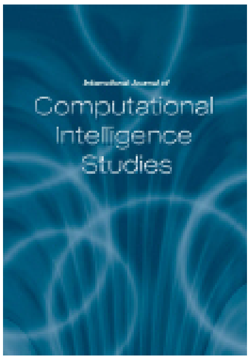


International Journal of Computational Intelligence Studies

ISSN online: 1755-4985 - ISSN print: 1755-4977
https://www.inderscience.com/ijcistudies

Research on spiking neural network in art visual image classification

Yiping Zhang

DOI: 10.1504/IJCISTUDIES.2023.10057739

Article History:

Received:	16 September 2022
Last revised:	14 March 2023
Accepted:	11 May 2023
Published online:	05 April 2024

Research on spiking neural network in art visual image classification

Yiping Zhang

Sino-German Institute of Design and Communication,
Zhejiang Wanli University,
Ningbo, 315100, China
Email: yipingzhangzwu4@yandex.com

Abstract: In the visual processing of artistic images, traditional CNN has a high resource demand, and SNN can solve this problem. The article selects SNN as the method for artistic visual image processing and combines it with CNN to simplify model training. After CNN adjustment and feedback adjustment algorithm processing, the classification accuracy of SNN can be improved. The results show that the accuracy of the adjusted CNN model is 80.25% and 79.60%, respectively, with an average training accuracy difference of 3.32%. Under the same pooling combination, the accuracy of the model with 11 and 12 iterations is 68.00% and 66.02%, respectively. The average classification accuracy of SNN is 78.80%, slightly lower than the adjusted CNN. SNN has a power consumption of approximately 0.0039 W per second in processing 742 images. The correlation classification method used in the article can reduce power consumption and has a high classification accuracy.

Keywords: spiking neural network; SNN; convolutional neural network; CNN; artistic visual image; accuracy; classification.

Reference to this paper should be made as follows: Zhang, Y. (2023) 'Research on spiking neural network in art visual image classification', *Int. J. Computational Intelligence Studies*, Vol. 12, Nos. 3/4, pp.206–222.

Biographical notes: Yiping Zhang is an Associate Professor. He graduated from Shanghai Jiaotong University in 2000 with a Bachelor's degree majoring Industrial Design BA. He studied at DFI College of Communication Art and New Media in Hamburg, Germany, majoring Communication Design, and was awarded MA degree of Design in Digital Media at University of Portsmouth. Currently, he is working in Zhejiang Wanli University at Sino-German Institute of Design and Communication. His research areas include digital media design, technology in digital interaction, user interface and user experiences. He has published eight academic papers in above area. He has hosted and participated nine research projects, and organised four international symposiums and conferences.

1 Introduction

Image classification is a type of image processing. By classifying images, different types of image features are obtained, thus providing a data basis for related research (Ma et al., 2017). Visual image classification is a kind of image classification, and there are many classification methods. Visual images are classified through Gaussian model-based metric forest, and the superiority of the algorithm is verified in related experiments such as flower image classification (Xu et al., 2018). Neural networks can also be used in visual image classification. During the development of neural networks, related image classification algorithms have been continuously improved, and spiking neural network (SNN) is one of them. This algorithm can be highly simulated. Biological neurons, with low power consumption, can be used in image classification. However, it is more difficult to model training. Convolutional neural network (CNN), on the other hand, can better handle image classification problems. After algorithm improvement, the depth of the model is deepened, and the model fitting degree becomes better. The classification accuracy in the cat and dog dataset is as high as 98.70%. Although the CNN algorithm is simple to train, it requires high energy consumption in practical applications. However, the SNN algorithm has high efficiency, but it has the shortage of training difficulties. Therefore, this paper combines the SNN algorithm with the CNN algorithm, and applies it to the classification of art vision images. To solve the problem of difficulty in training SNN models, in order to reduce the power consumption of model operation.

2 Related work

Yi and Zhu (2017) used the spectral visual codebook learning method to perform image classification and process large-scale local image patches. After verification, it is found that the method can obtain the nonlinear manifold of semantic image patches. Ji et al. (2020) encode visual image features based on a hybrid heterogeneous structure model. Through correlation verification, the method can improve the classification accuracy. And after being fused with the local features of the CNN, it also has a high classification accuracy. Based on the feature extraction algorithm of visual word bag, Mokhtar and Elnemr (2018) mine image features and study the image classification effect under different algorithms. From the related results, it is found that the discriminant analysis and Bayesian network algorithms perform better than other algorithms such as neural networks. Among them, in the COIL-100 dataset, the average accuracy of the Bayesian network is about 100.00%. Supriyanto et al. (2018) proposed a global weighting method based on intra- and inter-class term distributions for use in visual bag-of-words image classification. After relevant tests, it is found that the performance of this method is better, and it can better classify this type of images. On the basis of distributed computing, Hou et al. (2017) classify a large number of visual word bag images. After calculation, it is found that this method can reduce the calculation time and has better classification accuracy. Gadetska and Gorokhovatsky (2018) performed visual object image correlation analysis under the calculation of statistical methods. After the correlation level is determined, the correlation attributes and features are compared, and the effectiveness of the article application method is found after verification. Wei et al. (2017) combined low-dimensional data to give a fine-grained image classification method to solve the problem of redundancy in high-dimensional data and to distinguish subtle differences between fine-grained samples. During the relevant verification process, the feasibility of the method is confirmed, and it has high accuracy in the classification of unknown samples. In the classification of remote sensing hyperspectral images, Haut et al. (2019) et al. combined the visual attention drive technology with the residual network to reflect the spectral spatial information of the relevant data, and identify the most ideal features through masks to achieve classification. After dataset testing, the method used in the article has high classification accuracy.

Zhao et al. (2017) extracted bionic cortex features through SNN and evaluated the method under relevant data. Compared with other bionic models, this model has less running time and a test accuracy of 88.14% (Haut et al., 2019). When classifying sequence data, Tavanaei and Maida (2018) used a hybrid algorithm that combines SNNs and Hidden Markov Models, trained with spike-time-dependent plasticity rules. The results show that the hybrid method has good performance and can be further analysed. Bawane et al. (2018) used the SNN algorithm when identifying and classifying objects. After image preprocessing and feature extraction, they were sent to the classifier for classification operations. The results show that the algorithm has good classification performance. Lin et al. (2017) applied the SNN relative ranking learning rule, which is dynamic and adaptive in the adjustment of output spikes to adapt to the characteristics of input stimuli. During the correlation analysis, it is found that the rule has strong generalisation. Its trained network has high classification accuracy. Gautam and Singh (2020) combined the SNN algorithm and CNN, using univariate and multivariate data, to analyse the performance of convolutional spiking neural network (CSNN), the algorithm has good convergence, high classification accuracy.

