Northumbria Research Link

Feedforward Computational Model for Pattern Recognition with Spiking Neurons

Malu Zhang[*], Hong Qu[*], Ammar Belatreche[**] and Zhi Zeng[*]

Abstract

Humans and primates are remarkably good at pattern recognition and outperform the best machine vision systems with respect to almost any measure. Building a computational model that emulates the architecture and information processing in biological neural systems has always been an attractive target. To build a computational model that closely follows the information processing and architecture of the visual cortex, in this paper, we have improved the latency-phase encoding to express the external stimuli in a more abstract manner. Moreover, inspired by recent findings in biological neural system, including architecture, encoding, and learning theories, we have proposed a feedforward computational model of spiking neurons that emulates object recognition of the visual cortex for pattern recognition. Simulation results showed that the proposed computational model can perform pattern recognition task well. In addition, the success of this computational model suggests a plausible proof for feedforward architecture of pattern recognition in the visual cortex.

Key Words

Spiking neurons, computational model, spiking neural networks, pattern recognition

1. Introduction

The recognition of different patterns is a fundamental, frequently performed cognitive task [1]. During the

[*] School of Computer Science and Engineering, University of Electronic Science and Technology of China, China; email: {maluzhang@126.com}
[**] School of Computer and Information Sciences, Faculty of Engineering and Environment, Northumbria University, UK; email: ammar.belatreche@northumbria.ac.uk

last decades, some remarkable progresses have been made in solving pattern recognition problems using non-biological methods, such as maximum entropy classifier [2], naive Bayes [3] and SVMs [4]. However, the performance of the biological visual system still outperforms the best machine vision system [5]. This situation motivates us to build a biologically plausible computational model for rapid and robust pattern recognition.

The reason of rapid recognition in biological visual system has been attributed to a hierarchical and mainly feedforward computational model [6–8]. To develop a biological plausible computational model, another challenge is how information is encoded in the biological neural system. Traditionally, it is believed that neurons encode information through the firing rates of biological neurons (rate codes), and this encoding method has been widely used in different applications [9-11]. However, as precise spike-timing neural activities have been observed in many brain regions, including the retina [12–14], the lateral geniculate nucleus [15] and the visual cortex [16], the view that information in the brain is encoded by the timed spikes (temporal codes) rather than firing rate has received increasing attention [17], [18]. These observations have led to a more biologically plausible neuron model, namely spiking neurons, which encode information by the firing times of spikes [19-20]. Networks of spiking neurons (SNNs) can acquire new knowledge through changing the synaptic efficiency. Recently, many supervised learning methods have been proposed to train the SNNs, such as remote supervised method (ReSuMe) [21] and perceptron-based spiking neuron learning rule (PBSNLR) [22], with the aim to ensure that the output neuron can fire a desired spike train when the corresponding input pattern are presented.

The limitation of the original SNNs-based feedforward computational model [23] lies in its encoding scheme which did not incorporate any information extraction to preprocess the input patterns. Also, the learning efficiency of the existing model requires further improvements. This motivates us to develop an efficient computational model of spiking neurons for pattern recognition. In this paper, we made the following contributions: 1) We have improved the existing latency-phase encoding to code the

information in a more concise manner. Experimental results demonstrated that the improved latency-phase encoding method requires less encoding neurons than traditional latency-phase encoding method. 2) We have bridged the gap between the above independent studies including encoding theories, architecture theories, and learning theories. In addition, we have proposed a unified computational model of spiking neural networks (SNNs) for pattern recognition. We have demonstrated that the computational model of SNNs can perform pattern recognition task well. In addition, the success of the proposed approach suggests a plausible proof for a class of feedforward models of object recognition in the visual cortex and could be applied to a lot of neurological experiments.

2. Integrated Network for Pattern Recognition

In this section, we begin by presenting the whole computational model of spiking neurons. Then, the corresponding schemes used in different sub-models are introduced.

