capacitor (or inductor). In this figure, only the couplings between two cells are explicitly shown. In this figure, the thin lines between cells are not necessarily physical couplings. Even through each cell couples to any cell in its sphere of influence via a single output, the coupling between each state variable within a neighbor cell and each state variable within the central cell may exist mathematically. Since each cell has the same structure, now let us imagine that we put the four different kinds of capacitors (inductors) into four different "layers", then the coupling between cells will be reorganized

as shown in Fig.1(b). Figure 1(b) shows a four-layer CNN structure, only the couplings for the central cell in the first layer are shown.



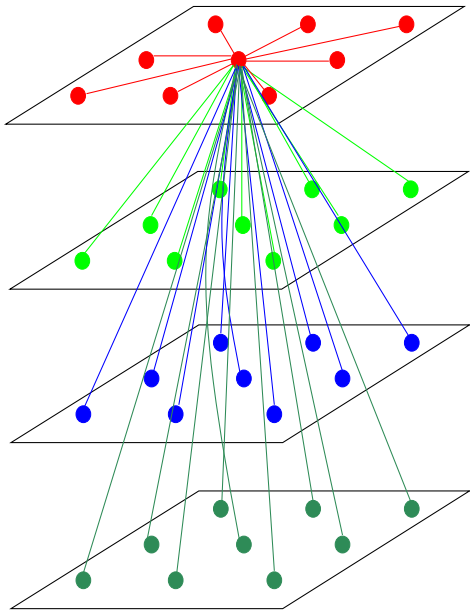
(a)

FIGURE 1. From a fourth-order single-layer CNN to a four-layer CNN. (a) A fourth-order single-layer CNN. (b) The corresponding four-layer CNN.

The following theorem gives the dynamic range of the solutions of the multi-layer CNN.

Theorem 1: *Dynamic range of solutions*
Any solution of the multi-layer CNN in Eq.(4) with any initial condition $\{x_{ij}^{[p]}(0)\}_{p=1}^{K}$ will approach the following compact set:

$$\mathcal{D} = (\mathbb{R}^{KMN}/\mathcal{D}_1) \cap (\mathbb{R}^{KMN}/\mathcal{D}_2) \tag{7}$$



(b)

Fig. 1 (*Continued*).

where

$$
\begin{aligned}
\mathcal{D}_1 \;=\; & \left\{ x_{ij}^{[p]} \left| \left[|x_{ij}^{[p]}| - \frac{1}{2} \left(\sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} |A_{pq}(i,j;k,l)| \delta(i,j;k,l) \right.\right.\right. \right. \\
& \left. \left. \left. + \sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} |B_{pq}(i,j;k,l)| + |z_{ij}^{[p]}| \right) \right]^2 \right. \\
& \left. > \frac{1}{4} \left(\sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} |A_{pq}(i,j;k,l)| \delta(i,j;k,l) \right.\right. \\
& \left. \left. + \sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} |B_{pq}(i,j;k,l)| + |z_{ij}^{[p]}| \right)^2 \right\}, \\
\mathcal{D}_2 \;=\; & \left\{ x_{ij}^{[p]} \left| |x_{ij}^{[p]}| > \left(\sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} |A_{pq}(i,j;k,l)| \delta(i,j;k,l) \right.\right. \right. \\
& \left. \left. \left. + \sum_{q=1}^{K} \sum_{C_{kl} \in S_{ij}} |B_{pq}(i,j;k,l)| + |z_{ij}^{[p]}| \right) \right\} \right.
\end{aligned}
\tag{8}
$$

where

$$\delta(i,j;k,l) = \begin{cases} 0, & \text{when } (k=i, l=j) \text{ and } A(i,j;k,l) \leq 0, \\ 1, & \text{when } (k=i, l=j) \text{ and } A(i,j;k,l) > 0, \\ 0, & \text{when } (k \neq i, l \neq j). \end{cases} \tag{9}$$

Proof: Since $|y_{ij}^{[q]}| \leq 1$, we have

$$A_{pq}(i,j;k,l) y_{kl}^{[q]} x_{ij}^{[p]} \leq |A_{pq}(i,j;k,l)\delta(i,j;k,l)||x_{ij}^{[p]}| \tag{10}$$

and since $|u_{ij}^{[q]}| \leq 1$, we have

$$B_{pq}(i,j;k,l) u_{kl}^{[q]} x_{ij}^{[p]} \leq |B_{pq}(i,j;k,l)||x_{ij}^{[p]}|. \tag{11}$$

First we construct the following Lyapunov function:

$$V_1 = \frac{1}{2} \sum_{q=1}^{K} \sum_{i=1}^{M} \sum_{j=1}^{N} (x_{ij}^{[p]})^2. \tag{12}$$