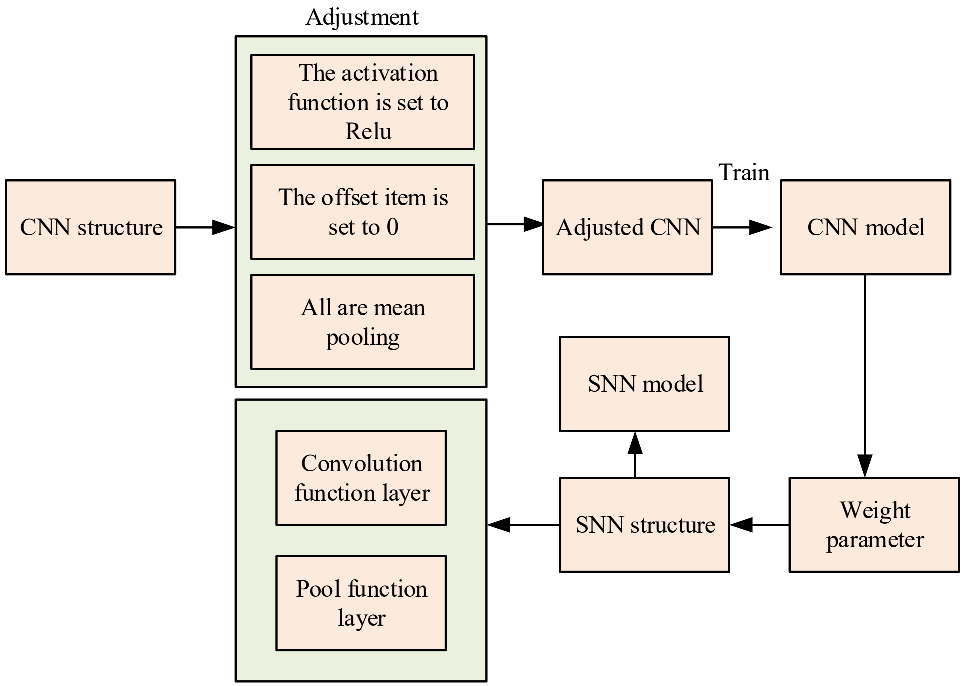
To sum up, in the process of visual classification, there are few studies on the analysis of artistic visual images. In this regard, the article takes artistic visual images as the object to carry out research. Considering the increase in power consumption of CNN algorithm due to the large amount of information in previous image processing, and according to the good convergence and adaptability of SNN algorithm in the application of image processing, this paper applies SNN algorithm to the classification of artistic visual images to reduce power consumption.

3 SNN-based art visual image classification

3.1 CNN mapping to SNN

With the rapid development of neural networks, people have gradually realised the ability of computer intelligent learning. Through intelligent learning, higher recognition accuracy can be achieved. Vision, as an important way for human beings to obtain external information, is of great practical significance to carry out visual recognition through computers. Taking art vision image classification as an example, CNN algorithm can show good application effect. Its model training is relatively simple, but its power consumption is large, which is not conducive to image classification. However, SNN has the characteristics of low power consumption of biological neurons, and it is less difficult to implement in hardware, and the neuron synapses it contains can connect neurons, and its weights can reflect the characteristics of the signal (Plessas et al., 2022; Jabbari and Karamati, 2022). This function is the same as the weight parameters obtained by CNN model training, therefore, the weight parameter of CNN can be used as the synaptic weight of SNN, and then the features of the input art visual image can be extracted from the SNN, and the relevant classification can be carried out to solve the problem of SNN training difficulty. When CNN is mapped to SNN, the relevant block diagram is shown in Figure 1.

Figure 1 Relevant block diagram (see online version for colours)



In the process of mapping CNN to SNN, because the mechanisms of the two algorithms are different, when the weight parameters of CNN act on SNN, several problems will occur, and these problems will reduce the classification accuracy of SNN. The first is that it is easy to output negative values in CNN. Because the LRelu function in this algorithm has a very small negative number mapping, the related operation process will appear negative values; also in this algorithm, by setting the bias term in the convolution layer, the accuracy is improved. increase, and the bias term and the input weights have negative values, which means that the output of the feature map may be negative. In addition, in CNN, subspace sampling is performed by maximum pooling to achieve the purpose of classification, but in SNN, two network layers are required to achieve the purpose of maximum pooling. In this process, the network structure will be complicated, the number of neurons will also increase, and the network accuracy will also be reduced. In order to reduce the loss of accuracy, the article mainly makes three adjustments: first, make the output of all layers of CNN positive; During training, set the bias item to 0 or not use the bias item; In SNN, the output is dimensionally reduced through mean pooling. In making the output of all layers of CNN positive, the article adjusts the LRelu function of CNN so that the output data is a non-negative value mapping, and its related expression is as formula (1).

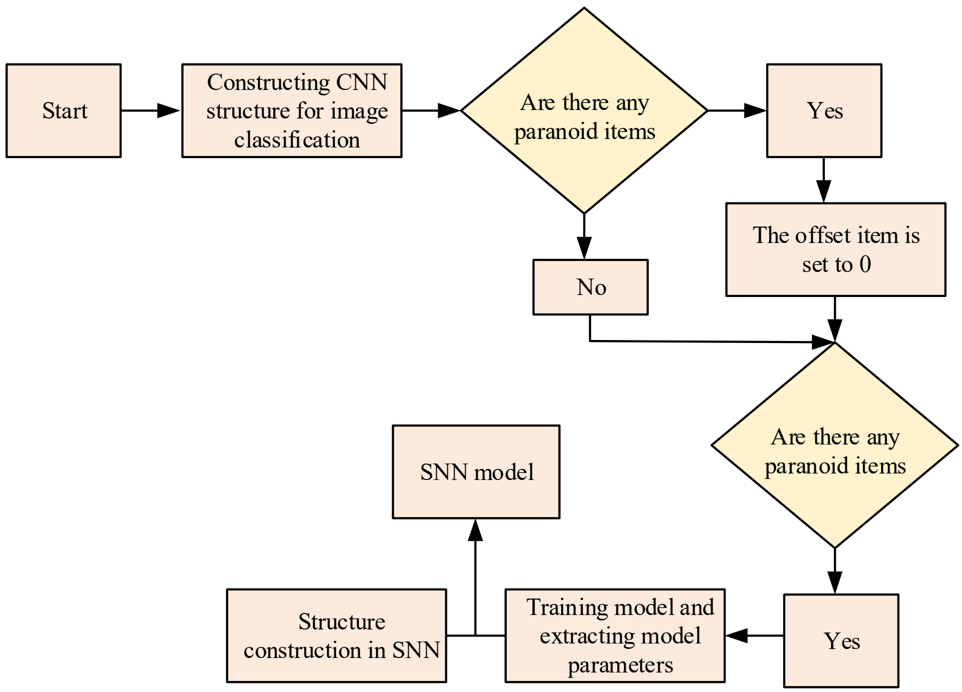
$$f(x) = \max(0, x) \tag{1}$$

In formula (1), x represents the independent variable. Depending on x the value, $f(x)$ it will change, and its mathematical expression is shown in formula (2).

$$\begin{cases} f(x) = 0(x \le 0) \\ f(x) = x(x > 0) \end{cases} \tag{2}$$

Then, on the basis of formula (2), the mapping range of Relu is determined to be [0, max]. For the problem that the bias item may be 0, by setting the bias item to 0 or not using the bias item during training. In SNN, the dimensionality reduction output is achieved through mean pooling, thereby avoiding the complexity of the SNN network structure. After adjustment, the flow chart of mapping CNN to SNN is shown in Figure 2.