2.1 The Proposed Computational Model

As shown in Fig. 1, the architecture of the proposed computational model composes of three functional parts: the encoding, the learning, and the classification part.
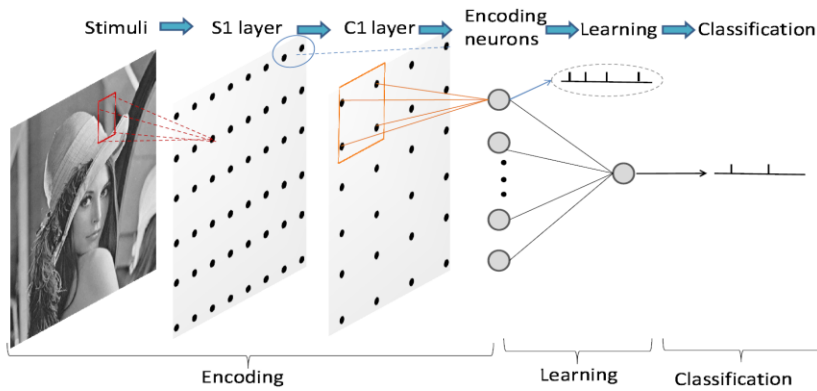
Figure 1. The proposed computational model. A grayscale image is presented to the encoding part. Each neuron in S1 layer linearly integrates the information from its receptive field (e.g., the red rectangular block). A neuron in S1 layer competes with its neighbors through the MAX operation to strive for survival in C1 layer (e.g., the blue circle in S1 layer). Each encoding neuron is associated with a receptive field (e.g., the jacinth rectangular block) to convert the activations of C1 neurons into spatiotemporal patterns. These converted spike patterns are passed to the learning part. The

final decision is made in the classification part.

The encoding part is used to convert the features of external sensory stimulation into explicit firing times of action potentials (spikes). With a combination of supervised spike-timing-based learning, the learning part tunes the synaptic weights, which ensures that the output part can respond to certain patterns correctly. The classification part decides on the extracted feature spikes using a network of spiking neurons.

2.2 Encoding Part

The original latency-phase encoding method is based on the pixel-level features, which makes a large number of redundant spike codes. The main idea of the improved latency-phase encoding is to apply a standard feature selection technique, then convert these features into explicit firing times by latency-phase encoding.

2.2.1 Feature Extraction

Inspired by a recent theory of the feedforward and hierarchical architecture in the visual cortex that accounts for the rapid and robust recognition performance, researchers proposed a computational model (HMAX) for pattern recognition [24]. With the usage of simple cells (SCs) and complex cells (CCs), the HMAX performs remarkably well in pattern recognition. The information flow in HMAX can be expressed as S1→ C1→ S2→ C2. In this paper, we have only adopted a simplified structure of two layers (only S1 and C1). In the S1 layer, difference of Gaussian (DoG) filters were used to mimic the neural processing mechanisms in the retina [5]. The function of DoG used here is

$$DoG_{s,l_c}(l) = G_{\sigma(s)}(l - l_c) - G_{3.\sigma(s)}(l - l_c) \tag{1}$$

$$G_{\sigma(s)}(l) = \frac{1}{2\pi \cdot \sigma(s)^2} \cdot \exp(\frac{\|l\|}{2\sigma(s)^2}) \qquad (2)$$

Where $G_{\sigma(s)}$ is the 2-D Gaussian function with the parameter of $\sigma(s)$. l is the position of photoreceptor cell and l_c is the center position of the filter. Then the activation of the cell in S1 is computed as

$$x = <I, \varphi_i> = \sum_{i \in R_i} I(l) \cdot \varphi_i(l) \qquad (3)$$

where $I(l)$ is the intensity of pixel l in photoreceptor cells. R_i is the receptive field of the corresponding neuron i, and φ_i is the weight of the DoG filter.

In the C1 layer, a MAX operation is implemented to increase invariance to size and position. Each neuron competes with its nearby neurons located within the same receptive field. After the MAX operation, each survival neuron in C1 layer denotes a feature.