Let us differentiate V_1 along the solution of Eq.(4)

$$
\begin{aligned}
& \left.\frac{dV_1}{dt}\right|_{Eq.(4)} \\
= & \sum_{q=1}^{K}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(-(x_{ij}^{[p]})^2+\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}A_{pq}(i,j;k,l)y_{kl}^{[q]}x_{ij}^{[p]}\right. \\
& \left.+\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}B_{pq}(i,j;k,l)u_{kl}^{[q]}x_{ij}^{[p]}+z_{ij}^{[p]}x_{ij}^{[p]}\right) \\
\leq & \sum_{q=1}^{K}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(-|x_{ij}^{[p]}|^2+\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}|A_{pq}(i,j;k,l)\delta(i,j;k,l)||x_{ij}^{[p]}|\right. \\
& \left.+\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}|B_{pq}(i,j;k,l)||x_{ij}^{[p]}|+|z_{ij}^{[p]}||x_{ij}^{[p]}|\right), \text{ due to Eqs.(10) and (11)} \\
= & \sum_{q=1}^{K}\sum_{i=1}^{M}\sum_{j=1}^{N}\left\{-\left[|x_{ij}^{[p]}|-\frac{1}{2}\left(\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}|A_{pq}(i,j;k,l)|\delta(i,j;k,l)\right.\right.\right. \\
& \left.\left.\left.+\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}|B_{pq}(i,j;k,l)|+|z_{ij}^{[p]}|\right)\right]^2\right. \\
& +\frac{1}{4}\left(\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}|A_{pq}(i,j;k,l)|\delta(i,j;k,l)\right. \\
& \left.\left.+\sum_{q=1}^{K}\sum_{C_{kl}\in S_{ij}}|B_{pq}(i,j;k,l)|+|z_{ij}^{[p]}|\right)^2\right\} \\
< & 0, \text{ if } x_{ij}^{[p]}\in\mathcal{D}_1.
\end{aligned}
$$

Next we construct KMN Lyapunov functions as follows:

$$
V_{ij}^{[p]}=x_{ij}^{[p]}\text{sign}(x_{ij}^{[p]}) \tag{13}
$$

where sign$(\cdot)$ is the signum function. Along the solution of Eq.(4), we calculate the Dini derivative as

$$\begin{aligned} D^+V_{ij}^{[p]}|_{Eq.(4)} &\leq \left(-|x_{ij}^{[p]}| + \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} |A_{pq}(i,j;k,l)|\delta(i,j;k,l) \right. \\ &\quad \left. + \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} |B_{pq}(i,j;k,l)| + |z_{ij}^{[p]}| \right) \\ &< 0, \text{ if } x_{ij}^{[p]} \in \mathcal{D}_2. \end{aligned}$$

Since the range $\mathcal{D}_2$ is larger than $\mathcal{D}_1$, the solution will first fall in $\mathbb{R}^{KMN}/\mathcal{D}_2$ and then approach $\mathbb{R}^{KMN}/\mathcal{D}_1$ and stay there forever. This complete the proof of this theorem. □

The following theorem guarantees that the multi-layer CNN has at least one equilibrium point.

Theorem 2: *Existence of equilibrium point*
The multi-layer CNN in Eq.(4) has at least one equilibrium point.

Proof: Let the right hand side of Eq.(4) be zero, we have

$$\begin{aligned} x_{ij}^{[p]} &= \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} A_{pq}(i,j;k,l)y_{kl}^{[q]} + \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} B_{pq}(i,j;k,l)u_{kl}^{[q]} + z_{ij}^{[p]}, \\ & 1 \leq p \leq K; 1 \leq i \leq M; 1 \leq j \leq N. \end{aligned} \tag{14}$$

We pack all KMN state variables in Eq. (4) into a $KMN \times 1$ column vector $\mathbf{x}= (x_1, x_2,..., x_i, ..., x_{KMN-1}, x_{KMN})^\top$, then consider the vector operator $\Psi = \{\psi_{ij}^{[p]}(\mathbf{x})\}_{KMN}$, where

$$\begin{aligned} \psi_{ij}^{[p]}(\mathbf{x}) &= \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} A_{pq}(i,j;k,l)y_{kl}^{[q]} \\ &\quad + \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} B_{pq}(i,j;k,l)u_{kl}^{[q]} + z_{ij}^{[p]}. \end{aligned} \tag{15}$$

Let

$$\begin{aligned} D &= \max_{(p,i,j)=(1,1,1)}^{(K,M,N)} \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} |A_{pq}(i,j;k,l)| \\ &\quad + \sum_{q=1}^{K} \sum_{C_{kl}\in S_{ij}} |B_{pq}(i,j;k,l)| + |z_{ij}^{[p]}| \end{aligned} \tag{16}$$

then the vector operator Ψ maps the following compact convex set

$$S = \{\mathbf{x}| \ |x_{ij}^{[p]}| \leq D, 1 \leq p \leq K; 1 \leq i \leq M; 1 \leq j \leq N\} \tag{17}$$

into itself. Since S is a convex compact set with nonempty interior, followed from Brouwer's Fixed Point Theorem, we know that $\Psi : S \mapsto S$ has at least one fixed point $\mathbf{x}^*$, which is an equilibrium point of the multi-layer CNN in Eq.(4). □

For the convenient of theoretical analysis, we first pack all KMN state variables in Eq.(4) into a $KMN \times 1$ column vector

$$\mathbf{x} = (x_1, x_2, ..., x_i, ..., x_{KMN-1}, x_{KMN})^\top$$

and rewrite Eq.(4) into the following form:

$$\begin{aligned} \frac{dx_i}{dt} &= -x_i + \sum_{j=1}^{KMN} a_{ij} f(x_j) + \sum_{j=1}^{KMN} b_{ij} u_j + z_i, \\ & \quad i = 1, 2, ..., KMN \end{aligned} \tag{18}$$

where a_{ij} and b_{ij} correspond to feedback and feedforward synaptic weights $A_{pq}(i, j; k, l)$ and $B_{pq}(i, j; k, l)$, respectively. z_i corresponds to the bias $z_{ij}^{[p]}$.

The following theorem guarantees that the multi-layer CNN has only one equilibrium point, which is globally stable.