Figure 2 Flow chart (see online version for colours)



First, construct the image classification CNN network structure, and then judge the bias item. When there is a paranoid item, set it to 0, and then judge the activation function; if not, judge the activation function directly. When the activation function is Relu, set it as Relu, then train the model to extract the model parameters, construct the structure in the SNN, and finally form the SNN model. The algorithm steps of mapping CNN to SNN are analysed. The main steps are four steps, which are to construct the CNN network structure for image classification, process the convolution layer, mean pooling and fully connected layer, and carry out the relevant parameters under the training dataset. Set, get the network model after training. According to the size of the convolution kernel, it can be known that the neurons in the output layer are connected to the neurons in the input layer; the weight parameters are extracted from the CNN model, and the weights of the convolution kernel are assigned to the synapses of the SNN. Then perform mean pooling. In this process, the synaptic weight is obtained according to the size of the pooling layer. For example, when the size of the pooling layer is 2*2, the corresponding synaptic weight is 0.25. Then, in the SNN, the neurons of the input layer and the output layer of the fully connected layer are correspondingly connected. In the SNN neuron model, the integral ignition model is adopted, which simulates neurons through a resistor-capacitance circuit (RC). Among them, the change process of neuron membrane potential is shown in formula (3).

$$I(t) = C_m \frac{dV_m(t)}{dt} \tag{3}$$

In formula (3), the membrane potential value is set V_m, C_m which means the membrane capacitance, which $I(t)$ means the input current, which t means the current passing time. The leakage current is introduced to correct the neuron current, so as to conform to the ion diffusion effect of biological neurons, and the relevant correction expression is shown in formula (4).

$$I(t) - \frac{V_m}{R_m} = C_m \frac{dV_m(t)}{dt} \tag{4}$$

In formula (4), the $\frac{V_m}{R_m}$ leakage current is expressed. The pulse signal generated by the synaptic neuron is the external current. The relationship between t the current generated by the $I(t)$ neuron at the research time and the pulse current i received by the neuron j is shown in formula (5).

$$I(t) = \sum_j w_{ij} \sum_f \alpha\left(t - t_j^{(f)}\right) \tag{5}$$

In formula (5), set the coefficient as α, set the synaptic weight as w_{ij}, and the corresponding time when neuron j emits the f pulse is $t_j^{(f)}$. In the SNN neuron model, after a time step, the membrane voltage $V(t)$ will be updated, and the relevant mathematical expression of its update is shown in equation (6).

$$V(t) = V(t-1) + L + X(t) \tag{6}$$

In equation (6), the constant is set as L, t and all synaptic inputs of neuron connections under time accumulate as $X(t)$. And according $V(t)$ to θ the relationship between and, $V(t)$ the value of, will change, and the relevant mathematical expression is shown in formula (7).

$$\begin{cases} IFV(t) \geq \theta, V(t) = 0 \\ IFV(t) < V\min, V(t) = V\min \end{cases} \tag{7}$$

In formula (7), if $V(t) \geq \theta$, the neuron is in an activated state, and there will be pulses generated; where θ represents the threshold. $V(t)$ can't go below the minimum value $V_{\min}$, usually $V_{\min} = 0$. In the SNN convolutional functional layer, if there are neurons (i, j), $X(t)$ the mathematical expression is shown in formula (8).

$$X(t) = \sum_{p,q}^{n} S_{p+i,q+j}(t) W_{p,q} \tag{8}$$

In formula (8), t the input pulse of the previous layer under time is $S_{p+i,q+j}(t)$, which (p, q) represents the neuron, and the size of the convolution kernel is n, when there is a pulse input, $S_{p+i,q+j}(t) = 1$; otherwise $S_{p+i,q+j}(t) = 0$, the weight of the convolution kernel shared by all neurons is $W_{p,q}$. In the mean pooling functional layer, if there are neurons (i, j), the neurons in this layer are obtained $X(t)$ as shown in formula (9).

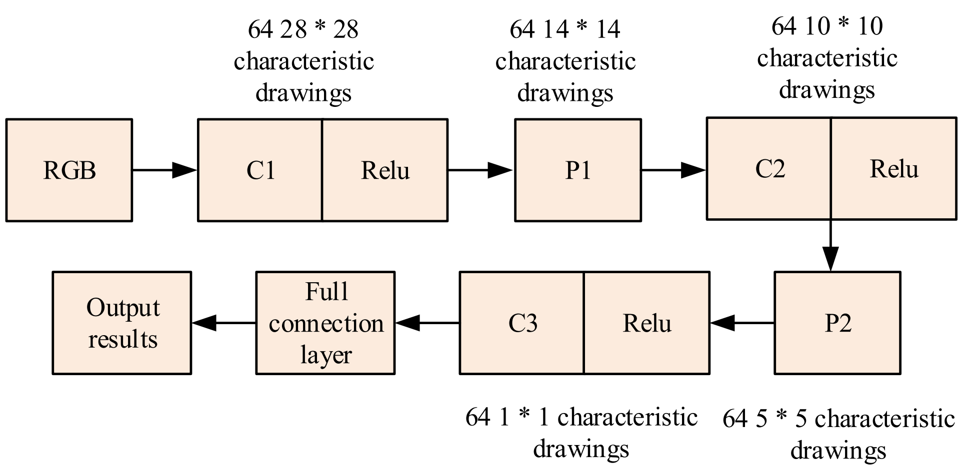
$$X(t) = \frac{1}{n^2} \sum_{p,q}^{n} S_{p+i,q+j}(t) \tag{9}$$

In formula (9), because the pooling layer is mean pooling and the size of the pooling layer is n, the synaptic weight parameter is set as $\frac{1}{n^2}$.