2.3.2 Latency-Phase Encoding

The final activations of neurons in C1 were used to produce spikes by latency-phase encoding [25]. The flowchart of the latency-phase coding scheme was illustrated in Fig. 2. Each encoding neuron was associated with a receptive field (RF) which consists of m × n cells. As shown in Fig. 2(c), the intensity value of a feature in C1 is converted to a action potential though a logarithmic intensity transformation, which was expressed as

$$t_i = f(s_i) = t_{max} - \ln(\alpha \cdot s_i + 1) \qquad (4)$$

where t_i is the spike time of neuron i, t_{max} is the time duration of the encoding window, α is a

scaling parameter, and s_i is the intensity of pixel i. Each cell in the receptive field has different subthreshold membrane oscillations (SMO). The SMO for the i th pixel in the receptive field is described as

$$OSC_i = \cos(\omega t + \varphi_i) \tag{5}$$

where ω determines the number of oscillation cycles, and φ_i is the initial phase of the i th pixel in the receptive field. φ_i is defined as

$$\varphi_i = \varphi_0 + (i-1)\Delta\varphi \tag{6}$$

where φ_0 is the initial phase (in this paper $\varphi_0 = 0$), and $\Delta\varphi$ is a constant phase gradient. After the intensity value was converted into a timed action potential, the alignment operation was implemented by reassigning each spike to the nearest peak of the corresponding SMO as illustrated in Fig. 2(d). As shown in Fig. 2(e), spikes generated from photoreceptor cells in the same receptive field were compressed into one sequence of spikes by the associated encoding neuron. The external stimulus can then be reconstructed using a simple inverse latency transformation process as shown in Fig. 2(f).
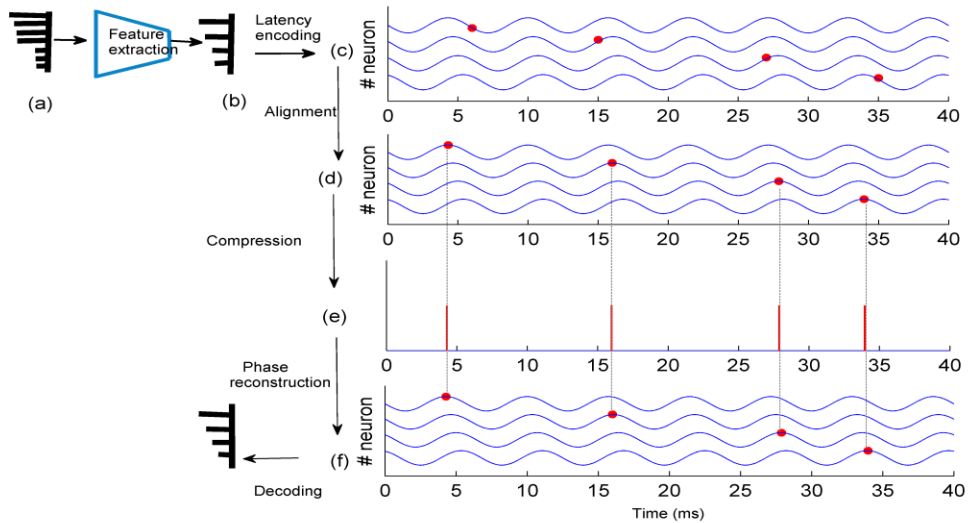
Figure 2. Illustration of the improved latency-phase coding method. (a) Stimulations intensities. (b) The information is converted into different spikes. (c) These converted spikes are assigned with their corresponding oscillations. (d) After converting the intensity value of each pixel into a timed action potential, the alignment procedure is implemented that the neurons fire only when the subthreshold membrane potential reaches their nearest peaks. (e) These spikes produced from the same receptive field in C1 layer are compressed into a sequence of spikes. (f) The compressed sequence of spikes could be reconstructed via phase reconstruction.

