Neurocomputing 238 (2017) 1–12

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Linking synaptic computation for image enhancement

Kun Zhan[a,*], Jinhui Shi[a], Jicai Teng[a], Qiaoqiao Li[b], Mingying Wang[a], Fuxiang Lu[a]

[a] School of Information Science and Engineering, Lanzhou University, Lanzhou, Gansu 730000, China
[b] Faculty of Systems Science and Technology, Akita Prefectural University, Honjo, Akita 015-0055, Japan

ARTICLE INFO

Article history:
Received 28 December 2015
Revised 14 January 2017
Accepted 14 January 2017
Available online 27 January 2017

Keywords:
Image enhancement
Neural networks
PCNN
LSCN
Synaptic computation

ABSTRACT

Linking synaptic computation network is proposed. The linking synapse is introduced into the neural network inspired by the gamma band oscillations in visual cortical neurons, and the neural network is applied to image representation. The linking synaptic mechanism of the network allows integrating temporal and spatial information. An image is input to the network and the enhanced result is obtained by the final linking synaptic state. The visual performance of the results boosts the details while preserving the information in the input image. The effectiveness of the method has been borne out by five quantitative metrics as well as qualitative comparisons with other methods.

1. Introduction

The gamma band oscillations in the primary visual cortex is discovered by Eckhorn et al. [1] and Gray et al. [2] independently, which is considered to be a significant progress in visual neuroscience [3–5]. After the discovery of the gamma band oscillations, Eckhorn et al. explain the synchronous activity of primate cortical neurons by a linking field model [6], and it is simplified to pulse-coupled neural networks (PCNN) [7–9]. As PCNN is developed directly from the studies of visual cortical properties, PCNN has become a general and powerful tool for image processing [9–11]. Based on the studies of the aforementioned models, we propose the linking synaptic computation network (LSCN) in this paper. As Eckhorn et al. introduce the crucial modulatory linking synapse which is inspired by the gamma band oscillations [6], LSCN emphasizes the linking synapse which has neurophysiological support. Using individual spikes allows integrating temporal and spatial information in synaptic computation as well as real neurons do [12]. We use the temporal and spatial integration effect of the linking synapse for image enhancement.

The aim of image enhancement is to improve the visual appearance evaluated by human visual perception, and enhancement is useful when image contrast is imperceptible or barely perceptible [13–15]. The most well-known image enhancement method is the classic histogram equalization because of its simplicity of implementation. The histogram based methods always produce inadequate detail preservation or an excessively enhanced image [16–21]. The transform domain methods achieve enhancement by boosting the coefficients of high-frequency subbands or magnifying the measured contrast [22–24]. However, the parameters of these methods are not suitable to every general image and these methods also magnify the noisy intensity. The linear scale-space theory supports structure-preserving while suppressing noise [25], and there are many improved structure-preserving smoothing techniques are proposed recently, such as bilateral filter [26], weighted least squares filter [27], L_0-smoothing filter [28], guided filter [29], *etc.* If these structure-preserving filters are applied to image enhancement, they usually suffer from “halo” artifacts around the major structures [29–31]. The human visual system (HVS) is powerful in enhancing a scene with precise representation of contrast, and there are many image enhancement methods are inspired from HVS [32–34]. Spiking cortical model (SCM), a variant of PCNN, is applied to image enhancement and its processing mechanism is consistent with HVS [35,36]. Weber–Fechner’s law and the Mach band effect are simulated well in SCM method. However, the results of SCM enhancement algorithm suffer from contrast degradation in bright regions and some pixels with the lowest intensity change to white.

We propose an image enhancement method based on mechanism of the linking synaptic computation. The goal of the image enhancement method is to improve visibility while preserving the information within the input image. Image details are enhanced by using the neural mechanisms related to the linking synapse. We conduct experiments to compare the proposed method with methods based on histogram equalization [37], SCM [35,36], generalized

* Corresponding author.
E-mail addresses: ice.echo@gmail.com, kzhan@lzu.edu.cn (K. Zhan).

http://dx.doi.org/10.1016/j.neucom.2017.01.031
0925-2312/

equalization model [21], and gradient distribution specification [38]. The experiment results illustrate that the LSCN-based method is effective in image enhancement as well as in preserving the information of the input image.

The contributions of this paper are summarized as follows:

1. A neural network LSCN is designed.
2. A general image enhancement framework is proposed.
3. A network iterative stopping condition is proposed.
4. We find that the final linking synaptic state is related with the stimulus image.

2. Linking synaptic computation network

2.1. Leaky integrator

The dynamic potential $v(t)$ of a neural oscillator is described via a leaky integrator,

$$\frac{\mathrm{d}v(t)}{\mathrm{d}t} = -av(t) + s \tag{1}$$

where t is time, s is the input and a is the leak rate.

The potential (1) can be discretized as,

$$\frac{V(n) - V(n-1)}{n-(n-1)} = -aV(n-1) + s \tag{2}$$

where $V(n)$ is the discretized potential, and n is the discrete time.