3.2 SNN visual image classification

By mapping CNN to SNN, the CNN network structure is adjusted to solve the problem of high energy consumption in the application of CNN. After the mapping is completed, According to the adjusted CNN network structure, set the network input to a colour art image, which is an RGB image, and the image size and number are 32*32 (3 images), and set the CNN three-layer convolutional layers to C1, C2, C3, the corresponding convolution kernel size and number are 5*5 (64), 5*5 (64), 5*5 (64), the network activation function is Relu, and the mean pooling layer is set to AVE-P1, AVE-P2, the corresponding pooling layer size and number are 2*2 (64), 2*2 (64), and the detailed adjusted CNN network structure is shown in Figure 3.

Figure 3 Detailed structure (see online version for colours)



When evaluating the adjusted CNN model, the accuracy index can be used for evaluation. Then perform SNN mapping. Since the structure of SNN is similar to CNN, before classifying artistic visual images in SNN, it is necessary to convert the image into a pulse signal through the pulse generator in the pulse generation layer, and set it as the network input. If you want SNN synaptic weights to correspond to CNN weight parameters, you need to use neuron connections in SNN to build convolutional layers, pooling layers, and fully connected layers. First of all, in the pulse generation layer, neurons can transmit information through the firing frequency, which will show randomness. With reference to probabilistic knowledge, the response function of the pulse neuron can be obtained $\rho(t)$ as shown in formula (10).

$$\rho(t)=\sum_{i=1}^{k}\delta(t-t_i) \tag{10}$$

In formula (10), the number of k pulses is, and the arrival time of pulses is t, which δ is the neuron response value. The number of pulses received at a smaller time interval is n shown in equation (11).

$$n=\int_{t1}^{t2}\rho(t)dt \tag{11}$$

In formula (11), d represents the differential, $t1$ and, $t2$ respectively represent the previous arrival time and the latter arrival time. And the instantaneous discharge frequency of the spiking neuron can be expressed as $\rho(t)$ the expectation $\gamma(t)$, and its mathematical expression is as shown in Equation (12).

$$\gamma(t)=\frac{dn(t)}{dt} \tag{12}$$

Mean value of $\gamma_M(t)$ the smaller time interval is $\rho(t)$ shown in equation (13).

$R^2_{adjusted}$

$$\gamma_M(t)=\frac{1}{M}\sum_{j=1}^{M}\rho_j(t) \tag{13}$$

In formula (13), M the number j represented, the $\rho(t)$ th $\rho(t)$ is $\rho_j(t)$. Assuming that different pulses are generated independently of each other, a pulse is generated in a time period, and k the probability of $P(n,t1,t2)$ the existence of a pulse in this time period is n shown in equation (14).

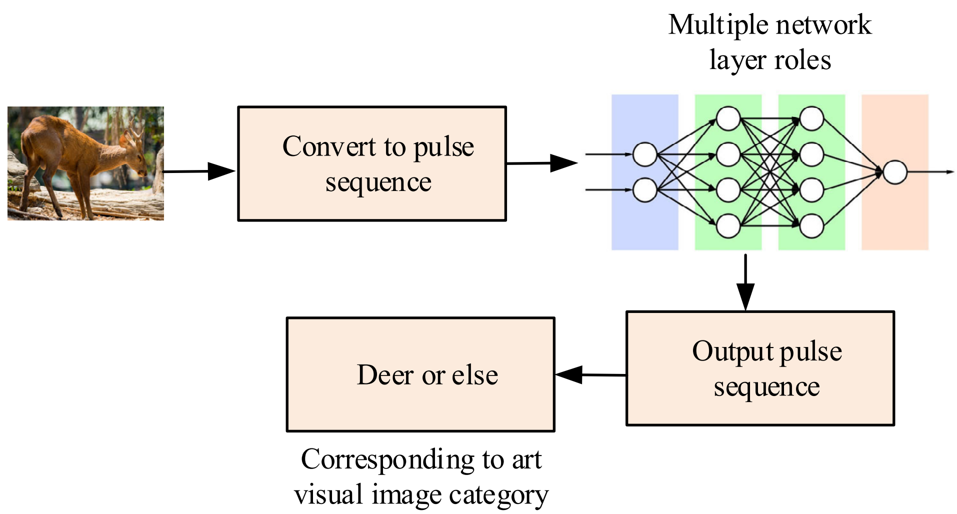
$$P(n,t1,t2)=\frac{k!}{(k-n)!n!}p^n q^{k-n} \tag{14}$$

In formula (14), $p=\frac{t1-t2}{T}$, $q=1-p$, T represents the time period, and the instantaneous discharge frequency is constant. When $k\to+\infty$ the average discharge frequency is constant, equation (14) can be transformed into equation (15).

$$P(n,t1,t2)=e^{-\gamma\Delta t\frac{\gamma\Delta t^n}{n!}} \tag{15}$$

In equation (15), e represents the average discharge frequency, and equation (15) can represent the pulse sequence coding method, and the pulse discharge process obeys the Poisson distribution. Therefore, through the pulse generation layer, Poisson pulse timing signals can be obtained. Then, a multi-layer feedforward SNN structure is used to represent a single neuron group as a single three-dimensional space structure. By setting the values of three parameters, the number of neurons in the neuron group and the corresponding position coordinates are represented. The neuron group presents In the shape of a cube, there is one neuron on each corner, and adjacent 8 neuron groups share the neurons on the corner equally, and form a neuron group spatial structure, and the layer composed of one-dimensional parameters can correspond to the convolution operation in CNN. Then, the neurons are connected through synapses to realise the transmission of CNN weight parameters and obtain the corresponding position output. Then in the input layer, this layer synapses connect all the neurons in the output layer, so that the connected neurons form a fully connected layer, thus constructing the SNN network structure and applying it in the classification of artistic visual images. As shown in Figure 4.