2.3 Learning Part

2.3.1 Spiking Neuron Model

The leaky integrate-and-fire (LIF) model [26] is used in this paper. The membrane potential of the LIF neuron is expressed as

$$V(t) = \sum_j w_{ij} \sum_f K(t - t_j^f) + V_{rest} \tag{7}$$

where t_j^f is the f th spike of presynaptic neuron j, and w_{ij} is the synaptic weight. V_{rest} is the rest potential of the LIF neuron. K denotes the normalized postsynaptic potential (PSP) kernel defined as

$$K(t - t_j^f) = V_0 \left[\exp(-(t - t_j^f)/\tau_m) - \exp(-(t - t_j^f)/\tau_s) \right] \tag{8}$$

where τ_m and τ_s are two parameters that determine the shape of the postsynaptic potential (PSP). V_0 normalizes postsynaptic potential so that the maximum value of the kernel $K(t - t_j^f)$ is 1. When the membrane potential $V(t)$ of neuron reaches the firing threshold ϑ at time t^{f_r} as

$$V(t^{f_r}) \geq \vartheta \quad \text{and} \quad \frac{dV(t^{f_r})}{dt} > 0 \tag{9}$$

an action potential (spike) will be triggered.

In this paper, the firing threshold ϑ is set to 1 mV. After firing, $V(t)$ immediately starts

dropping to resting potential ($V_{rest} = 0$), and the neuron cannot fire regardless of the number of the input spikes. This phase is known as refractory period R_a (In this paper, R_a is set to 3 ms).

2.3.2 PBSNLR Learning rule

PBSNLR [22] was proposed to train the spiking neurons to output desired spike trains. Experimental results demonstrated that the learning performance and accuracy of PBSNLR were better than that of other learning methods. The PBSNLR [22] transformed the supervised learning task into a classification problem using the perceptron learning rule with the samples defined as follows. All desired firing times were regarded as positive samples and other time points were regard as negative samples. Then, these misclassified samples were trained as

$$\overrightarrow{W^{new}} = \overrightarrow{W^{old}} + \beta \overrightarrow{P^t} \qquad \text{if } d_t = 1 \text{ and } V(t) < \vartheta \tag{10}$$

$$\overrightarrow{W^{new}} = \overrightarrow{W^{old}} - \beta \overrightarrow{P^t} \qquad \text{if } d_t = 0 \text{ and } V(t) \geq \vartheta \tag{11}$$

Where $\overrightarrow{P^t} = (P_1^t, P_2^t, ..., P_N^t)$, P_i^t is the sum of PSPs induced by all the spikes that have arrived through the ith synapse at t. $d_t = 1$ (or $d_t = 0$) means t is the desired (or undesired) output time.

2.4 Classification Part

In the classification part, we predefine different spike sequences for different input patterns and the learning neuron is trained to fire a desired sequence of spikes when a corresponding pattern is present. In order to quantitatively evaluate the learning performance, a correlation-based measure of spike timing [27] was adopted to measure the similar degree between the target and observed output spike trains. When the actual output spike train is the same with the desired output spike

8

train, the measure $C = 1$, and the C decreases towards 0 for loosely correlated sequences of spikes.

3. Simulation Results

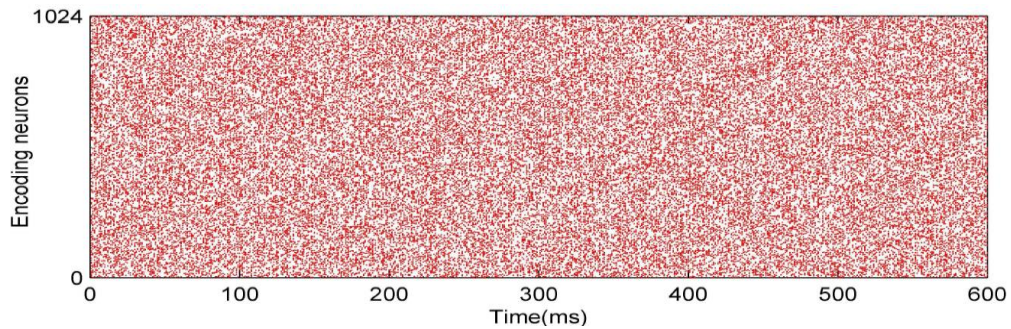### 3.1 The Performance of the Improved Latency-Phase Encoding

To demonstrate the improved latency-phase encoding, a 256×256 grayscale image (as the stimuli) was selected to show the encoding performance of the improved latency-phase encoding. In this simulation, the parameters were selected as: $a = 0.0032$, $t_{max} = 600$ $\omega = 100\pi$, $\Delta\varphi = 2\pi / 68$, $m = n = 6$.