We can rewrite (2) as,

$$V(n) = bV(n-1) + s \tag{3}$$

where b is the attenuation constant of the leaky integrator.

2.2. LSCN

The linking synapse, the membrane potential and the threshold are instantiated as leaky integrators.

The postsynaptic potential feeds back to modulate the linking synapse. The linking synapse is represented by a leaky integrator [6–9],

$$L_{ij}(n) = lL_{ij}(n-1) + \sum_{pq} W_{ijpq} Y_{pq}(n-1) \tag{4}$$

where each neuron is denoted with indices (i, j), one of its neighboring neurons is denoted with (p, q), l is the linking constant, W_{ijpq} is the weight applied to a linking synapse and $Y_{pq}(n-1)$ is the postsynaptic action potential.

Cortical networks have both feedback and feedforward components and the feedforward component integrates the stimulus-driven neuronal input [6,12,39]. The two components are combined together to produce the membrane potential. In this paper, the membrane potential is represented by a leaky integrator,

$$U_{ij}(n) = fU_{ij}(n-1) + S_{ij}(1 + \beta L_{ij}(n)) \tag{5}$$

where f is the membrane potential attenuation constant, S_{ij} carries the stimulus information and β is the linking strength.

The threshold is an evolution from the neuron analog in [40]. The absolute and relative refractory period are simulated well by the threshold [40]. The threshold of a neuron is represented by a leaky integrator [6,40].

$$\Theta_{ij}(n) = g\Theta_{ij}(n-1) + hY_{ij}(n-1) \tag{6}$$

where g is the threshold attenuation constant, h is a magnitude adjustment, and $Y_{ij}(n-1)$ is the postsynaptic action potential.

At the beginning of the network iteration, the threshold decays from the initial value $\Theta_{ij}(0)$ before the first spike occurs,

$$\Theta_{ij}(n) = g^n \Theta_{ij}(0). \tag{7}$$

Threshold can be replaced with a linear decay function [9],

$$\Theta_{ij}(n) = \Theta_{ij}(n-1) - \delta + hY_{ij}(n-1) \tag{8}$$

where δ is a positive small constant.

Similarly, the threshold decays from the initial value $\Theta_{ij}(0)$ before the first spike occurs,

$$\Theta_{ij}(n) = \Theta_{ij}(0) - n\delta \tag{9}$$

When the membrane potential of a neuron exceeds its threshold in the network iteration, the neuron produces a spike,

$$Y_{ij}(n) = \begin{cases} 1, & \text{if } U_{ij}(n) > \Theta_{ij}(n), \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

The linking synaptic computation network (LSCN) is described by (4), (5), (6), and (10). The schematic of LSCN is shown in Fig. 1. LSCN retains two significant properties in the linking field network [6]. The first is that the linking synapse is represented by the leaky integrator [1,6] and the second is the dynamic threshold [6,40]. The two properties of LSCN are also the significant differences from the conventional integrate-and-fire model. LSCN has two differences from PCNN, the first is that the membrane potential is represented by the leaky integrator, and the second is that the feeding input is simplified to the stimulus only. The modifications are under the consideration that the membrane potential of the most biological neural networks is represented by the leaky integrator and the main contribution of feeding input is the stimulus [6]. The main difference from SCM is that LSCN represents the linking synapse as a leaky integrator.

As the linking strength β is usually set to a small value [9–11], we assume that it is set to 0 and obtain Fig. 2 based on (5) and (9). The firing condition is that the membrane potential is larger than the decaying threshold. As can be seen from Fig. 2, the spike timing is when the membrane potential is almost equal to the threshold,

$$\Theta_{ij}(n) = U_{ij}(n). \tag{11}$$

Once a neuron fires the first spike, in the next iteration of the network the threshold is changed to,

$$\Theta_{ij}(n+1) = gU_{ij}(n) + h, \tag{12}$$

then the threshold delays exponentially according to its attenuation constant g. The second and the following spikes produce when the threshold is almost equal to the membrane potential again, so the firing cycle C_{ij} of a neuron is given by,

$$C_{ij} = \log_g \left(\frac{U_{ij}}{gU_{ij} + h} \right). \tag{13}$$

2.3. Multiple pass

The proposed multiple pass working form of LSCN is a network stopping condition. The stopping condition is how to set the iterative times of the network. The iterative times is usually set manually [9–11]. Different from the conventional PCNN, the iterative process is automatic stop rather than manually set the iterative times in the proposed multiple pass form.

The multiple pass stopping condition is that the network stops when all neurons in the network produce their spikes and it is given by Algorithm 1.

In Algorithm 1, N is the total number of neurons in a network, *firednum* is counting the number of the fired neurons in each iteration, Y_c with initial values of 0 has the same size of Y, and the function ‘or’ performs an element-by-element ‘or’ operator between matrices Y_c and Y.