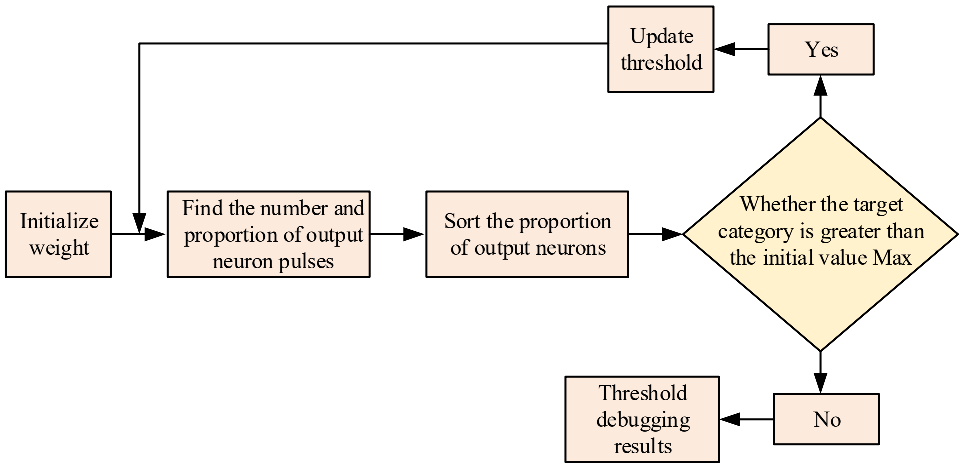
Figure 4 Classification process (see online version for colours)



In the SNN network structure, the artistic visual image is first processed and converted into a pulse sequence. Then, the screening of the pulse sequence is carried out. According to the difference of the pulse sequence, the selected neuron groups are different. Under the action of multiple network layers, an output pulse sequence is obtained, which can correspond to the categories of artistic visual images. Since the neuron group threshold will affect the SNN classification results, it is necessary to set the threshold to find the best threshold. In this regard, on the basis of the back-propagation algorithm, this paper proposes a feedback-based threshold adjustment algorithm to set the best threshold. The algorithm flow is shown in Figure 5 (Yu et al., 2022; Zhang et al., 2022).

In the algorithm flow, the threshold is combined with the final classification effect, and the initial threshold is set θ'. According to the threshold, the number of pulses of the corresponding category and the proportion of the number of pulses in the total number of output pulses are obtained, which are assigned to max1 variables, and then saved, and then the results are sorted. If the result category is not less than or equal to $\theta'_{\max}$, the corresponding result is set to max, and the threshold is updated. When the update result is less than max, stop the update, and then set the last threshold to the best threshold of the neuron group. It is worth noting that in the same algorithm flow, only one neuron group threshold can be set.

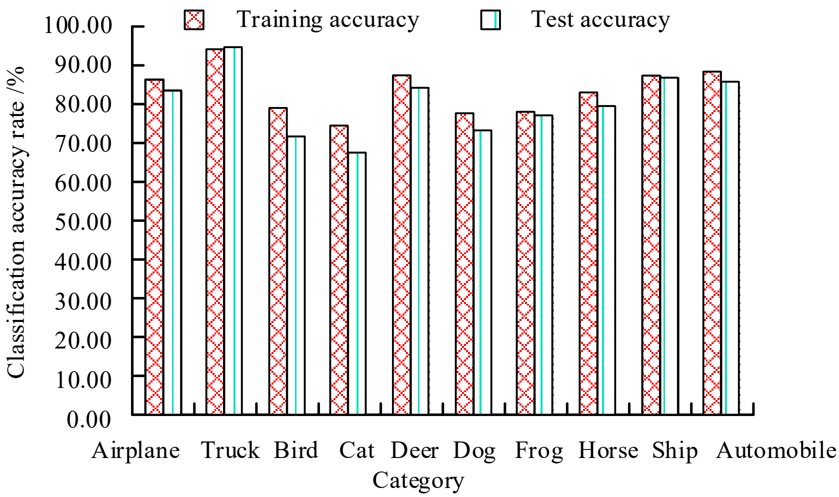
Figure 5 Algorithm flow (see online version for colours)



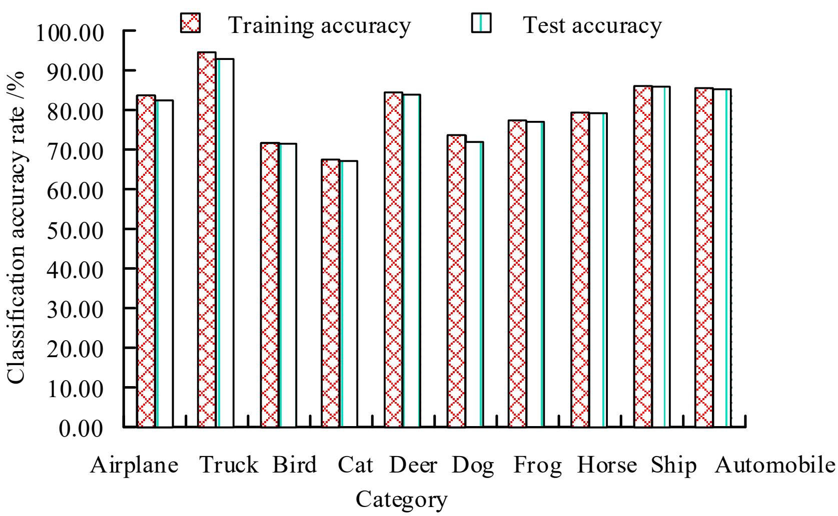
4 Experimental performance test

Evaluate the adjusted CNN model in the article, take the CNN model before adjustment as a comparison, and compare the size and number of convolution kernels of models C1, C2, and C3 are 5*5 (3), 5*5 (64)), 5*5 (64 pieces), the corresponding output feature map size and quantity are 28*28 (64 pieces), 10*10 (64 pieces), 1*1 (64 pieces), the pooling layer P1, The size of P2 is 2*2 and 2*2 in turn. The output neurons of the fully connected layers F1 and F2 are 64 and 10 respectively. The output layer finally outputs 10, and the output layer is 3*32*32 art visual images. The image is a colour RGB image. The three primary colours of the image are red, green and blue, which are superimposed to form different colours. During model training, the segmented learning rate is used, and the number of iterations is divided into four stages: 0–40,000, 40,000–50,000, 50,000–60,000, 60,000–70,000, and the learning rates of the corresponding stages are 0.001, 0.0001, 0.00001, and 0.000001 respectively. In the experimental environment, the selected operating system is Ubuntu 16.04, the main frequency of the Central Processing Unit (CPU) is 3.60GHz, the memory of 32.00GB is selected, the python version is 2.7, the Cafe is selected as the CNN framework, and the Cifar10 dataset is used. It is a dataset of common classification questions, and it has 10 categories. The training model is verified by the Cifar10 dataset, and the training accuracy and test accuracy of the two models are shown in Figure 6.

Figure 6 Training accuracy and testing accuracy of the model, (a) classification accuracy of CNN before adjustment (b) adjusted CNN classification accuracy (see online version for colours)



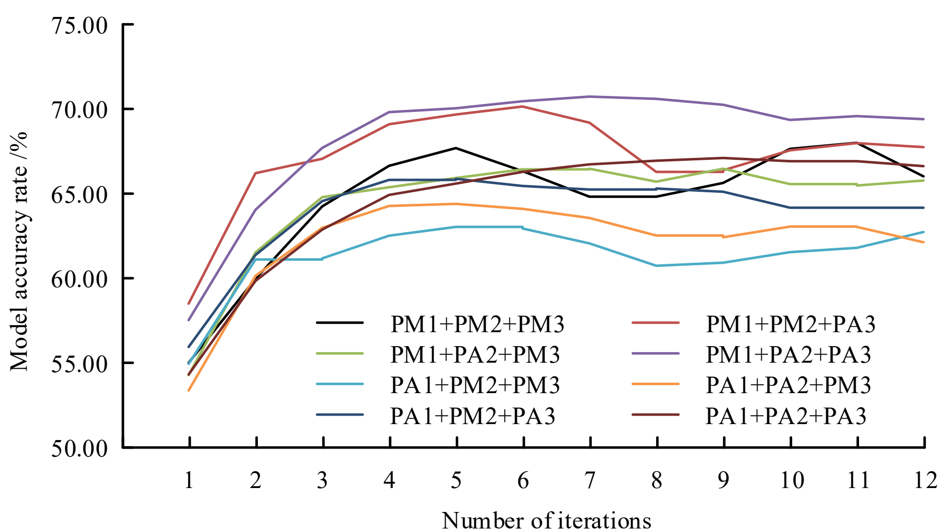
(a)



(b)

In Figure 6, there is a certain difference between the training accuracy and test accuracy of the CNN model before and after the adjustment; the training accuracy and test accuracy of the same CNN model are also different, but the change trend of the two accuracy rates is the same; different types of data The classification accuracy is different. In the pre-adjusted CCN model, the frog category training accuracy is 87.42%. The test accuracy rates of the CNN model before adjustment in the ship category and cat category are 86.70% and 67.5% respectively; while the test accuracy rate of the dog category under the same model is 73.20%, which is obviously lower than that of the truck category. The latter is accurate The rate is 94.70%. From the classification accuracy of the CNN model in different categories of art image data before adjustment, it can be seen that the truck category has the highest training accuracy and test accuracy, of which the training accuracy is 94.16%. The accuracy of the adjusted CNN model is analysed. The trend of the classification accuracy of the model in different categories of data is the same as that of the pre-adjusted CNN model. It also has the highest classification accuracy in the truck category, especially the training accuracy rate of 94.55%. It is 0.39% higher than the training accuracy of the CNN model before adjustment. The test accuracy of the adjusted CNN model on the deer category is 83.60%, which is 0.65% lower than the training accuracy of the same category. The average training accuracy and test accuracy of the CNN model before and after the adjustment are calculated. It can be seen that the average training accuracy and average test accuracy of the CNN model before the adjustment are 83.57% and 80.39% respectively, while the corresponding accuracy of the adjusted CNN model is 83.57% and 80.39%. The average training accuracy rate of the CNN model before and after adjustment is 3.32%. It can be seen that the adjusted CNN model can be used in SNN mapping. The learning rate is 0.001, and the weight attenuation is 0.004. The parameters of CNN are set. On the basis of adjusting the CNN model, a pooling layer is added. The three pooling layers are set to P1, P2, and P3 in turn, and each layer contains the maximum pool Set them as P M 1, P A 1, P M 2, P A 2, P M 3, P A 3 respectively, and study the effect of pooling combination and iteration times on model accuracy as shown in Figure 7.

Figure 7 Model accuracy (see online version for colours)



In Figure 7, the number of iterations and the pooling combination will have a certain impact on the accuracy of the model, especially the number of iterations. Under the same pooling combination, with the increase of the number of iterations, the model accuracy shows a large change. In the P M 1+P M 2+ P M 3 pooling combination, when the number of iterations is 5, the model accuracy rate is 67.69%, which is 1.30% higher than that when the number of iterations is 6. Under the same pooling combination, the iterative The accuracy rates of the models with times 11 and 12 are 68.00% and 66.02%, respectively, and the former is 1.98% higher than the latter. When the number of iterations is 1, the accuracy of the model under the P A 1+P A 2+ P A 3 pooling combination is 54.31%, which is lower than that under the P A 1+P M 2+ P A 3 pooling combination. 1.60%. In the pooling combination of P M 1+P M 2+ P A 3, the accuracy of the model when the number of iterations is 6 is the highest of 70.10%; the second is when the number of iterations is 5. When the number of iterations is 8, the accuracy of the model under the pooling combination of P M 1+P A 2+ P A 3 is 70.60%, which is higher than that when the number of iterations is 9, and the accuracy of the model when the number of iterations is 9 is 70.25 %. Under the same number of iterations, the model accuracy rates of different pooling combinations are different. The model accuracy rate of the A 1+P A 2+ P M 3 pooling combination when the number of iterations is 10 is 63.05%, and when the number of iterations is 3, the model accuracy is 63.05%. The model accuracy is 62.98%, the latter is 0.07% lower than the former. When the number of iterations is 6, the model accuracy rate under the P A 1+P M 2+ P M 3 pooling combination is 62.93, which is higher than the model accuracy rate under other pooling combinations. Only when the number of iterations is 1, the model accuracy rate Below 60.00%, the accuracy rate is 54.94. According to the relevant model accuracy data, it can be seen that the number of iterations has a greater impact on the model accuracy. Under the Cifar10 dataset, the size of the artistic visual image is 32*32, and the CNN model has a 3-layer network structure. There are three cases in which the number of convolution kernels in one layer is 16, 32, and 64. The number of convolution kernels is studied. The impact on the accuracy of CNN is shown in Table 1.

Table 1 Model accuracy under different convolution output combinations

Convolution output combination	16-16-16	32-32-32	64-64-64	16-16-32	16-16-64
Model accuracy (%)	71.21	74.04	76.07	72.23	73.05
Convolution output combination	32-32-64	16-32-64	64-32-16	64-32-32	32-32-16
Model accuracy (%)	74.42	74.63	74.21	74.95	74.68
Convolution output combination	32-16-16	64-16-16	16-64-16	16-64-32	
Model accuracy (%)	72.65	72.23	74.81	75.23	
Convolution output combination	16-32-16	32-64-32	32-64-16		
Model accuracy (%)	72.62	75.75	74.98		

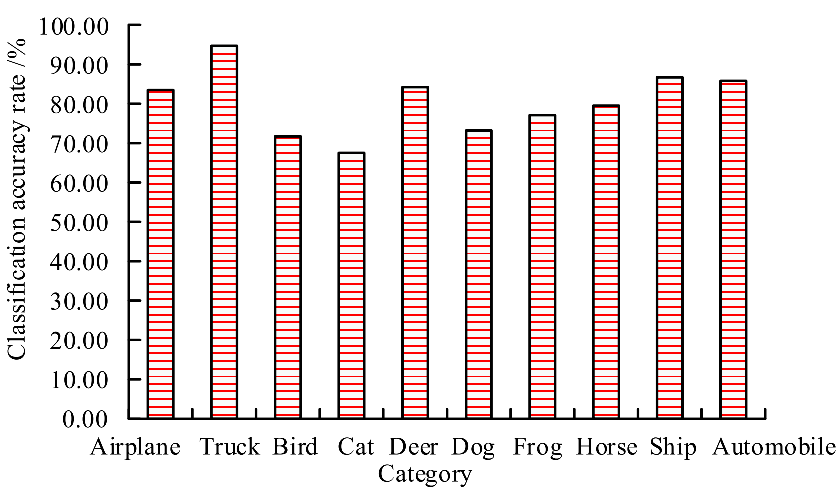
In Table 1, there are differences in the accuracy of the obtained models under different convolution output combinations. From the perspective of the same number of convolution output combinations, with the increase of the convolution kernel, the accuracy of the model continues to improve. When the number of convolution kernels in each convolution layer is the same, that is, when the number is 64, the accuracy of the model increases. The highest accuracy rate is 76.07%, while the model accuracy rate under the 16-16-16 combination is 71.21%, which is lower than the other two combinations. From the perspective of increasing number of convolution output combinations, the 16-16-32 combination has a lower model accuracy than the 16-16-64 combination, indicating that the third layer uses multiple outputs to achieve better results; 32-32-64 combination, 16-32-. The model accuracy rates of the 64 combinations are 74.42% and 74.63%, respectively. The former is 0.21% lower than the latter, and the latter has the highest accuracy. In the decreasing number of convolution output combinations, the 64-32-16 combination has a lower model accuracy than the 64-32-32 combination, and the accuracy rates of the two are 74.21% and 74.95%, respectively; The model accuracy of -16-16 combination is 0.42% lower. To study the effectiveness of the feedback-based threshold setting algorithm, see the SNN before setting the threshold as a comparison. After the threshold setting, the thresholds of C1–C3 in the SNN are 6.8, 0.1, and 0.1, respectively, and the thresholds of P1 and P2 are both 0.75. The two fully connected layers The thresholds are all 0.7, and the classification data comes from the test of the Cifar10 dataset. The classification results are shown in Table 2.

Table 2 Category classification: accurate quantity comparison

Model	*Category*	*Airplane*	*Automobile*	*Bird*	*Cat*	*Deer*
SNN	Category classification accurate quantity (Zhang)	80	74	67	68	69
H-SNN		82	84	72	60	72
Model	*Category*	*Dog*	*Frog*	*Horse*	*Ship*	*Truck*
SNN	Category classification accurate quantity (Zhang)	66	58	73	82	91
H-SNN		73	67	76	80	90

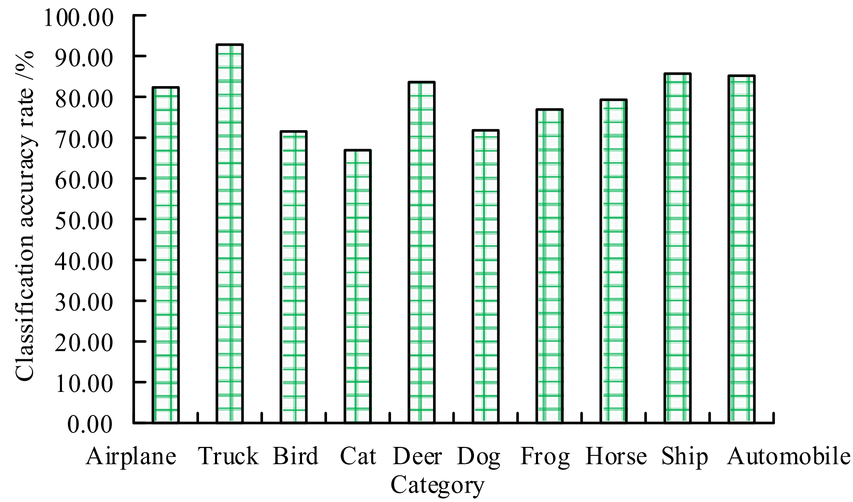
In Table 2, there are originally 100 images in each category, and the threshold is set as H-SNN. After model classification, the classification accuracy of different categories is different. In general, H-SNN obtains more accurate class classifications than SNN, and only in a few parts has fewer classes than SNN. In the bird category, the classification accuracy of H-SNN is 72, while the classification accuracy of SNN is 67, the former is 5 more than the latter. The H-SNN classification accuracy in the cat category is 68, which is 8 more than the SNN. According to the size of the classification accuracy, the classification accuracy of SNN in the truck category is at most 91, which is slightly higher than that of H-SNN. Set the CNN before adjustment as Q-CNN, and study the classification accuracy of CNN, Q-CNN and SNN under the test of Cifar10 dataset as shown in Figure 8.

Figure 8 Classification accuracy of three models, (a) Q-CNN classification accuracy (b) classification accuracy of CNN (c) SNN classification accuracy (see online version for colours)

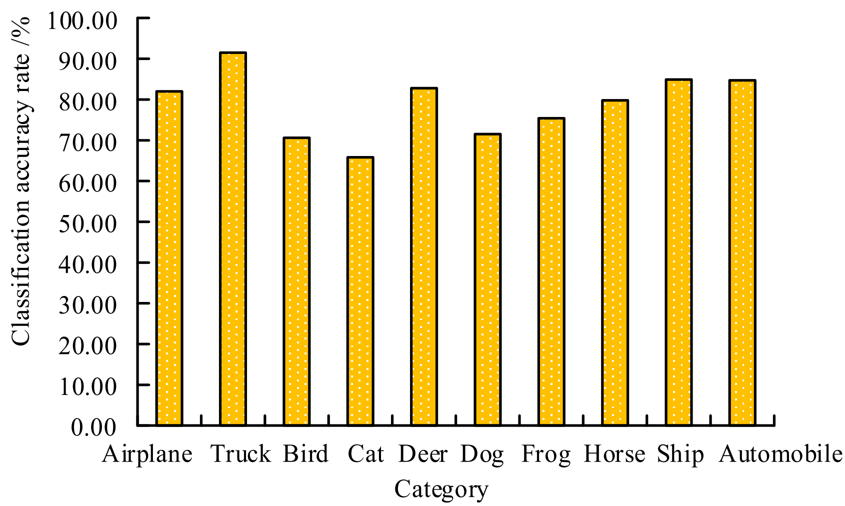


(a)

Figure 8 Classification accuracy of three models, (a) Q-CNN classification accuracy (b) classification accuracy of CNN (c) SNN classification accuracy (continued) (see online version for colours)



(b)



(c)

In Figure 8, the three models have different classification accuracies under different categories. The classification accuracy of Q-CNN under the airplane category is 83.50%, which is 11.80% higher than that of the bird category under the same model; the average classification accuracy of Q-CNN is 80.39, which is higher than the other two models. While the classification accuracy of CNN and SNN is the same as that of Q-CNN, the average classification accuracy of these two models is 79.60% and 78.80%, respectively. There is not much difference between the two. After the power consumption calculation, the SNN used in the article consumes 0.0039 W to process 742 art images per second, which is much smaller than Q-CNN.

5 Conclusions

In order to realise the classification of artistic visual images, the article uses SNN algorithm to classify them, so as to reduce energy consumption and improve classification efficiency. The adjusted CNN algorithm is used to prepare the CNN mapping SNN, and then the CNN weight parameters are passed to the corresponding synapses of the SNN, so as to realise the mapping of the CNN to the SNN, and solve the problem of difficulty in training the SNN model; and through the feedback-based threshold adjustment algorithm to find the best threshold of the neuron group to improve the classification accuracy. After analysis of relevant results, it is found that there is a certain difference between the training accuracy and test accuracy of the CNN model before and after the adjustment. The test accuracy of the adjusted CNN model in the deer category is 83.60%, which is 0.65% lower than the training accuracy of the same category. The average training accuracy difference of the CNN model before and after adjustment is 3.32%. In the P M 1+P M 2+ P M 3 pooling combination, when the number of iterations is 5, the model accuracy rate is 67.69%, which is 1.30% higher than that when the number of iterations is 6; under the same number of iterations, different pools The model accuracy of the pooling combination is different. The model accuracy of the A 1+P A 2+ P M 3 pooling combination is 63.05% when the number of iterations is 10. In the decreasing number of convolution output combinations, the model accuracy of the 64-32-16 combination is lower than that of the 64-32-32 combination, and the accuracy rates of the two are 74.21% and 74.95%, respectively. In the bird category, the accurate number of H-SNN classification is 72, while the accurate number of SNN classification is 67; in general, the accurate number of classifications obtained by H-SNN is more than that of SNN. The average classification accuracy of CNN and SNN is 79.60% and 78.80%, respectively. The method used in this paper can better classify artistic visual images. In the future, the article can also select other datasets for experiments and improve SNN adaptability.

References

Bawane, P., Gadariye, S., Chaturvedi, S. and Khurshid, A.A. (2018) 'Object and character recognition using spiking neural network', *Materials Today: Proceedings*, Vol. 5, No. 1, pp.360–366.

Gadetska, S.V. and Gorokhovatsky, V.O. (2018) 'Statistical measures for computation of the image relevance of visual objects in the structural image classification methods', *Telecommunications & Radio Engineering*, Vol. 77, No. 12, pp.1041–1053.

Gautam, A. and Singh, V. (2020) 'CLR-based deep convolutional spiking neural network with validation based stopping for time series classification', *Applied Intelligence*, Vol. 50, No. 3, pp.830–848.

Haut, J.M., Paoletti, M.E., Plaza, J., Plaza, A. and Li, J. (2019) 'Visual attention-driven hyperspectral image classification', *IEEE Transactions on Geoence and Remote Sensing*, Vol. 57, No. 99, pp.8065–8080.

Hou, C., Zhang, Q., Wang, B., Chang, P. and Sun, S. (2017) 'Image classification approach of bag of visual words model based on Hadoop', *Tianjin Daxue Xuebao*, Vol. 50, No. 6, pp.643–648.

Jabbari, M.B. and Karamati, M.R. (2022) 'The effects of temperature on the dynamics of the biological neural network', *Journal of Biological Physics*, Vol. 48, No. 1, pp.111–126.

Ji, Z., Yang, Y., Wang, F., Xu, L. and Hu, X. (2020) 'Feature encoding with hybrid heterogeneous structure model for image classification', *IET Image Processing*, Vol. 14, No. 10, pp.2166–2174.

Lin, Z., Ma, D., Meng, J. and Chen, L. (2017) 'Relative ordering learning in spiking neural network for pattern recognition', *Neurocomputing*, 31 January, Vol. 275, pp.94–106.

Ma, L., Li, M., Ma, X., Cheng, L., Du, P. and Liu, Y. (2017) 'A review of supervised object-based land-cover image classification', *ISPRS Journal of Photogrammetry & Remote Sensing*, August, Vol. 130, pp.277–293.

Mokhtar, S. and Elnemr, H. A. (2018) 'A comparative study of data mining approaches for bag of visual words based image classification', *Journal of Computer Science*, Vol. 14, No. 1, pp.53–66.

Plessas, A., Espinosa-Ramos, J.I., Parry, D., Cowie, S. and Landonl, J. (2022) 'Machine learning with a snapshot of data: Spiking neural network 'predicts' reinforcement histories of pigeons' choice behavior', *Journal of the Experimental Analysis of Behavior*, Vol. 117, No. 3, pp.301–319.

Supriyanto, C., Nugroho, H.A. and Adji, T.B. (2018) 'A global weighting scheme based on intra-class and inter-class term distributions in bag-of-visual words image classification', *IAENG International Journal of Computer Science*, Vol. 45, No. 2, pp.228–236.

Tavanaei, A. and Maida, A.S. (2018) 'Training a hidden markov model with a bayesian spiking neural network', *Journal of Signal Processing Systems*, Vol. 90, No. 2, pp.211–220.

Wei, J., Wu, J. and Meng, M. (2017) 'Fine-grained image classification with low-dimensional visual feature embedding', *Jisuanji Fuzhu Sheji Yu Tuxingxue Xuebao/Journal of Computer-Aided Design and Computer Graphics*, Vol. 29, No. 12, pp.2330–2335.

Xu, Y., Zhang, Q. and Wang, L. (2018) 'Metric forests based on Gaussian mixture model for visual image classification', *Soft Computing: A Fusion of Foundations, Methodologies and Applications*, Vol. 22, No. 2, pp.499–509.

Yi, H. and Zhu, W. (2017) 'Learning visual codebooks for image classification using spectral clustering', *Soft Computing*, Vol. 22, No. 6, pp.1–10.

Yu, L., Xie, L., Liu, C., Yu, S., Guo, Y. and Yang, K. (2022) 'Optimization of BP neural network model by chaotic krill herd algorithm – ScienceDirect', *Alexandria Engineering Journal*, Vol. 61, No. 12, pp.9769–9777.

Zhang, L., Wang, C., Fang, M. and Xu, W. (2022) 'Spectral reflectance reconstruction based on bp neural network and the improved sparrow search algorithm', *IEICE Transactions on fundamentals of Electronics, Communications and Computer Sciences*, Vol. 105, No. 8, pp.1175–1179.

Zhao, B., Ding, R., Chen, S., Linares-Barranco, B. and Tang, H. (2017) 'Feedforward categorization on aer motion events using cortex-like features in a spiking neural network', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 26, No. 9, pp.1963–1978.