Theorem 3: *Globally stable equilibrium point*
Let $\mathcal{A} = (|a_{ij}|)_{KMN \times KMN}$, if the spectral radius of $\mathcal{A}$, $\rho(\mathcal{A}) < 1$, then the multi-layer CNN in Eq.(4) has only one equilibrium point, which is globally stable.

Proof: Theorem 2 guarantees that the multi-layer CNN has at least one equilibrium point. We then prove that it has only one equilibrium point. Let the right hand side of Eq.(18) be zero, then we have

$$x_i = \sum_{j=1}^{KMN} a_{ij} f(x_j) + \sum_{j=1}^{KMN} b_{ij} u_j + z_i, \ i = 1, 2, ..., KMN. \tag{19}$$

Suppose $\mathbf{x}^* = \{x_i^*\}_{KMN \times 1}$ and $\mathbf{x}^{**} = \{x_i^{**}\}_{KMN \times 1}$ are two solutions of Eq.(19), then we have

$$
\begin{aligned}
|x_i^* - x_i^{**}| &= \left[\sum_{j=1}^{KMN} a_{ij} f(x_j^*) + \sum_{j=1}^{KMN} b_{ij} u_j + z_i\right] \\
&\quad - \left[\sum_{j=1}^{KMN} a_{ij} f(x_j^{**}) + \sum_{j=1}^{KMN} b_{ij} u_j + z_i\right] \\
&= \sum_{j=1}^{KMN} a_{ij}(f(x_j^*) - f(x_j^{**})) \\
&\leq \sum_{j=1}^{KMN} |a_{ij}|\ |f(x_j^*) - f(x_j^{**})| \\
&\leq \sum_{j=1}^{KMN} |a_{ij}|\ |x_j^* - x_j^{**}|, i = 1, 2, ..., KMN. \qquad (20)
\end{aligned}
$$

Let us rewrite Eq. (20) into the following vector form over the entire K-layer CNN array, we have

$$
\begin{aligned}
& (|x_1^* - x_1^{**}|, |x_2^* - x_2^{**}|, ..., |x_{KMN}^* - x_{KMN}^{**}|)^\top \\
\leq\ & \mathcal{A}(|x_1^* - x_1^{**}|, |x_2^* - x_2^{**}|, ..., |x_{KMN}^* - x_{KMN}^{**}|)^\top \\
\leq\ & ... \leq \mathcal{A}^n(|x_1^* - x_1^{**}|, |x_2^* - x_2^{**}|, ..., |x_{KMN}^* - x_{KMN}^{**}|)^\top. \qquad (21)
\end{aligned}
$$

Since $\rho(\mathcal{A}) < 1$, we have

$$\lim_{n \to +\infty} \mathcal{A}^n = 0. \qquad (22)$$

We then have $x_i^* = x_i^{**}$, $i = 1, 2, ..., KMN$. This means that each equilibrium point of the K-layer CNN is identical to the others. In the other word, there is only one equilibrium point.

We denote the unique equilibrium point as $\mathbf{x}^* = (x_i^*)_{KMN \times 1}$, then from Eq.(18) we have

$$
\begin{aligned}
\frac{d(x_i - x_i^*)}{dt} &= -(x_i - x_i^*) + \sum_{j=1}^{KMN} a_{ij}[f(x_j) - f(x_j^*)], \\
& \qquad i = 1, 2, ..., KMN. \qquad (23)
\end{aligned}
$$

Since $\rho(\mathcal{A}) < 1$, $(E - \mathcal{A})$ is an M matrix, where E is the identity matrix. Thus there exists a group of positive real numbers, $w_i > 0$, $i =$

$1, 2, ..., KMN$, such that

$$-w_j + \sum_{i=1}^{KMN} w_i |a_{ij}| < 0, \ j = 1, 2, ..., KMN. \tag{24}$$

We construct the following Lyapunov function:

$$V = \sum_{j=1}^{KMN} w_j (x_j - x_j^*) \text{sign}(x_j - x_j^*). \tag{25}$$

Along the solution of Eq.(23), we calculate the Dini derivative of V

$$\begin{aligned} D^+V|_{Eq.(23)} &\leq \sum_{j=1}^{KMN} w_j \left(-|x_j - x_j^*| + \sum_{i=1}^{KMN} |a_{ji}| \ |f(x_i) - f(x_i^*)| \right) \\ &= \sum_{j=1}^{KMN} -w_j |x_j - x_j^*| \\ &\quad + \sum_{j=1}^{KMN} w_j \sum_{i=1}^{KMN} |a_{ji}| \ |f(x_i) - f(x_i^*)| \\ &\leq \sum_{j=1}^{KMN} -w_j |x_j - x_j^*| + \sum_{j=1}^{KMN} w_j \sum_{i=1}^{KMN} |a_{ji}| \ |x_i - x_i^*| \\ &= \sum_{j=1}^{KMN} -w_j |x_j - x_j^*| + \sum_{i=1}^{KMN} w_i \sum_{j=1}^{KMN} |a_{ij}| \ |x_j - x_j^*| \\ &= \sum_{j=1}^{KMN} \left(-w_j + \sum_{i=1}^{KMN} w_i |a_{ij}| \right) |x_j - x_j^*| < 0, \text{ if } \mathbf{x} \neq \mathbf{x}^*. \end{aligned} \tag{26}$$

□

3. Using NO diffusion model to improve edge detection

The "standard" edge detection CNN[6] for grey-scale image is given by

$$\begin{aligned} \boxed{\text{A}} &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \boxed{\text{B}} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}, \\ z &= -0.5, x_{ij}(0) = 0. \end{aligned} \tag{27}$$

The edge-detection CNN in Eq.(27) is sensitive to the difference of different regions of images. We show this kind of sensitivity in Fig.2. Figure 2(a) shows the original image. In this image, the difference of different regions is not homogeneous. This results in a very poor edge detection shown in

Fig.2(b) with threshold $z = -0.5$. Observe that only boundaries between two significantly different regions are detected as edges. Most edges that can be distinguished by naked eyes are not picked out. If we gradually increase the bias z, the detected edges will be enhanced while noise is also introduced. This can be clearly seen from Figs.2(c), (d) and (e), with thresholds $z = -0.3, -0.1$, and 0, respectively.