(a) The original image (left), processing results in S1 (middle) and the results in C1 layer (right).
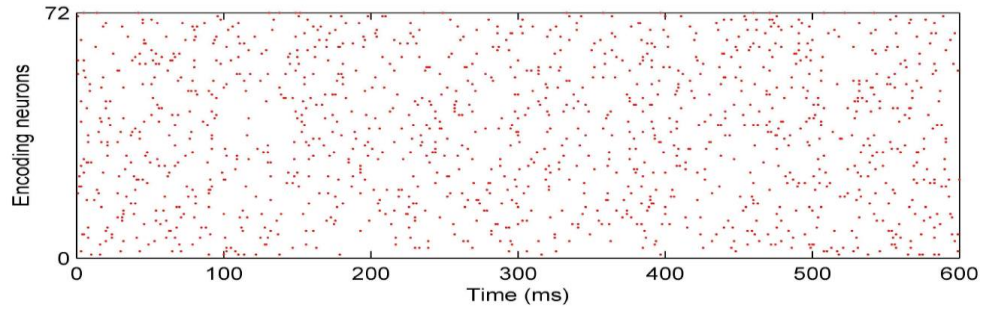
(b) Spikes encoded by improved latency-phase encoding method (each red dot denotes a spike)

(c) Spikes encoded by original latency-phase encoding method
Figure 3. The performance of the improved latency-phase encoding method.

9

First, the image information (intensity) was transmitted to layer S1. DoG (difference of Gaussian) filter was used in layer S1 to mimic that how neural processing in the retina of the eye extracts details from external stimuli. The processing result in S1 was shown in the middle panel of Fig. 3(a). S1 neurons competed with local neighbors through the MAX operation to strive for survival in C1 layer. Through the MAX operation, the original image was sparsely presented in C1 layer (right panel of Fig. 3(a)). The final activations of neurons in C1 were used to produce spikes through latency-phase encoding method. After encoding, the original image was converted into several sequences of spikes (Fig. 3(b)). Each spike generated by encoding neuron is denoted by a red dot. To make a comparison between our improved and the original latency-phase encoding methods, we showed the spike patterns which were encoded by original latency-phase encoding in Fig. 3(c). The number of encoding neurons needed in our method is 72, while the original method requires about 1024 encoding neurons. Thus, the improved latency-phase encoding is more efficient. Moreover, the spikes generated by our method was much less than that of the original method, which might decrease the complexity of processing the timing of spikes.
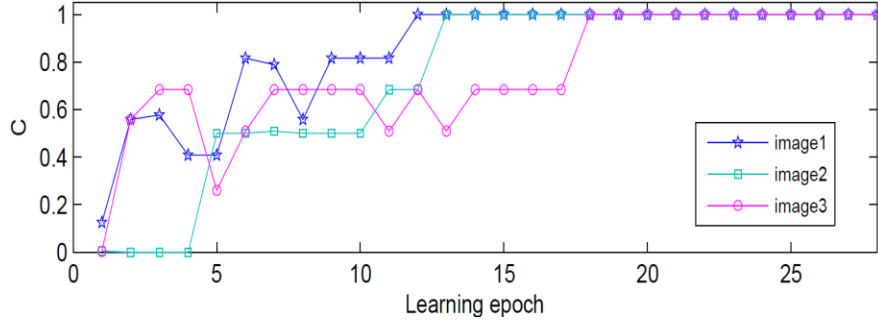
3.2 Learning Performance

To explore the learning capability of the proposed computational model on pattern recognition, we trained the proposed model to recognize three different images as shown in Fig. 4(a). After the encoding part, the three images were converted into three different spatiotemporal spike patterns. The three spike patterns were transmitted to the consecutive network for learning. To simplify the problem for a classification task, each desired pattern was defined as a spike train with three spikes, [50, 250, 450] ms for image1, [120, 320, 520] ms for image2, and [190, 390, 590] ms for image3.

