Journal of Artificial Intelligence and Data Mining (JAIDM), Vol. 11, No. 2, 2023, 247-257.

Shahrood University of Technology

Journal of Artificial Intelligence and Data Mining (JAIDM)
Journal homepage: http://jad.shahroodut.ac.ir

Research paper

Digit Recognition in Spiking Neural Networks using Wavelet Transform

Hedyeh Aghabarar, Kourosh Kiani* and Parviz Keshavarzi

Faculty of Electrical and Computer Engineering, Semnan University, Semnan, Iran.

Article Info

Article History:
Received 15 January 2023
Revised 04 April 2023
Accepted 01 May 2023

DOI:10.22044/jadm.2023.12613.2415

Keywords:
Wavelet transform, digit recognition, convolutional SNN, constant-current-LIF encoding.

**Corresponding author: kourosh.kiani@semnan.ac.ir (K. Kiani).*

Abstract

Nowadays, given the rapid progress in pattern recognition, new ideas such as theoretical mathematics can be exploited to improve the efficiency of these tasks. In this paper, the Discrete Wavelet Transform (DWT) is used as a mathematical framework to demonstrate hand-written digit recognition in spiking neural networks (SNNs). The motivation behind this method is that the wavelet transform can divide the spike information and noise into separate frequency sub-bands, and also store the time information. The simulation results show that DWT is an effective and worthy choice, and brings the network to an efficiency comparable to previous networks in the spiking field. Initially, DWT is applied to MNIST images in the network input. Subsequently, a type of time encoding called constant-current-Leaky Integrate and Fire (LIF) encoding is applied to the transformed data. Following this, the encoded images are input to the multi-layer convolutional spiking network. In this architecture, various wavelets are investigated, and the highest classification accuracy of 99.25% is achieved.

1. Introduction

Due to the rapid progression of science and technology in various recognition fields, research continues on various platforms in order to improve the accuracy of network performance and inference. A large number of these studies are concerned with the initial data pre-processing stage including the transformation and coding of input images. In this process, the amount of data to be processed is reduced, redundant information is discarded, and only valuable and critical features are extracted from the input data and used in the network.

Approaches based on multi-resolution or multi-channel analysis such as Gabor filters [1] and wavelet transform [2-6] can be used to extract features from images in pattern recognition applications. Gabor filters need to fine-tune the parameters on different scales. Since their output is not orthogonal, a significant correlation may be observed between the obtained features. Most of these problems can be resolved by means of the wavelet transform, a time-frequency analysis method widely used in signal and image processing, singularity detection, data compression, and denoising. Instead of representing signals directly in the time-frequency space as in the Fourier transform, the wavelet transform displays them in the time-scale space. The entire frequency information is obtained in the Fourier transform; therefore, the overall Fourier coefficients can be affected by local changes in the signal. The wavelet transform, however, performs very well in accurately estimating and separating high- and low-frequency components with short- and long-term resolution, respectively. While the low-frequency coefficients approximate the signal, the high-frequency coefficients show its details, which is called the multi-level decomposition property or coarse-to-fine strategy. The wavelet transform performs this process and does not need a large number of calculations, as is the case with the human visual system.

SNNs have been introduced as the third generation of neural networks, a kind of artificial neural network (ANN), that are more precisely inspired by biological neural networks. SNNs can effectively imitate the intelligent behavior of the brain using spiking neurons. The concept of time is also considered in this model. As for ANNs, the output of each neuron is calculated and updated in every single iteration. In SNNs, on the other hand, each neuron generates a spike only when its membrane potential or membrane voltage exceeds a threshold. In biological neural networks, the excitable neuron transmits information to other neurons by generating an electrical signal or spike. The test accuracy in SNN is lower than in ANN. In order to increase accuracy, the network can be extended to deep architectures [7-10]. One type of deep spiking platform is the spiking convolutional neural network (spiking CNN), which is used in this article. For further comprehension, it is recommended to review articles on deep learning in SNNs [11-15].

As mentioned earlier, information in an SNN is processed and transmitted through spike trains. Therefore, the conversion of the analog input data into a spike train is the first step in implementing an SNN that cannot be ignored and must be done utilizing information encoding methods. The input signal must be pre-processed before encoding using one or more methods. Nowadays, it has been shown that the wavelet transform can be widely used as a useful tool to extract the main features of images. This observation motivated us to use DWT for the input signal pre-processing in the proposed spiking network. In this way, the network can efficiently extract key features and information from images and use them to perform the recognition process with high accuracy.

The rest of the paper is organized as what follows. In Section 2, recent related studies are summarized. Section 3 provides a detailed description of the proposed model. Section 4 contains the evaluation results. Finally, the work is concluded in Section 5.

2. Related Works

In most systems with arbitrary architecture, learning method, and simulation framework, pre-processing of input data is required to extract essential features. Here, the focus is on the spiking structure. For this reason, a review of some related studies has been carried out on pre-processing of input signals in spiking networks.

In the studies done in the field of pattern recognition in SNNs, the wavelet transform has rarely been used for pre-processing of input images. For instance, in [16], iris data is embedded in the content of a digital image to authenticate the owner's identity. An SNN called Pulse Coupled Neural Network (PCNN) algorithm is used to enhance the contrast of the iris image and to separate its boundaries from the human eye. The PCNN is based on the visual cortex of a cat, and has an entirely different structure than the convolutional SNN we have used. In this algorithm, Daubechies wavelet extracts the iris texture features in the pre-processing stage. Afterward, the iris code is inserted and extracted into digital images using DWT. The quality estimation parameter called the correction rate has reached the highest value of 0.9775.

In [17], to simulate the human visual system, a different SNN from the proposed method is used. Using Mallat wavelet transform on different scales, the network classified the textures by extracting the essential features of input images from the Brodatz album and obtained an average accuracy of 98.938. Another difference is the use of a Fast Wavelet Transform (FWT) here, where the input is down-sampled after passing through the filter each time. Furthermore, after performing the FWT, the normalized energy in each sub-channel is used as extracted features.

In [18], a shallow convolutional SNN (see Figure 1) performs object recognition in natural images, different from the proposed method, using Reward-modulated Spike-Time-Dependent-Plasticity (R-STDP) learning. Compared to the conventional STDP learning, which extracts repetitive features, the network extracts distinct visual features. In the first layer, four Gabor filters with different orientations are used to pre-process the input image. The time encoding is applied, yet the computed value of each filter is converted into only one spike with a latency proportional to the inverse of the output value of the filter, using an intensity-to-latency conversion mechanism. In [19, 20], the same intensity-to-latency procedure is performed in a deep convolutional SNN (as shown in Figure 2) for digit recognition. The layers are trained using R-STDP or STDP learning. Instead of DWT, DoG filters are applied to the input data in the first layer. The network has achieved 97.2% accuracy on MNIST.

The mentioned intensity-to-latency time coding and STDP learning are also used in [21, 22]. In [21], a 5-layer SNN, which used the ETH dataset and 3D-Object for object recognition tasks, is presented. Next, in [22], a spiking DNN consisting of multiple convolutional and pooling layers designed by Kheradpisheh was tested on Caltech, ETH-80 and MNIST datasets and achieved 99.1%, 82.8%, and 98.4% accuracy, in turn. The two articles are different from our study in which DWT is used for feature extraction, whereas in [21], the input images are convoluted with 4 Gabor filters to detect bars and edges, and in [22], the DoG filters are used in the first layer to detect contrasts. Moreover, in [23], based on [21, 22], a deep spiking CNN was presented using a combination of STDP learning, backpropagation and employing DoG filters in the network input. The system has achieved 98.5% and 85.3% accuracy on MNIST and EMNIST datasets, respectively. Table 1 shows a summary of recent works, their parameters, and network accuracy, which has been explained before.

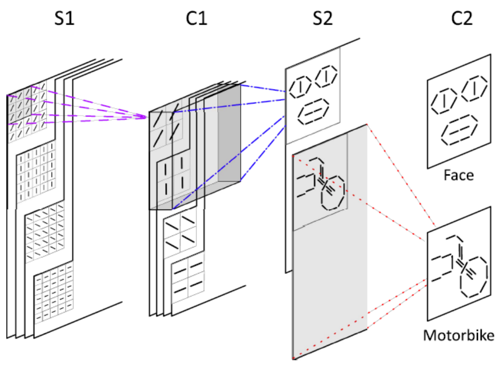

Figure 1. Network architecture in [18].

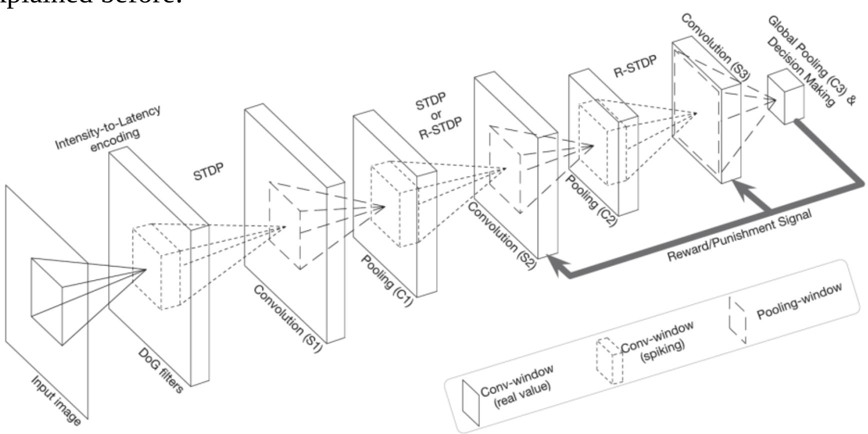

Figure 2. Network architecture in [19].

Table 1. Summary of the recent related works in SNNs.

Ref.	Architecture	Preprocessing	Dataset	Performance
(Hassanien *et al*., 2009) [16]	SNN	Wavelet transform	Iris images	97.75
(Zhang *et al*., 2015) [17]	SNN	Fast Wavelet Transform	Brodatz album	98.938
(Mozafari, Ganjtabesh, Nowzari-Dalini, Thorpe, *et al*., 2019) [19]	Convolutional	DoG filtering	MNIST	97.2%
(Kheradpisheh *et al*., 2018) [22]	Convolutional	DoG filtering	MNIST Caltech ETH-80	98.4% 99.1% 82.8%
(Mozafari, Ganjtabesh, Nowzari-Dalini, & Masquelier, 2019) [20]	Convolutional	DoG filtering	MNIST	96.9%
(Vaila *et al*., 2022) [23]	Convolutional	DoG filtering	MNIST	98.5%
(Mozafari *et al*., 2018) [18]	Convolutional	Gabor filtering	Caltech ETH-80	98.9% 89.5%
(Kheradpisheh *et al*., 2016) [21]	Convolutional	Gabor filtering	ETH-80	81.1%

3. Proposed Method

3.1. SNN architecture

Our proposed convolutional SNN architecture is shown in Figure 3. Firstly, in a pre-processing layer, the pixels of each input image are multiplied by a discrete wavelet filter, and a feature map is created whose cells are approximation coefficients that are obtained from the wavelet transform. In an encoding layer, this feature map is then converted into spike trains through a time encoding method. Subsequently, the spike trains are converted into spike maps over time by two hierarchical convolutional and pooling layers. The convolutional layers consist of LIF neurons. Besides, a max-pooling layer with the same number of kernels is added after each convolutional layer to reduce the size of the maps and remove redundant information.

Next, a fully connected (FC) layer and an integrated layer for readout are embedded in the system structure. These two layers are composed of LIF and ReLU neurons, in turn. In the end, there is a decoding layer with 10 output neurons containing the same number of classes as input images. For this purpose, the maximum value of the neurons membrane potential is examined over time and their softmax is calculated. The gradient descent method is used to train the network. The proposed network is tested on the MNIST dataset. This network has been evaluated on different wavelet functions, by changing the threshold voltage of network neurons, and inverting the input image. The results indicate that the network can compete against its spiking counterparts and achieve comparable accuracies when different wavelets and threshold voltages are used.

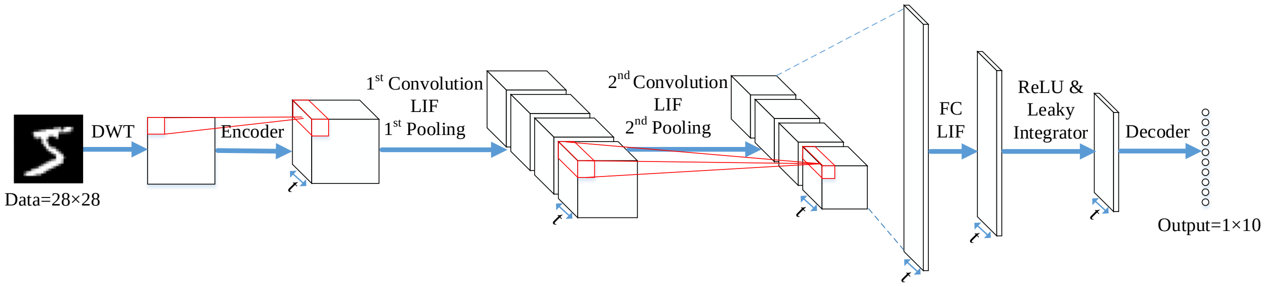

Figure 3. Proposed spiking network. SNN with DWT pre-processing, an encoder, two hierarchical LIF convolutional and spike-based max-pooling layers, a LIF FC layer, a non-spiking ReLU leaky integrator, and finally, a decoder.

3.2. Neural dynamics

To model the behavior of spiking neurons in the proposed network, the LIF neuron model was selected from various models presented for the neuron, including Hodgkin-Huxley (HH) [24] and Izhikevich (IZ) [25]. The LIF neuron model can appropriately mimic the biological behavior of neurons with less computational complexity [26-31]. It is expressed as:

$$\frac{dv(t)}{dt}=\frac{1}{\tau_{mem}}\left(v_{reset}-v(t)+I(t)\right) \quad (1)$$

$$\frac{dI(t)}{dt}=-\frac{1}{\tau_{syn}}\left(I(t)-I_{in}\right) \quad (2)$$

In the above equations, the membrane and synaptic time constants are τ_{mem} and τ_{syn}, respectively. v is the neuron membrane voltage, and I is the sum of the currents entering the neuron. The constant parameter v_{reset} is the neuron membrane voltage at rest, and the parameter I_{in} is the input current, i.e. the intensity of each input pixel into the neuron. A threshold voltage *v_th* is defined for the neuron, which can vary and is generally around 1 mV. When the neuron membrane voltage exceeds the threshold, it fires a spike and returns to its resting state. Figure 4 shows the LIF neuron membrane voltage and the fired spikes for three different threshold voltages. What can be seen from the graphs is that the lower the spiking threshold of the neuron, the higher the firing rate. Therefore, the spikes are fired at shorter time intervals.

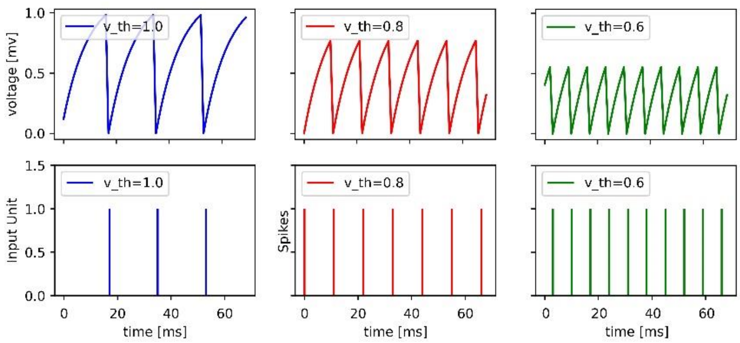

Figure 4. LIF neuron membrane voltage (top row) and its corresponding spike train (bottom row) under changing the threshold voltage v_th: 1.0mV (left side), 0.8 mV (middle), 0.6 mV (right side). The input pixel value to this neuron is 1.1795, and the simulation time is assumed to be 70 ms.

3.3. DWT pre-processing unit

The superiority or exceptional performance of wavelet transform theory is the ability to perform simultaneous time-frequency localization of signals. Due to this property, a specific and fast pattern recognition routine can be developed in the proposed spiking network. Instead of transmitting signals into the time-frequency domain directly, the wavelet transform expresses them in the time-scale domain, where each scale represents a specific frequency range. Therefore, the wavelet transform can separate the high- and low-frequency components accurately. References [2-6] [32-36] are recommended for studying the wavelet transform and its applications.

Wavelet analysis is based on small signals, called wavelets, all of which are obtained from the expansion and transition of a basic wavelet function, called the mother wavelet. The wavelet transform with discrete wavelets, i.e. DWT, on a two-dimensional function $f(x,y)$ such as an image is described as follows:

$$W_{\varphi}(j_0,m,n)=\frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\varphi_{j_0,m,n}(x,y) \quad (3)$$

$$W_{\psi}(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \psi^{i}_{j,m,n}(x,y) \tag{4}$$

The obtained coefficients W_{φ} and W_{ψ} are approximation and detail coefficients. $\varphi(x,y)$ and $\psi(x,y)$ are scale and wavelet functions. j is the scale parameter, and m and n are shift parameters. The wavelet coefficients are defined in three directions and follow the horizontal, vertical, and diagonal changes in the desired function.

The multilevel decomposition means that it starts with the coarsest scale and proceeds to the finer scales [37]. The image is divided into four regions or subbands, including one approximation and three details in each level. In the next level, the approximation coefficients are then decomposed again in the same way. The process continues until the desired level is reached. The approximation and detail coefficients are coarse and fine coefficients that show the obtained key features.

3.4 Input encoding

As mentioned earlier, encoding the numerical input data into spiking input data is required in the first stage of modeling a spiking network. The encoding process maps the vector level to the spike level. The rate and time coding schemes are the most important among the various schemes that have been used in many studies [38, 39]. We use a time encoding method called constant-current-LIF encoding [40] in the proposed spiking network. In this coding method, each pixel's intensity of the image is input as a constant current to a corresponding LIF neuron. Each neuron fires at most once and that time is the first moment its membrane voltage rises above the threshold value. The firing moment of this single spike is in inverse proportion to the intensity of the corresponding pixel. Finally, after running the encoding scheme, we have multiple feature maps distributed during the encoding interval for an initial feature map. The input dataset is the MNIST dataset, from which we use the original images and their inverses to evaluate the system. Figure 5 shows the spiking activity of a sample of these images and its inverse after applying the constant-current-LIF coding.

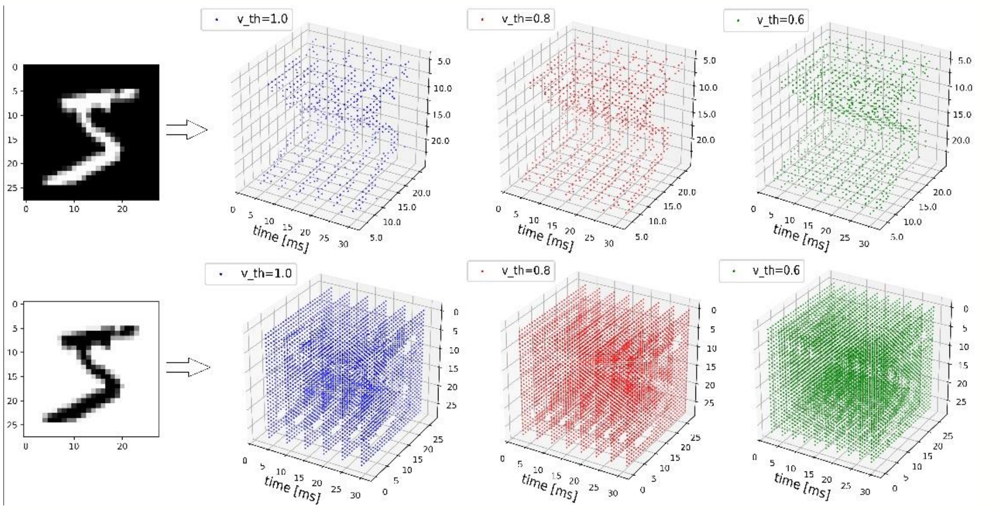

Figure 5. Constant-current-LIF encoding of digit 5 input image (top row) and its inverse (bottom row) by changing the threshold voltage v_th: 1.0 mV (left side), 0.8 mV (middle), 0.6 mV (right side). The simulation time is considered 32 ms.

The well-known MNIST dataset of handwritten digits [41-43] with 28 x 28 images consists of two sub-sets, a trainset with 60,000 sample images for learning the network and a testset with 10,000 sample images for its evaluation. In Figure 5, the spiking activity of the digit 5 image and its inverse are plotted for three different threshold voltages of the LIF neuron. These grayscale images are first transformed into a tensor and then normalized. They are then converted into a feature map by applying a DWT. Finally, after entering the coding block, they change into spike trains as shown in the figure. It is evident from the figure that the lower the neuron threshold voltage, i.e. its spiking threshold, the more spikes it fires in a shorter time interval.

3.5. Learning procedure

Biological synapses are communication interfaces between pre- and post-synaptic neurons that transmit information through neurotransmitters or electrical signals. To modify the strength of the synapses or the weights of the neural network, a learning process is used, which can be a known supervised, unsupervised or reinforcement learning method. This proposed network uses a supervised gradient descent algorithm. The loss function is based on the negative log-likelihood, which gives us the predicted probabilities assigned to correct labels to see how good the prediction was. Then we minimize this loss function using the gradient descent method.

In the last layer of the proposed network, after selecting the neuron with the highest output voltage as the winner and comparing it with the correct label, the prediction error is calculated using the negative log-likelihood function. The gradient descent algorithm returns the error backward to the network to improve the weights. The training of the network is repeated in each epoch for all images in the MNIST trainset. Next, the trained network is evaluated on all MNIST testset images for different threshold voltages and different wavelets. To determine the efficiency of the method, the number of correct decisions made by the system is divided by the total number of test examples. In this way, performance accuracy is obtained.

4. Results and Discussion

In the proposed system architecture, the values of the receptive fields in the convolutional and pooling layers are practically selected based on the architectural properties of the network and the statistics of input images. The convolutional layers comprise 5 × 5 windows with a stride length of 1. These windows have initial random weights with a uniform distribution in the range of 0 to 1. Non-overlapping 2 × 2 windows with weights of 1 are used in the pooling layers. The output of the second pooling layer is drawn to 800 neurons and then to 500 neurons using the FC layer. Eventually, the readout layer writes it to 10 output neurons over time. We varied the LIF neurons voltage of the network from 1 mV to 0.4 mV in steps of 0.05 mV In the simulations, moreover, the duration of the simulations is 70 ms, which are repeated on 300 epochs.

We have seen that redundant features are removed from the input image in DWT pre-processing stage. Therefore, encoding the residual essential information of the image into spike trains and then implementing the LIF neurons behavior in the spiking network will be easier. Different wavelets were tested, and different threshold voltages were also tried per wavelet for the LIF neurons. Furthermore, all these steps were performed for both MNIST images and their inverses. Tables 2 and 3 show the system recognition accuracy for all these conditions. It can be recognized that the best overall accuracy is 99.25% when a Coiflet wavelet of scale 9 with a threshold voltage of 0.7 mV is chosen for the original MNIST image. Accuracies greater than 99% were also achieved in many cases. The results prove that DWT pre-processing method can extract important features from the input image with a very high capability. Therefore, it is concluded that the wavelet transform is a compatible platform with the spiking structure of the neural network.

According to the accuracies summarized in Tables 2 and 3, discrete Coiflet wavelets on a scale of 8 to 10 performed best in the system for both original MNIST images and their inverses. The Coiflets family includes compactly supported wavelets with the highest number of vanishing moments for both scale and wavelet functions. Among wavelets of the same family with different scales, larger kernels provide better frequency resolution but poorer time resolution. Lower frequencies (scales) have the lowest output accuracy. Since the wavelet function has the largest expansion in the smaller scale, it loses the low-frequency content of the signal and focuses only on the high-frequency content. To make a tradeoff between time and frequency resolution, the wavelets were selected with different members or kernel elements. The results in Tables 2 and 3 are in terms of the average recognition rate and provide a summary of the digit recognition results for each wavelet function.

The confusion matrix for the case where a Coif9 wavelet is used is presented in Figure 6 in order to evaluate the classification accuracy. In addition, to estimate the classifier output quality in the proposed network, recall and precision metrics were calculated. The recall and precision values are equal to 0.9923 and 0.9924, in turn. High scores for both the precision and recall indicate that the classifier provides accurate results and a majority of all positive results, respectively. In Table 4, a comparison is made between the test accuracy of some spiking networks and the test accuracy of the proposed architecture for the pattern recognition of MNIST digits. Each spiking network has its own preprocessing, encoding, and learning method. In addition, This architecture consists of the preprocessing block of the wavelet transform, time encoding, and gradient descent training process. In this case, our proposed network has achieved 99.25% accuracy, which is comparable with the highest accuracy achieved in other spiking networks of this type. In [22], a spiking DNN consisting of multiple convolutional and pooling layers is proposed. DoG filters extract contrasts in the first layer of the network, which are then converted into spike trains by a time encoding block.

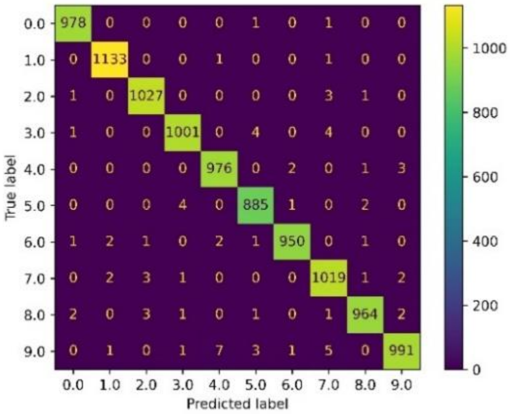

Figure 6. Confusion matrix for the Coif9 wavelet.

It has generally achieved 98.4% accuracy on MNIST digits by the STDP update rule. The proposed network in [44] does not use a preprocessing stage. It uses a rate coding instead of the time coding, which is used in our proposed method. The rate coding can achieve a slightly higher accuracy due to the generation of more spikes during the simulation time, which requires more computational effort. This architecture has achieved 99.28% accuracy utilizing a spike-based backpropagation training method preceded by an STDP-based pre-training scheme due to the speed, power and better generalization of this training method. The convolutional SNN presented in [45] does not also use a filtering stage. It has achieved an accuracy of 98.54% using rate coding and STDP-based probabilistic training method. It is also so computationally efficient due to the use of binary weights. Additionally, in [19, 20], a convolutional SNN was introduced using DoG filters and time encoding, from which an accuracy of 97.2% was achieved. The weights of three convolutional layers embedded in this architecture are updated through the STDP or R-STDP method. Concerning the article [46], an accuracy of 97.4% was achieved by means of an FC SNN with two hidden layers and the introduction of a delay-based backpropagation learning method which is based on the difference between the actual and target firing times. The input was not filtered, only time encoded.

Table 2. System recognition accuracy on images of MNIST digits.

Wavelet	*v_th*												
	1	**0.95**	**0.9**	**0.85**	**0.8**	**0.75**	**0.7**	**0.65**	**0.6**	**0.55**	**0.5**	**0.45**	**0.4**
Haar	98.46	98.22	98.44	98.53	98.45	98.48	98.53	98.39	98.30	98.53	98.38	98.20	98.38
Bior1.1	98.38	98.49	98.51	98.50	98.40	98.23	98.48	98.34	98.27	98.59	98.12	98.41	98.26
Bior5.5	98.34	98.47	98.45	98.67	98.58	98.60	98.38	98.36	98.60	98.62	98.61	98.57	98.55
Rbio1.1	98.97	98.70	98.77	98.65	98.54	98.66	98.58	98.66	98.63	98.64	98.71	98.64	98.62
Rbio5.5	98.82	98.77	98.68	98.85	98.77	98.77	98.67	98.70	98.69	98.70	98.81	98.75	98.58
Coif8	99.11	99.14	99.10	99.19	99.01	99.03	99.06	99.20	98.99	99.12	98.97	99.08	79.38
Coif9	**99.21**	99.14	**99.24**	99.15	99.14	99.14	**99.25**	99.02	99.06	99.04	99.05	99.11	99.01
Coif10	99.11	99.00	99.00	98.97	99.04	99.06	99.06	98.88	98.89	98.91	98.91	98.86	98.77
Dmey	99.03	98.98	99.00	99.03	98.93	98.96	98.89	98.84	98.90	98.79	98.94	98.97	99.00
Db1	98.58	98.83	98.81	98.77	98.51	98.59	98.73	98.59	99.00	98.56	98.70	98.69	98.58
Db20	98.93	98.90	99.07	98.98	99.03	98.84	98.90	98.91	98.91	98.95	98.83	98.82	98.77
Sym2	98.76	98.66	98.56	98.79	98.47	98.65	98.47	98.57	98.66	98.57	98.72	98.69	98.50
Sym20	99.06	99.17	98.95	99.03	98.55	99.07	99.02	99.07	98.97	99.06	99.00	99.23	98.91

Table 3. System recognition accuracy on inverse images of MNIST digits.

Wavelet	*v_th*												
	1	**0.95**	**0.9**	**0.85**	**0.8**	**0.75**	**0.7**	**0.65**	**0.6**	**0.55**	**0.5**	**0.45**	**0.4**
Haar	98.61	98.66	98.63	98.40	98.31	98.27	98.68	98.36	97.71	98.17	98.61	98.52	98.22
Bior1.1	98.34	98.52	98.70	98.40	98.65	98.08	98.27	98.23	98.53	98.35	98.71	98.11	98.03
Bior5.5	98.89	99.01	98.88	98.84	98.74	98.67	98.54	98.39	98.32	98.19	98.73	98.71	98.87
Rbio1.1	98.87	98.84	98.86	98.80	98.68	98.80	98.58	98.88	98.38	98.67	98.61	98.79	98.67
Rbio5.5	98.97	98.98	98.91	98.83	98.52	98.61	98.99	98.61	98.79	98.98	98.74	98.81	98.77
Coif8	99.16	**99.22**	99.16	99.18	99.07	99.19	98.97	98.90	9.80	98.95	99.05	99.09	98.84
Coif9	**99.22**	**99.23**	99.04	99.18	99.13	99.11	99.01	98.92	98.84	99.01	99.09	98.64	9.80
Coif10	99.04	99.02	99.09	98.93	98.97	98.83	98.97	98.75	98.92	98.93	98.90	98.90	98.82
Dmey	99.09	99.11	99.07	99.01	98.98	98.90	98.51	98.61	98.70	98.73	98.92	9.80	98.75
Db1	98.73	98.94	98.88	98.85	98.40	98.83	98.54	98.35	98.63	98.59	98.75	98.66	98.59
Db20	99.02	99.06	98.84	98.81	98.89	98.98	98.72	99.00	98.77	98.24	98.87	98.75	98.89
Sym2	98.86	98.80	98.78	98.64	98.24	98.59	98.57	98.26	98.14	98.62	98.69	98.65	98.51
Sym20	99.16	99.14	99.08	99.11	98.69	99.05	98.94	98.74	98.99	9.80	98.96	99.02	98.88

The recognition of MNIST handwritten digits is discussed in the spiking architecture presented in this paper. Thus for MNIST digits 0 to 9, there are 10 neurons in the output layer. After training this network, the neuron with the highest membrane voltage is known as the winner among the 10 output neurons. In Figure 7, the membrane voltage of all 10 output neurons is plotted in relation to each other for four different input MNIST digits, with each neuron having a specific color. On the left side of each row of the figure, the network has not yet been trained. On the right side, the trained network has been able to learn the image patterns properly and recognize the label corresponding to the input image in each state correctly. The handwritten digits dataset has a huge variety since each person's writing style is unique. No accepted mathematical model has yet been presented to follow the patterns and changes in these images. Moreover, there may be some correlation and similarity between some digits. For instance, between digits 3 and 5 or 0 and 8. Therefore, there is a possibility that the resulting membrane voltage of neurons corresponding to the digits are close to each other. However, based on the graphs shown in Figure 7, it is clear that even in such cases, the output voltage of the winner neuron is higher than that of the others by a recognizable distance so that according to the applied approaches, the winner neuron has been determined accurately in this architecture.

Table 4. Comparison of MNIST digit recognition performance on some spiking networks.

Ref.	Architecture	Pre-processing	Coding	Learning-rule	Performance
(Kheradpisheh *et al*., 2018) [22]	Convolutional	DoG filtering	Time	STDP	98.4%
(Lee *et al*., 2018) [44]	Convolutional	---	Rate	STDP & Backpropagation	99.28%
(Srinivasan & Roy, 2019) [45]	Convolutional	---	Rate	Hybrid-STDP & Backpropagation	98.54%
(Mozafari, Ganjtabesh, Nowzari-Dalini, Thorpe *et al*., 2019) [19]	Convolutional	DoG filtering	Time	STDP & R-STDP	97.2%
(Mozafari, Ganjtabesh, Nowzari-Dalini & Masquelier, 2019) [20]	Convolutional	DoG filtering	Time	STDP & R-STDP	96.9%
(Kheradpisheh & Masquelier, 2020) [46]	Multilayer FC	---	Time	Backpropagation	97.4%
This paper	Convolutional	DWT	Time	Backpropagation	**99.25%**

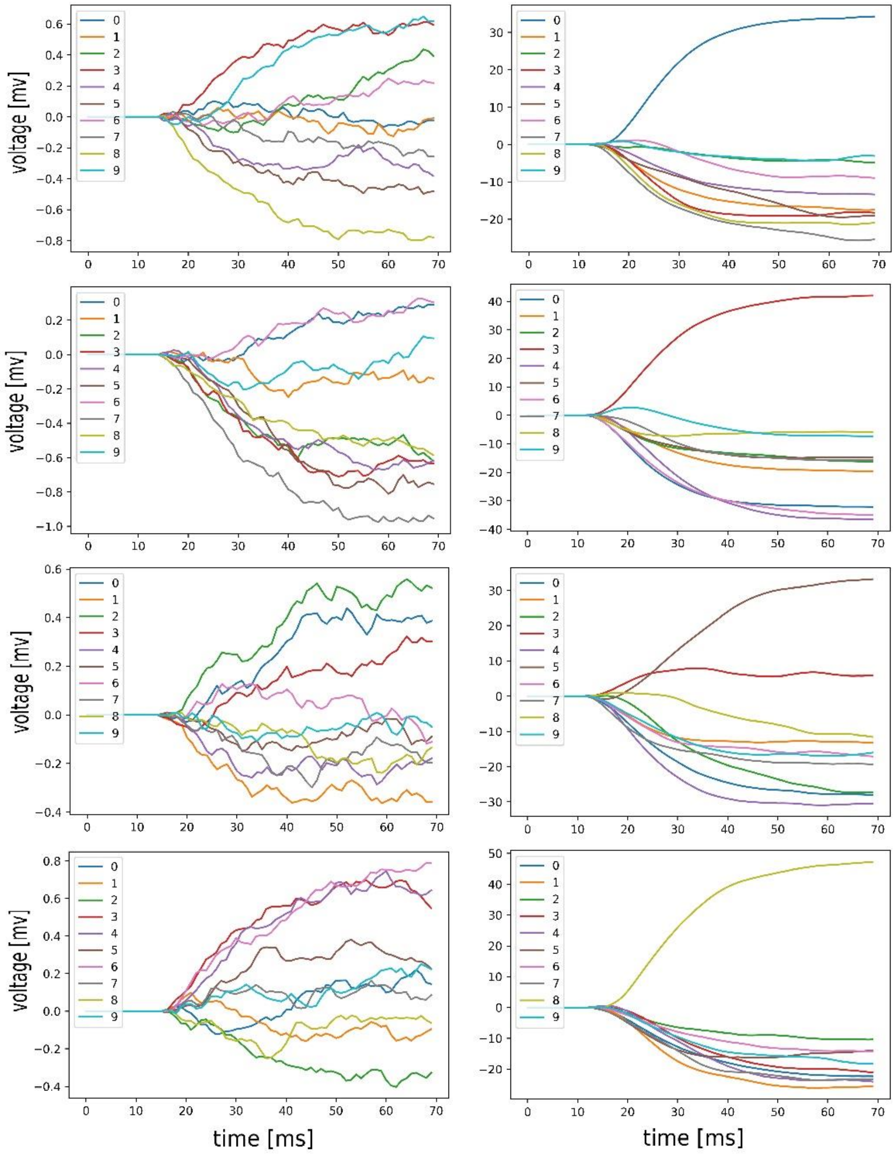

Figure 7. Membrane voltage of 10 output neurons for a sample input image of digit: 0 (first row), 3 (second row), 5 (third row), and 8 (last row), before training (left-hand side), after training (right-hand side).

5. Conclusion

In this work, wavelet analysis was studied as a mathematical tool that could be used in the pre-processing unit before information coding. Wavelet analysis consists of functions such as small waves that are limited in the frequency domain like those in Fourier analysis. However, unlike Fourier analysis, the waves are limited in the time domain, too. Using the discrete type of wavelet algorithms, i.e. DWT, the high- and low-frequency contents of the signals corresponding to fast and slow transitions can be separated easily. In this way, the approximation and precise details of signals can be shown in the time domain.

In this paper, a spiking convolutional architecture was proposed. It is evident that extracting critical features through some kind of input data transformation and then encoding the transformed spectrum into spike trains are primary and crucial steps in a spiking network. To extract the basic features of the input data in the spiking system, the feasibility of using the wavelet transform as a preprocessing step of the input image was evaluated.

The proposed wavelet transform method in this study was tested with the aim of pattern recognition in the MNIST handwritten digit dataset. Most of the discrete wavelets, each with multiple core elements, were used in the evaluations experimentally. The network accuracy was obtained and listed for each wavelet scale. The Coiflet basic or mother wavelet and its larger kernels ensure the best system performance of all filters. The output accuracy is lower on the smaller scales of the filters since the wavelet function is stretched to the maximum, and the low-frequency information of the signal is lost, thus only high values are displayed. On large scales, where the wavelet is less stretched, the frequency resolution is higher and functions relatively well. The highest accuracies of the network output using the Coiflet wavelet function at scale 9 with MNIST input images and their inverses were obtained at 99.25% and 99.23%, respectively. The obtained results confirm that this type of preprocessing is a reliable analysis that provides an adequate representation of the original data before feeding it into the network. This capability can be used to extract high-level features from the input data.

References

[1] D. Gabor, "Theory of communication. Part 1: The analysis of information," *Journal of the Institution of Electrical Engineers*, vol. 93, no. 26, pp. 429-441, 1946.

[2] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 11, no. 7, pp. 674-693, 1989.

[3] G. Strang and T. Nguyen, *Wavelet and Filter banks*. Wellesley-Cambridge Press, 1997.

[4] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on pure and applied mathematics*, vol. 41, no. 7, pp. 909-996, 1988.

[5] C. K. Chui, *An introduction to wavelets* (vol. 1). Academic Press, 1992.

[6] M. Farge, "Wavelet transforms and their applications to turbulence," *Annual review of fluid mechanics*, vol. 24, no. 1, pp. 395-458, 1992.

[7] V. Sze, Y. H. Chen, T. J. Yang, and J. S. Emer, "Efficient Processing of Deep Neural Networks: A Tutorial and Survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.

[8] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, 2017.

[9] T. Bouwmans, S. