Supervised Learning of Single-Layer Spiking Neural Networks for Image Classification

Qiang Ma, Xianghong Lin and Xiangwen Wang

College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

Email: linxh@nwnu.edu.cn

Abstract. The traditional artificial neural networks encode information through the spike firing rate. Spiking neural networks fall into the third-generation artificial neural network models, which use the precisely timed spike trains to encode neural information. The computational models can accurately simulate the neural network activities of human brain, and provide powerful capabilities of signal processing to solve the complex problem. In this paper, we propose a supervised learning algorithm for single-layer spiking neural networks based on the spike train kernel function, which can implement the complex spatio-temporal pattern learning of spike trains. Furthermore, a pattern classifier based on single-layer spiking neural networks is constructed for image recognition problem. We test the learning performance of the proposed algorithm by the image classification task on the LabelMe dataset. The experimental results show that the proposed algorithm has got good image classification accuracy for the test dataset, and the different sizes of receptive fields influence classification accuracies significantly.

1. Introduction

The traditional artificial neural networks usually use the spike firing rate to encode external information [1]. However, the precisely timed spike train encoding strategy is used in networks of spiking neurons which are more effective than the traditional neural network models [2]. Spiking neural networks can simulate different neuronal signals and arbitrary continuous functions, so they are very suitable for processing the nervous signals in the brain. Spiking neural networks have been applied to many fields now, such as image recognition, natural language processing, and medical systems. In recent years, some progress has been made in the supervised learning algorithms for spiking neural networks [3]. Researchers have presented many supervised learning algorithms of networks of spiking neurons, such as the gradient descent rule [4], ReSuMe (Remote Supervised Method) based on spike timing-dependent plasticity [5]. In addition, Mohamed et al. [6] proposed SPAN (Spike Pattern Association Neuron) algorithm based on the kernel function convolution and use the Widrow-Hoff rule to adjust synaptic weights. Yu et al. [7] apply traditional Widrow-Hoff rule to spiking neural networks and proposed a PSD (Precise-Spike-Driven) supervised learning algorithm, in which only the input spike trains are converted.

Image recognition and classification problems are very important for the future development of spiking neural networks. Spiking neurons conform to the characteristics of biological neural networks and can be used to explain the activities of complex biological networks [8]. There are spatio-temporal pattern recognition problems are solved by the learning methods for spike times encoding in spiking...
neural networks [9][10]. Wu et al. [11] constructed a multi-layer neural network with integrate-and-fire neurons, which is applied to image processing such as image edge detection and image segmentation. The results show that their work has better image processing performance. Combining temporal encoding with Hebbian reinforcement learning method, Meftah et al. [12] put forward a spiking neural network model for image segmentation. Bawane et al. [13] proposed a network of leaky integrate-and-fire neurons for character recognition, extracted the features from the image using the extended histogram of gradients and other methods, and added the features to the classifier to classify the characters. Combining the K-means classifier, Mukhopadhyay et al. [14] proposed a spiking neural network classifier based on a memristive crossbar network. The results show that it has good fault tolerance due to the robustness of the spiking neural networks. In this paper, we propose a supervised learning method for single-layer spiking neural networks based on spike train kernel and apply it for image recognition and classification problems.

2. Spike Train Kernel Supervised Learning Rule

The spike train \( s = \{t^f \in \mathcal{F} : f = 1, \ldots, N \} \) represents the ordered trains of spike times fired by spiking neurons in time interval \( \mathcal{F} = [0, T] \). A spike train can be expressed as:

\[
s(t) = \sum_{j=1}^{N} \delta(t - t^j)
\]

where \( N \) is the number of spikes, \( \delta(x) \) is the Dirac delta function, and \( t^j \) is the fire time of the \( f \)th spike.

Due to the spike train is a set of discrete time, in order to facilitate the analysis and calculation, we choose a specific kernel function \( \kappa(t) \) to transform the discrete spike train into a unique continuous function:

\[
\tilde{s}(t) = s(t) * \kappa(t) = \sum_{j=1}^{N} \kappa(t - t^j)
\]

For convenience, we use the linear Poisson neuron model to describe the relationship between the input spike trains and the output spike train [15]. The transformed postsynaptic spike train \( \tilde{s}_p(t) \) can be expressed as the linear summation of the transformed presynaptic spike trains \( \tilde{s}_i(t) \) with corresponding synaptic weight \( w_{oi} \):

\[
\tilde{s}_p(t) = \sum_{i=1}^{N_i} w_{oi} \tilde{s}_i(t)
\]

where \( \tilde{s}_i(t) \) and \( \tilde{s}_p(t) \) is the transformed postsynaptic spike train and presynaptic spike train, \( N_i \) is the number of presynaptic neurons.

The synaptic weights of spiking neural networks are adjusted by supervised learning algorithm, and finally the network outputs the target spike trains. Firstly, we define the spike train error function according to the transformed actual output spike train \( \tilde{s}_a(t) \) and target output spike train \( \tilde{s}_d(t) \). The network error at time \( t \) is the square difference between \( \tilde{s}_a(t) \) and \( \tilde{s}_d(t) \) for all output neurons, it can be defined as:

\[
E(t) = \frac{1}{2} \sum_{i=1}^{N_o} \left[ \tilde{s}_a(t) - \tilde{s}_d(t) \right]^2
\]

where \( N_o \) is the number of output neurons. Therefore, the total error of spiking neural networks in the time interval \( \mathcal{F} \) is expanded as \( E = \int_{\mathcal{F}} E(t) dt \).

The delta update rule is applied to adjust all synaptic weights. The weight adjustment for presynaptic neurons \( i \) to postsynaptic neurons \( o \) is calculated as:
\[
\Delta w_{io} = -\eta \nabla E_{io}
\]

where \( \eta \) denotes the learning rate and \( \nabla E_{io} \) represents the gradient value of the spike train error function \( E \) for the synaptic weight \( w_{io} \). It can be expressed as the integral of the derivative of the error function \( E(t) \) with respect to the weight \( w_{io} \) in the time interval \( \Gamma \):

\[
\nabla E_{io} = \int_{\Gamma} \frac{\partial E(t)}{\partial w_{io}} dt
\]

Using the chain rule, the derivative of the error function \( E(t) \) to the synaptic weight \( w_{io} \) at time \( t \) can be derived:

\[
\frac{\partial E(t)}{\partial w_{io}} = \frac{\partial E(t)}{\partial \tilde{x}_i^o(t)} \frac{\partial \tilde{x}_i^o(t)}{\partial w_{io}}
\]

According to Eq. (4), the first partial derivative term on the right-hand part of Eq. (7) is calculated as:

\[
\frac{\partial E(t)}{\partial \tilde{x}_i^o(t)} = \frac{1}{2} \sum_{j=1}^{N_1} \left[ \tilde{x}_i^o(t) - \tilde{x}_j^o(t) \right] = \tilde{z}_i^o(t) - \tilde{z}_j^o(t)
\]

According to Eq. (3), the second partial derivative term of the right-hand part of Eq. (7) is computed as:

\[
\frac{\partial \tilde{x}_i^o(t)}{\partial w_{io}} = \frac{\partial}{\partial \tilde{x}_i^o(t)} \left[ \sum_{j=1}^{N_1} w_{io} \tilde{x}_j(t) \right] = \tilde{s}_i(t)
\]

Combining Eq. (8) and Eq. (9), the derivative of the error \( E(t) \) on the weight \( w_{io} \) at time \( t \) is equal to:

\[
\frac{\partial E(t)}{\partial w_{io}} = \frac{\partial E(t)}{\partial \tilde{x}_i^o(t)} \frac{\partial \tilde{x}_i^o(t)}{\partial w_{io}} = \left[ \tilde{z}_i^o(t) - \tilde{z}_j^o(t) \right] \tilde{s}_i(t)
\]

According to Eq. (6) and Eq. (10), the gradient \( \nabla E_{io} \) is calculated as follow:

\[
\nabla E_{io} = \int_{\Gamma} \tilde{z}_i^o(t) - \tilde{z}_j^o(t) \tilde{s}_i(t) dt = \int_{\Gamma} \tilde{z}_i^o(t) \tilde{s}_i(t) dt - \int_{\Gamma} \tilde{z}_j^o(t) \tilde{s}_i(t) dt
\]

Based on the derivation process discussed above, we get a spike train kernel supervised learning rule for single-layer spiking neural networks, the synaptic weight adjustment is expressed as:

\[
\Delta w_{io} = -\eta \left[ \int_{\Gamma} \tilde{z}_i^o(t) \tilde{s}_i(t) dt - \int_{\Gamma} \tilde{z}_j^o(t) \tilde{s}_i(t) dt \right] = \eta \left[ \sum_{k=1}^{N_2} \sum_{i=1}^{N_1} \kappa(t_{i'}^o - t_{i}^j) - \sum_{h=1}^{N_2} \sum_{i=1}^{N_1} \kappa(t_{i'}^o - t_{i}^j) \right]
\]

where \( \eta \) is the learning rate, and \( \kappa(\cdot) \) represents the kernel function.

### 3. Pattern Classifier for Image Recognition

#### 3.1. Spiking Neuron Model

The spiking neuron model is applied to analyse the network dynamic characteristics and learning process. Assuming that \( N_i \) is the total of synaptic inputs, \( N_j \) is the number of spikes in the \( i \)th spike train. \( w_{io} \) is synapse weight from \( i \)th input neuron to the \( o \)th output neuron. \( t_{ij} \) is the \( f \)th spike firing time of the input neuron \( i \). The potential of the neuron is defined as the sum of the postsynaptic potential induced by all input spikes from presynaptic neurons. When the internal state \( u(t) \) exceeds the neuron threshold \( \theta \), the output neuron firing a spike, and the internal state decreases instantly to the resting potential, this phase is called repolarization. The potential of the postsynaptic neuron at time \( t \) is calculated as:

\[
\Delta w_{io} = -\eta \left[ \int_{\Gamma} \tilde{z}_i^o(t) \tilde{s}_i(t) dt - \int_{\Gamma} \tilde{z}_j^o(t) \tilde{s}_i(t) dt \right] = \eta \left[ \sum_{k=1}^{N_2} \sum_{i=1}^{N_1} \kappa(t_{i'}^o - t_{i}^j) - \sum_{h=1}^{N_2} \sum_{i=1}^{N_1} \kappa(t_{i'}^o - t_{i}^j) \right]
\]
In Eq. (13), $\varepsilon(s) = (s/\tau)\exp(1-s/\tau)H(s)$ represents the spike response function, $\rho(s) = -\theta\exp(-s/\tau_R)H(s)$ represents the refractoriness function, where $\tau$ and $\tau_R$ are time constants, $H(s)$ is the Heaviside function.

### 3.2. Spiking Neural Network Architecture

Similar to the architecture of traditional artificial neural networks, spiking neural networks have feedforward, recurrent, and other network architectures. In this paper, we adopt a feedforward network architecture with input layer and output layer, which is divided into encoding stage and learning stage for image recognition and classification. Fig. 1 shows the network architecture for image recognition and classification.

![Figure 1. The network architecture for image recognition and classification](image)

The spiking neural network model is applied to image recognition and classification problems. In the coding stage, the latency-phase encoding method is used to transform the pixels of image receptive field into precisely timed spike trains, where the spike trains are used to represent external image stimuli information [16]. In the learning stage, each spike train corresponds to an input neuron and is inputted into the spiking neural network. The synaptic weights are learned by the spike train kernel supervised multi-spike learning method. The spiking neural network outputs the target spike pattern for given images.

### 4. Simulations and Results

In order to demonstrate the learning performance of the proposed algorithm, we perform a set of experiments in this section. The first experiment verifies the learning performance of learning algorithm for spiking neural networks on the Lena and Baboon grayscale images recognition tasks. In the second experiment, we select the outdoor road images and the outdoor city street images from LabelMe dataset [17] and perform an image classification problem to test the classification ability of the proposed supervised learning algorithm with different sizes of receptive fields.

In order to determine the accuracy of image recognition and classification, we introduce the similarity measure function $C$ to measure the similarity between the actual output spike train and the target output spike train after spiking neural network learning [18]. $C$ is defined as:

$$C = \frac{\tilde{s}_a^x(t) \cdot \tilde{s}_x^a(t)}{\|\tilde{s}_a^x(t)\| \cdot \|\tilde{s}_x^a(t)\|}$$

(14)
where $\mathbf{s}_o(t) \cdot \mathbf{s}_o(t)$ represents the inner product of $\mathbf{s}_o(t)$ and $\mathbf{s}_o(t)$, $|\cdot|$ represents the Euclidean norm of spike train. When two spike trains are completely identical, the value of $C$ equals to 1. When the correlation decreases gradually, the value of $C$ gradually tends to zero.

### 4.1. Image Recognition Task

In order to test the learning performance of spike train kernel supervised learning algorithm, we choose the Lena image and the Baboon image for recognition task. The original 256 × 256 image is encoded into spike trains by the latency-phase encoding with 8 × 8 receptive field. Each spike train corresponding to an input neuron inputted into the network and sets the target output spike train of two images. The target output spike train of the Lena image is $\{20, 40, 60\}$ ms, while the target output spike train of the Baboon image is $\{40, 60, 80\}$ ms.

Fig. 2 shows the learning results of spike train kernel supervised learning algorithm on Lena image and the Baboon image recognition task. Fig. 2(a) and Fig. 2(b) show the original Lena image and Baboon image, respectively. Fig. 2(c) and Fig. 2(d) show the learning process of Lena image and Baboon image, respectively, where * is the target output spike train, and · is the actual output spike trains of the neuron. It can be seen that with the increasing of iterations, the actual output spike trains are closer to the target output spike train for both Lena image and Baboon image. Finally, the output spike train is consistent with the target output spike train. Fig. 2(e) and Fig. 2(f) illustrate the change of recognition accuracy on Lena image Baboon image respectively in the learning process. It can be seen that with the increasing of iterations, the value of $C$ increases gradually for both Lena image and Baboon image. After 32 and 38 learning epochs respectively, the value of $C$ reached to 1. It indicates that the actual output spike train is equal to the target spike train. This experiment shows that the proposed method can well perform image recognition task.
4.2. Image Classification Experiment

In this experiment, we test the learning performance of our proposed learning method by an image classification problem. We choose the outdoor road images and the outdoor city street images from the LabelMe dataset in the experiment. Each kind of images includes 20 samples, in total of 40 samples. Fig. 3 shows some typical outdoor road images (Fig. 3(a)) and outdoor city street images (Fig. 3(b)). In our experiment, we choose 10 samples randomly from the outdoor road images and the outdoor city street images respectively to constitute the training set, and the remaining samples are constituted the testing set. The original 256 × 256 images are encoded into spike trains by the latency-phase encoding. In addition, we need to set the target output spike trains of two kinds of images. The target output spike train of the outdoor road images is set as {20, 40, 60, 80} ms, while the target output spike train of the outdoor city street images is set as {40, 60, 80, 100} ms.

Table 1 shows the image classification results on the LabelMe dataset with different size of receptive field. The number of input neurons is equals to the size of image divided by the size of receptive field. The size of receptive field takes 2×2, 4×4, 8×8, 16×16, 32×32 and 64×64 in total of six values. As seen from the table, with the increasing of the size of receptive field, both the training accuracy and the test accuracy are firstly increased, and then decreased. When the size of receptive field is 8×8, the training accuracy is 100% and the test accuracy is 94.77%. It shows that the receptive field
can't be too big or too small, the appropriate size of receptive field will obtain higher training accuracy and test accuracy. The experimental results show that the proposed spike train kernel supervised learning algorithm can be well applied to image classification problem and achieve high classification accuracy.

Table 1. The image classification results with different sizes of receptive fields.

| Number of input neuron | Receptive filed size | Training accuracy | Testing accuracy |
|------------------------|----------------------|-------------------|------------------|
| 16384                  | 2 × 2                | 94.33% ± 0.08     | 74.82% ± 0.17    |
| 4096                   | 4 × 4                | 100%              | 85.34% ± 0.12    |
| 1024                   | 8 × 8                | 100%              | 94.77% ± 0.04    |
| 256                    | 16 × 16              | 100%              | 93.33% ± 0.09    |
| 64                     | 32 × 32              | 100%              | 91.71% ± 0.11    |
| 16                     | 64 × 64              | 98.68% ± 0.06     | 83.18% ± 0.07    |

5. Conclusion
This paper uses the spike train kernel to transform the discrete spike train into a unique continuous function, and then proposes a spike train kernel supervised learning algorithm for single-layer spiking neural networks. The proposed supervised learning method is firstly successfully applied on Lena image and the Baboon image recognition task. It is also applied on an image classification problem on the LabelMe dataset with different size of receptive field. The experiment results show that the proposed algorithm can effectively solve image recognition and image classification problems. In future research, we will study more efficient supervised learning algorithms for spiking neural networks and apply them to more complex spatio-temporal pattern recognition problems.

Acknowledgment
This research is supported by the National Natural Science Foundation of China under Grant no. 61762080, and the Medium and Small Scale Enterprises Technology Innovation Foundation of Gansu Province under Grant no. 17CX2JA038.

References
[1] Haykin S S 2009 Neural Networks and Learning Machines. Upper Saddle River: Pearson Education
[2] Ghosh-Dastidar S and Adeli H 2009 Spiking neural networks International Journal of Neural Systems 19(4) 295-308
[3] Lin X H, Wang X W, Zhang N, et al. 2015 Supervised learning algorithms for spiking neural networks: a review Acta Electronica Sinica, 43(3) 577-586
[4] Florian R V 2012 The chronotron: a neuron that learns to fire temporally precise spike patterns PLoS One 7(8) e40233
[5] Ponulak F, Kasiński A 2010 Supervised learning in spiking neural networks with ReSuMe: sequence learning, classification, and spike shifting Neural Computation 22(2) 467-510
[6] Mohemmed, A, Schlebs S, Matsuda S and Kasabov N 2012 SPAN: spike pattern association neuron for learning spatio-temporal spike patterns International Journal of Neural Systems 22(04) 1659-85
[7] Yu Q, Tang H, Tan K C, et al. 2013 Precise-spike-driven synaptic plasticity: learning hetero-association of spatiotemporal spike patterns PLoS One 8(11) e78318
[8] Paugam-Moisy H and Bohte S 2012 Computing with spiking neuron networks Handbook of natural computing pp 335-376
[9] Kasabov N K 2014 NeuCube: A spiking neural network architecture for mapping learning and understanding of spatio-temporal brain data Neural Networks 52 62–76
[10] Tapson J C, Cohen G K, Afshar S, et al. 2013 Synthesis of neural networks for spatio-temporal spike pattern recognition and processing *Frontiers in Neuroscience* 7(7) 153

[11] Wu Q X, McGinnity T M, Maguire L P, et al. 2008 Processing visual stimuli using hierarchical spiking neural networks *Neurocomputing* 71(10-12) 2055-2068

[12] Meftah B, Lezoray O and Benyettou A 2010 Segmentation and edge detection based on spiking neural network model *Neural Processing Letters* 32(2) 131-146

[13] Bawane P, Gadariye S, Chaturvedi S, et al. 2018 Object and character recognition using spiking neural network *Materials Today Proceedings* 5(1) 360-366

[14] Mukhopadhyay A K, Chakrabarti I, Basu A, et al. 2018 Power efficient spiking neural network classifier based on memristive crossbar network for spike sorting application *arXiv preprint* 1802, 09047

[15] Carnell A and Richardson D 2005 Linear algebra for times series of spikes. *European Symposium on Artificial Neural Networks* pp 363-368

[16] Hu J, Tang H, Tan K C, et al. 2013 A spike-timing-based integrated model for pattern recognition *Neural Computation* 25(2) 450-472

[17] Russell B C, Torralba A, Murphy K P and Freeman W T 2008 LabelMe: A database and web-based tool for image annotation *International Journal of Computer Vision* 77(1-3) 157-173

[18] Schreiber S, et al. 2003 A new correlation-based measure of spike timing reliability *Neurocomputing* 52 925-931