From the examples in Fig.2 we find that there is a critical tradeoff: If we want to get a complete edge information, we have to suffer from noise. This phenomena also inspire us that if we can increase the bias locally along boundaries while decrease the bias in the other regions, we can find the complete edge information yet with few noise. To do this, a NO diffusion model together with a feedback mechanics are used. The result is the following two-layer CNN.

Layer 1: *Edge detection, synaptic events and NO sources*

$$\dot{x}_{ij}^{[1]} = -x_{ij}^{[1]} + \boxed{A_{11}} * f(x_{ij}^{[1]}) + \boxed{B_{11}} * u_{ij}^{[1]} + z_{ij}^{[1]} \tag{28}$$

where

$$\boxed{A_{11}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \boxed{B_{11}} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix},$$

$$z_{ij}^{[1]} = y_{ij}^{[2]}, x_{ij}^{[1]}(0) = 0, \tag{29}$$

where the bias $z_{ij}^{[1]}$ is space-varying and is modified by the 2nd CNN layer. $y_{ij}^{[2]}$ is the output of the 2nd layer. The output equation for the first layer is the standard CNN output equation as follows.

$$y_{ij}^{[1]} = \frac{1}{2}(|x_{ij}^{[1]} + 1| - |x_{ij}^{[1]} - 1|). \tag{30}$$

Layer 2: *NO diffusion model and synaptic plasticity*

$$\dot{x}_{ij}^{[2]} = -x_{ij}^{[2]} + g(x_{ij}^{[1]}) + \boxed{A_{21}} * g(y_{ij}^{[1]}) \tag{31}$$

where

$$\boxed{A_{21}} = \begin{pmatrix} 0 & 0 & -2 & 0 & 0 \\ 0 & -4 & 16 & -4 & 0 \\ -2 & 16 & -40 & 16 & -2 \\ 0 & -4 & 16 & -4 & 0 \\ 0 & 0 & -2 & 0 & 0 \end{pmatrix} \tag{32}$$

and

$$g(x) \triangleq \begin{cases} 1, & x > 1, \\ x, & 0 < x \leq 1, \\ 0, & \text{otherwise}, \end{cases} \tag{33}$$

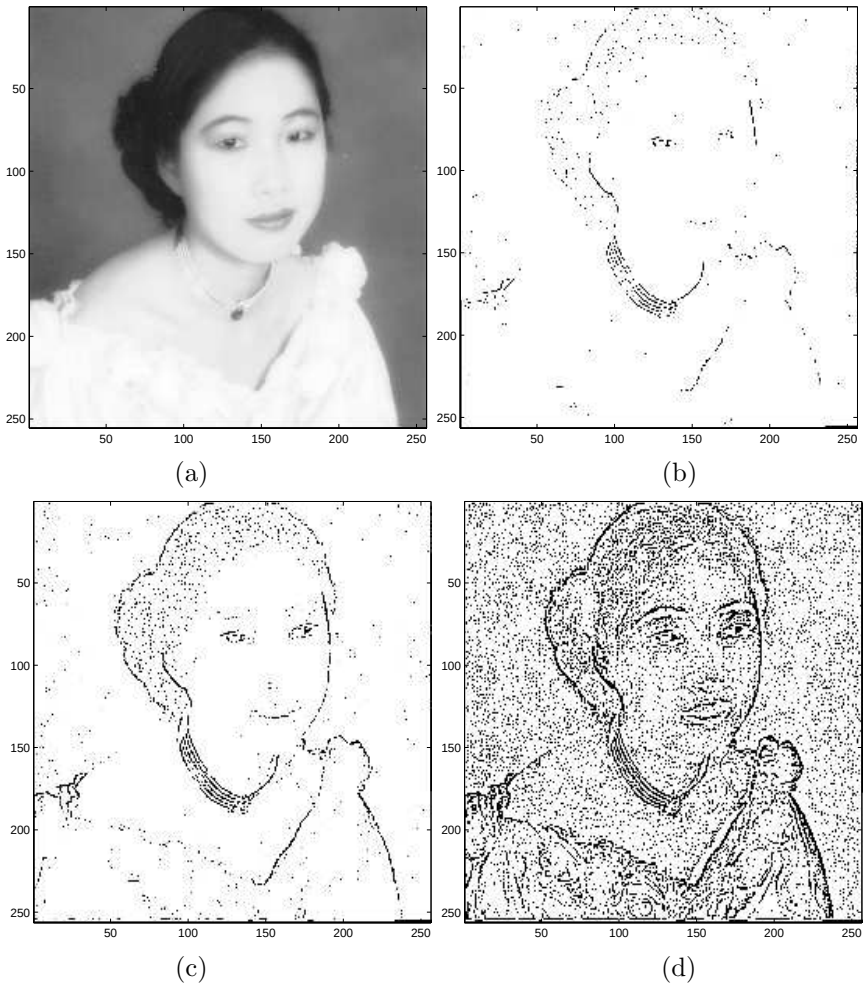


FIGURE 2. Outputs of edge detection CNN with different biases. (a) Original image. (b) Edge detection result with bias $z = -0.5$. (c) Edge detection result with bias $z = -0.3$. (d) Edge detection result with bias $z = -0.1$. (e) Edge detection result with bias $z = 0$.