(a) Three different input patterns
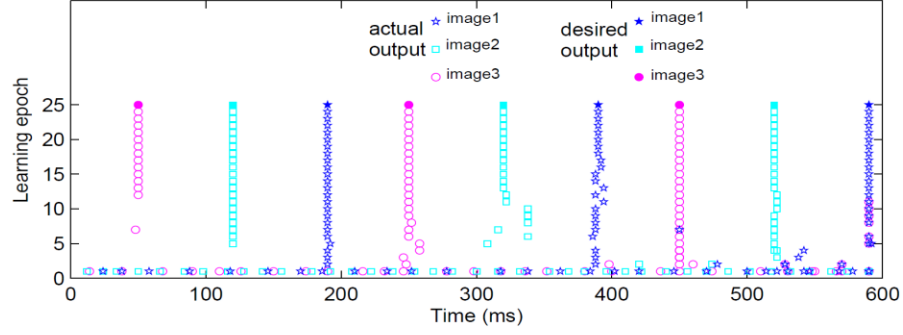
(b) Response of the output neuron.

(c) The correlation C between the desired and observed output spike trains

Figure 4. The learning performance of the proposed computational model.

Fig. 4(b) illustrated the evolution of the spike patterns generated by the output neuron for different input patterns. Before learning, the output neuron fired at random times, which was very different from the corresponding desired output spikes. After several learning epochs, spikes firing at undesired times disappeared and the observed output spikes approached the corresponding target spike patterns. After about 20 learning epochs, the neuron in the classification part can generate desired output spike trains for every input pattern. The measure C plotted as a function of learning epoch is shown in Fig. 4(c), which indicates that the initial values of C were close to zero, and the values of C reached to 1 after 20 learning epochs, which meant that these three images can

11

be recognized perfectly.

3.3 Generalization Capability

 Despite the achievement in the applications of spiking neuron, most of them assume noise-free condition for learning and testing. This assumption, though fairly general, ignores the fact that noise widely exists in spiking neural networks (SNNs) and the neural response can be significantly disturbed by noise [22]. To study the generalization capability of the proposed computational model, we tested the performance of the trained computational model on noisy data and noisy conditions. Noisy data was simulated by adding partially occluded, salt-and-pepper noise to the input images, and noisy condition was simulated by injecting Gaussian white noise voltage to the output neuron. The partially occluded noise was specified by the area of occluded n, the salt-and-pepper noise was specified by the noise intensity d, and the background noise was specified by the variance σ_b. For each kind of noise with different intensities, we tested the trained network with 50 noisy images, and the correlation C of a distance between the target and the observed output train was calculated. The experiment was performed according to two scenarios: the computational model was trained under noise-free conditions or the neuron was trained in the presence of noise.
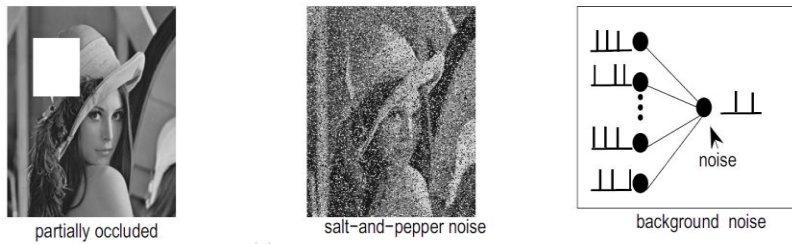
（α）Examples of different types of noise to the input images and networks

12
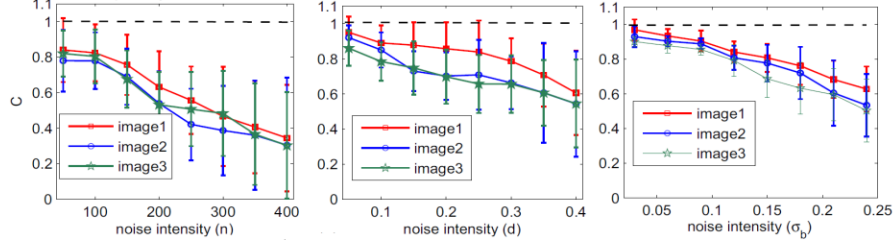
(b) Response of the computational model after noise-free training
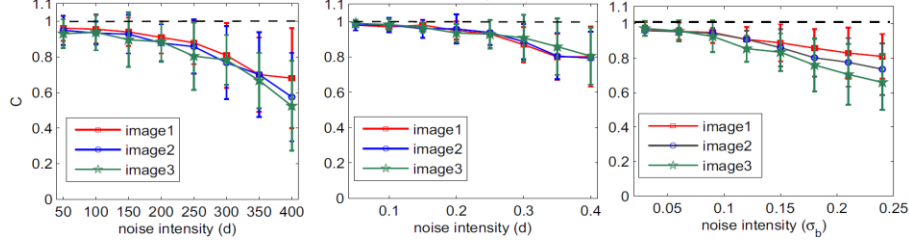
(c) Response of the computational model after noisy training

Figure 5. The robustness of the proposed computational model.

After training, we tested the performance of the generalization of the proposed model. The plots

of C obtained after deterministic training and noisy training were shown in Fig. 5(b) and Fig. 5(c),

respectively. In the first scenario, we observed that different types of noise had a great effect on the

learning results and the correlation C dropped almost linearly with the invariance of noise. In

contrast, the computational model was significantly less sensitive to noise if the noise training is

performed. The model was more robust to different types of noise, and the correlation C kept a high

value even in high level of noise. These results confirmed that the noisy training enables the neuron

to precisely and more reliably reproduce target firing patterns even for the relatively high level of

noise.

4. Discussion and Conclusion

In this paper, firstly we improved the latency-phase encoding method. By incorporating an

information extraction to preprocess the input patterns, the encoding neurons and spikes of our

method are much less than those of original latency-phase encoding method. As shown in Fig. 3, to

encode a 256×256 grayscale image, the number of encoding neurons needed in our method is 72,

while the original method requires about 1024 encoding neurons. Therefore, the proposed improved latency-phase encoding method is more efficient.

There seems to be at least two directions that could be followed to further improve the performance of the proposed model. First, very recent experiments suggested that the addition of extra layers, in agreement with the structure of the visual cortex, might provide a significant gain in performance. Second, in order to extract the main features of the external stimuli, we applied traditional filter techniques, which were only suitable for analogue values. Future work will focus on the extraction of features from sequences of spikes.

References

[1] Yan, R., Tee, K. P., Chua, Y., Li, H. & Tang, H, Gesture recognition based on localist attractor networks with application to robot control. *IEEE Comput. Intell,* 2012, 64-74.

[2] Keysers, D., Och, F. J., & Ney, H. Maximum entropy and Gaussian models for image object recognition. *In Joint Pattern Recognition Symposium.* 2002, 498-506.

[3] Liu, Y. F., Guo, J. M., & Lee, J. D., Halftone image classification using LMS algorithm and naive Bayes. *IEEE Transactions on Image Processing*, 2011, 2837-2847.

[4] Chi, M., & Bruzzone, L., Semisupervised classification of hyperspectral images by SVMs optimized in the primal. *IEEE Transactions on Geoscience and Remote Sensing*, 2007, 1870-1880.

[5] Yu, Q., Tang, H., Tan, K. C. & Li, H. Rapid feedforward computation by temporal encoding and learning with spiking neurons. *IEEE Trans. Neural Netw. Learn. Syst,* 2013, 1539-1552.

[6]Thorpe, S., Fize, D. & Marlot, C. Speed of processing in the human visual system. *Nature*, 1996, 520-522.

[7]Thorpe, S. J. & Fabre-Thorpe,M. Seeking categories in the brain. *Nature*, 2011, 260-262.

[8]Zhao, B., Ding, R., Chen, S., Linares-Barranco, B. & Tang, H. Feedforward categorization on AER motion events using cortex-like features in a spiking neural network. *IEEE Trans. Neural Netw. Learn. Syst*, 2014, 1963-1978.

[9] Perrett, D. I., Hietanen, J. K., Oram, M. W., Benson, P. J. & Rolls, E. T., Organization and functions of cells responsive to faces in the temporal cortex. *Phil. Trans. Royal Soc,* 1992, 23-30.

[10] Hung, C. P., Kreiman, G., Poggio, T. & DiCarlo, J. J., Fast readout of object identity from macaque inferior temporal cortex. *Science*, 2005, 863-866.

[11] Seung, H.S., Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. *Neuron*, 2003,1063-1073.

[12] Berry, M. J., & Meister, M.,Refractoriness and neural precision. *J. Neurosci.* 1998, 2200-2211.

[13] Uzzell, V. J., & Chichilnisky, E. J.，Precision of spike trains in primate retinal ganglion cells. *J. Neurophysiol.* 2004,780-789.

 [14] Gollisch, T., & Meister, M. Rapid neural coding in the retina with relative spike latencies. *Science*, 2008, 1108-1111.

[15] Reinagel, P., & Reid, R. C., Temporal coding of visual information in the thalamus. *J. Neurosci.* 2000, 5392-5400.

[16] Bair,W., & Koch, C. Temporal precision of spike trains in extra striate cortex of the behaving macaque monkey., *Neural Comput.* 1996, 1185-1202.

[17] Xiurui Xiu, et al. "Efficient Training of Supervised Spiking Neural Network via Accurate

Synaptic-Efficiency Adjustment Method." *IEEE Trans. Neural Netw. Learn. Syst*, 2016.

[18] Malu Zhang, Hong Qu, Ammar Belatreche & Xiurui Xie. EMPD: An Efficient Membrane Potential Driven Supervised Learning Algorithm for Spiking Neurons. *IEEE Transactions on Cognitive and Developmental Systems*, DOI: 10.1109/TCDS.2017.2651943 , 2017.

[19] Bermak, A., VLSI implementation of a neuromorphic spiking pixel and investigation of various focal-plane excitation schemes. *International Journal of Robotics and Automation*, 2004, 197-205.

[20] Malu Zhang, Hong Qu, Xiurui Xie , et al. Supervised learning in spiking neural networks with noise-threshold. *Neurocomputing,* 2017, 333-349.

[21] Ponulak, F., & Kasinski, A., Supervised learning in spiking neural networks with ReSuMe: Sequence learning, classification, and spike shifting. *Neural Comput.* 2010, 467-510.

[22] Xu, Y., Zeng, X., & Zhong, S. A new supervised learning algorithm for spiking neurons. Neural *Comput.*, 2013, 1472-1511.

[23] Hu, J., Tang, H., Tan, K. C., Li, H., & Shi, L. A spike-timing-based integrated model for pattern recognition, *Neural Comput.* 2013, 450-472 .

[24] M. Riesenhuber, & T. Poggio, Hierarchical models of object recognition in cortex, *Nature Neuroscience,* 1999, 1019–1025.

[25] Nadasdy Z. Information encoding and reconstruction from the phase of action potentials. *Frontiers in systems neuroscience*, 3: 6, 2009.

[26] Gerstner, W., & Kistler, W. M., Spiking neuron models: Single neurons, populations, plasticity. *Cambridge university press*, 2002.

[27] Schreiber, S., Fellous, J. M., Whitmer, D., Tiesinga, P. & Sejnowski, T. J. A new correlation-based measure of spike timing reliability. Neurocomputing, 2003, 925-931.