and the output equation for layer 2 is given by

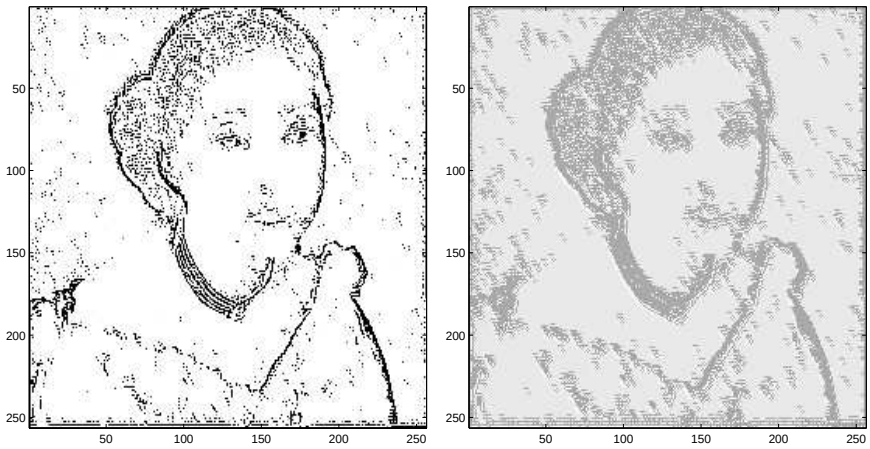
$$y_{ij}^{[2]} = g_1(x_{ij}^{[2]}) \triangleq \tanh(x_{ij}^{[2]})/4 - 0.25. \tag{34}$$

(e)
Fig. 2 (*Continued*).

Observe that the second layer makes the bias of the first layer a value within $(0, -0.5)$. Whenever a cell in the first layer gets a high output, this information triggers a NO source to generate NO. The strength of NO is denoted by $y_{ij}^{[1]}$. In the second layer, the NO strength will be diffused around the NO source. The diffused NO will increase the bias, which in turn makes the first CNN layer more sensitive to edges in the vicinity of the NO source (the detected edge). This makes "weak" edges in the vicinity of "strong" edges be detected.

Figure 3 shows the simulation result. Figure 3(a) shows the output of the first CNN layer, $y_{ij}^{[1]}$. Figure 3(b) shows the output of the NO diffusion model, $y_{ij}^{[2]}$. Comparing the results in Figs.3(a) and 2(b)-2(e), the continuity of the edges in Fig.3(a) is as good as that in Fig.2(d), while the noise is as less as that in Fig.2(c). The inter-cellular plasticity introduced by the NO diffusion model does a very good trade-off between the enhancement of edge and suppression of noise.

Another drawback of the standard edge detection CNN is that it is image-dependent. For different images, the optimal bias is quite different. For example, for the image in Fig.2(a), the optimal bias is around 0.3 but for the image in Fig.4(a), the optimal bias is around 0.5. By introducing the NO diffusion model, we found that the final results are also sensitive to the images. Figure 4 shows the simulation results. Figure 4(a) shows the original image. Figure 4(b) shows the edge detection result without using NO diffusion model. Figure 4(c) shows the edge detection result with the



(a) (b)

FIGURE 3. Improve edge detection using NO diffusion model implemented by a 2-layer CNN. (a) Edge detection result using NO diffusion model. (d) Output of the NO diffusion model.

NO diffusion model. Figure 4(d) shows the output of the NO diffusion model. From Figs.4(c) and (d) we can see that the NO diffusion effect is too strong.

To overcome this problem, we need feedback mechanics from the output of the first layer to control the diffusion effect of the second layer. We thus need a third CNN layer to generate this kind of feedback. We design the following 3-layer CNN to improve further the edge detection:

Layer 1: *Edge detection, synaptic events and NO sources*

$$\dot{x}_{ij}^{[1]} = -x_{ij}^{[1]} + \boxed{A_{11}} * f(x_{ij}^{[1]}) + \boxed{B_{11}} * u_{ij}^{[1]} + z_{ij}^{[1]} \tag{35}$$

where

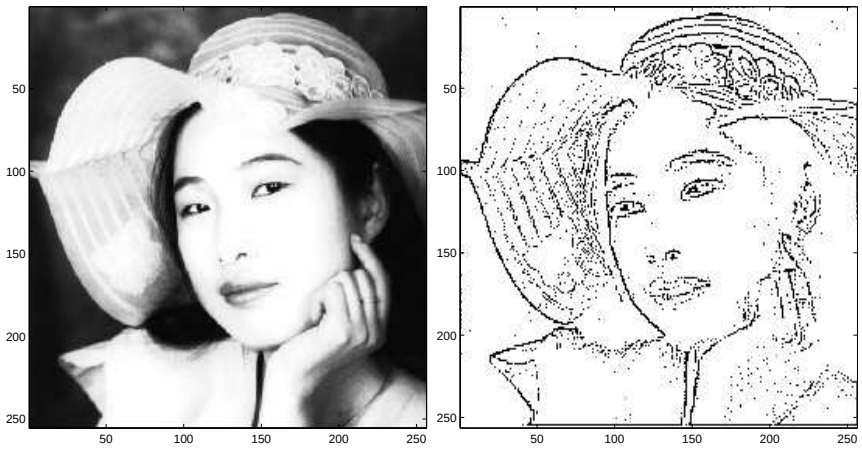
$$\boxed{A_{11}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \boxed{B_{11}} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix},$$

$$z_{ij}^{[1]} = y_{ij}^{[2]}, x_{ij}^{[1]}(0) = 0. \tag{36}$$

The first layer is the same as that in Eq.(28).

Layer 2: *NO diffusion model and synaptic plasticity*

$$\dot{x}_{ij}^{[2]} = -x_{ij}^{[2]} + g(x_{ij}^{[1]}) + \boxed{A_{21}} * g(y_{ij}^{[1]}) \tag{37}$$



(a) (b)

FIGURE 4. Improve edge detection using two different NO diffusion models implemented by a 2-layer CNN and a 3-layer CNN, respectively. (a) Original image. (b) Edge detection result without using NO diffusion model. (c) Edge detection result using NO diffusion model implemented by a 2-layer CNN. (d) Output of the NO diffusion model corresponding to (c). (e) Edge detection result using NO diffusion model with feedback mechanics implemented by a 3-layer CNN. (f) Output of the NO diffusion model with feedback mechanics corresponding to (e).

where $\boxed{A_{21}}$ and $g(x)$ are the same as in Eqs.(32) and (33), respectively. The output equation for layer 2 is given by

$$
\begin{aligned}
y_{ij}^{[2]} &= g_2(x_{ij}^{[2]}, x_{ij}^{[3]}) \\
&= \begin{cases} g_1(x_{ij}^{[2]})(2 - 0.5x_{ij}^{[3]}/3), & x \leq 3 \\ g_1(x_{ij}^{[2]})[1 - 0.5(x_{ij}^{[3]} - 3)/3], & \text{otherwise} \end{cases}
\end{aligned} \tag{38}
$$

where $g_1(\cdot)$ is given by Eq.(34). The output function of the second layer is different from that in Eq.(31). The state variable of the third CNN layer is fed into the 2nd CNN layer.

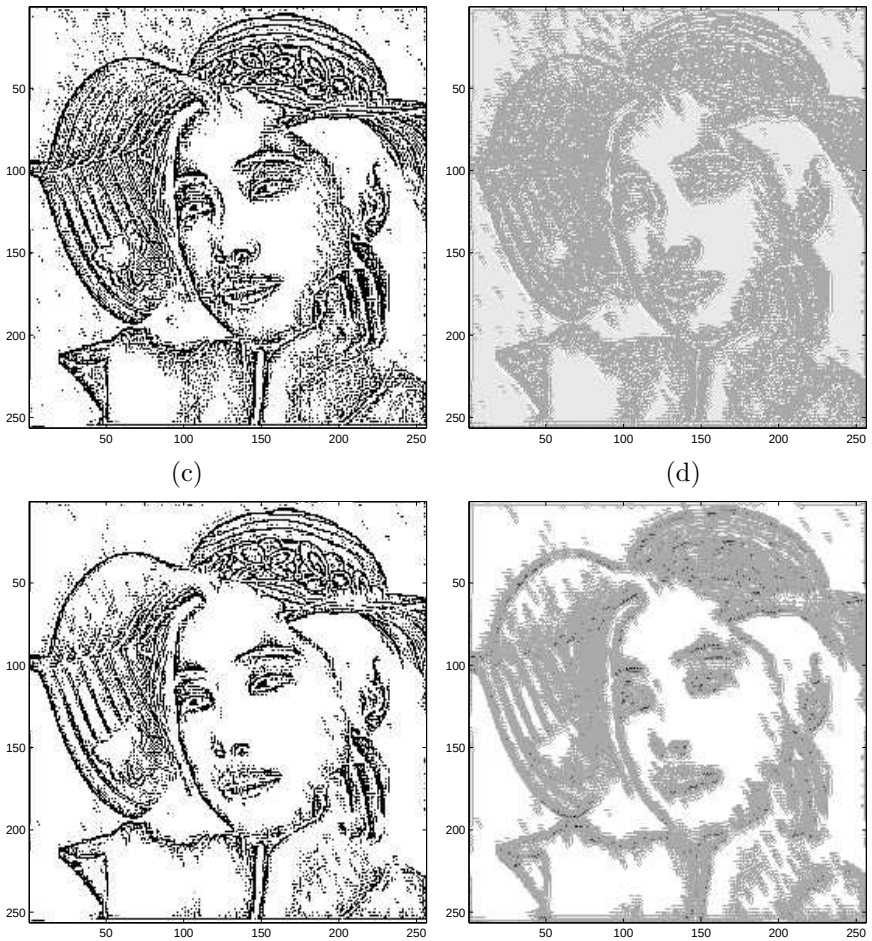


Fig. 4 (*Continued*).

Layer 3: *Feedback for controlling NO diffusion*

$$\dot{x}_{ij}^{[3]} = -x_{ij}^{[3]} + \boxed{A_{31}} * g(y_{ij}^{[1]}),$$

$$\boxed{A_{31}} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}. \tag{39}$$

Observe that the third layer counts how many "black" pixels within a 5×5 neighborhood of the output of the first layer. If the NO diffusion makes the first layer too sensitive to edge detection there will be more cells would like to become "black". The second layer will reduce this kind of sensitivity if many "black" cells are found and will enhance this kind of sensitivity if a few "black" cells are found by the third layer.

The simulation results are shown in Figs.4(e) and (f). Figure 4(e) shows the edge detection result when the feedback for the NO diffusion model in the 2nd CNN layer is used. Observe that the sensitivity has been reduced in regions where no edge exist but almost keep the same in regions where edge exist. Figure 4(f) shows the NO diffusion result with feedback from the first layer. Observe that in many regions far away from edges, the NO diffusion levels have been reduced.

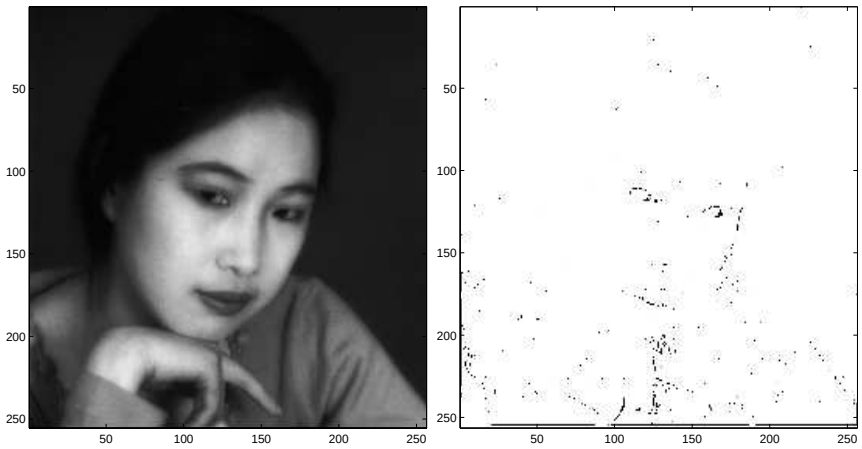
To show the effect of the CNN NO diffusion model, in Fig.5 we show the simulation results of another image where the edge information is very hard to be detected. Figure 5(a) shows the original image. Figure 5(b) shows the output of the "standard" edge detection CNN. Observe that the detected edges are very few. This is because the change of gray values along boundaries is very smooth. Figure 5(c) shows the edge detection result with NO diffusion model without the feedback to control the diffusion. Observe that many noise is introduced while edges are enhanced significantly. Figure 5(d) shows the NO diffusion corresponding to Fig.5(c). Figure 5(e) shows the edge detection result with NO diffusion model and the feedback to control the diffusion. Observe that this is a trade-off between Figs.5(b) and 5(c). Figure 5(f) is the NO diffusion corresponding to Fig.5(e).

4. Concluding remarks

In this paper we embed the NO diffusion model into CNN's for providing "plasticity" of templates. Since different CNN templates have different kinds of sensitivities to their parameter change, the applications of the CNN NO diffusion model presented in this paper are only the tip of an iceberg. The examples presented in this paper show that the CNN NO model can significantly enhance the edge detection results.

Appendix: An example of effect of NO on synaptic plasticity

Although it is well-known that synaptic and action potentials transmitted along neural circuits are the main means of neural information processing, inter-cellular messagers such as NO can propagate information without wired neural circuits. Since NO has a low molecular weight and no electric charge, it is freely diffusible through cell membrane and brain tissue([5], chap.13). This property is important for cerebellar synaptic plasticity[5].

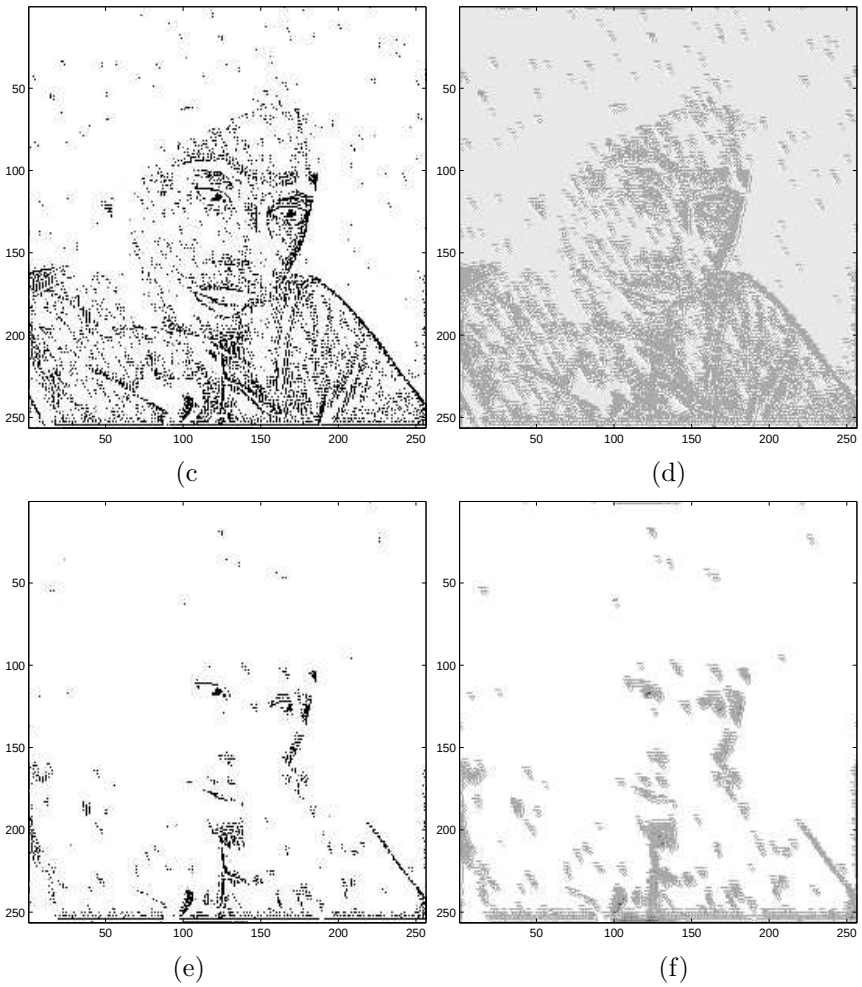


(a) (b)

FIGURE 5. Improve edge detection using two different NO diffusion models implemented by a 2-layer CNN and a 3-layer CNN, respectively. (a) Original image. (b) Edge detection result without using NO diffusion model. (c) Edge detection result using NO diffusion model implemented by a 2-layer CNN. (d) Output of the NO diffusion model corresponding to (c). (e) Edge detection result using NO diffusion model with feedback mechanics implemented by a 3-layer CNN. (f) Output of the NO diffusion model with feedback mechanics corresponding to (e).

As an example we examine how NO to affect synaptic plasticity([5], chap.13). The basic configuration of the neural circuit of the cerebellar cortex is shown in Fig.6(a). In cerebellar cortex, outputs are sent through axons of Purkinje cells. Inputs of Purkinje cells are transmitted through parallel and climbing fibers. When a Purkinje cell receives these two inputs simultaneously, long-term depression (LTD) will be evoked in the parallel fiber/Purkinje cell synapses. This kind of LTD is believed to be the cellular mechanism for cerebellar motor learning.

Cerebellar outputs sent through axons of Purkinje cells is monitored by extra-cerebellar error-detecting neural circuits. If the outputs of Purkinje cells have some errors, errors are fed back to the cerebellum via climbing fibers, and evoke LTD in the parallel fiber/Purkinje cell synapses via the NO released at the climbing fiber input shown in Fig. 6(b). The highly



(c (d)

(e) (f)

Fig. 4 (*Continued*).

diffusible characteristic of NO allows positive interactions between climbing fibers. This is because NO release evoked by many climbing fibers during the same time period can rise the NO level within the local region. In turn, the higher NO level will enable the synaptic error correction. From above we can see the following feedback mechanism: Errors → NO release → error correction. This mechanism has been used in this paper.

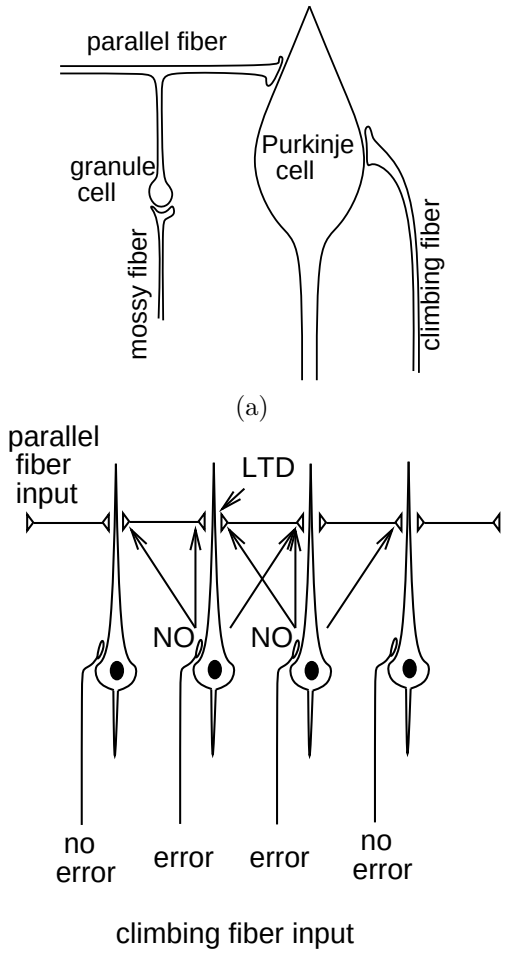


FIGURE 6. (a) Basic configuration of the neural circuit in the cerebellar cortex. (b) The local diffusion of NO generated by climbing fiber input.

REFERENCES

[1] L.O. Chua. *CNN: A vision of complexity*. World Scientific, Singapore : River Edge, N.J., 1998.

[2] L.O. Chua and L. Yang. Cellular neural networks: Applications. *IEEE Transactions on Circuits and Systems*, 35(10):1273–1290, 10 1988.

[3] L.O. Chua and L. Yang. Cellular neural networks: Theory. *IEEE Transactions on Circuits and Systems*, 35(10):1257–1272, 10 1988.
[4] Steven R. Vincent (Ed.). *Nitric oxide in the nervous system*. Neuroscience perspectives. Academic Press, London; San Diego, 1995.
[5] Hiroshi Takagi, Noboru Toda, and R.D. Hawkins (Eds.). *Nitric oxide : roles in neuronal communication and neurotoxicity*, volume 17 of *Taniguchi symposia on brain sciences*. CRC Press, Boca Raton, 1994.
[6] T. Yang. *Handbook of CNN Image Processing: All You Need to Know about Cellular Neural Networks*, volume 1 of *YangSky.com Monographs in Information Sciences*. Yang's Scientific Research Institute, Tucson, AZ, Sept. 2002. ISBN:0-9721212-0-X.

Department of Electrical Engineering and Computer Sciences, Yang's Scientific Research Institute (http://www.YangSky.com), 741 East First Street, Tucson, Arizona 85719-4830, USA

E-mail address: taoyang@yangsky.com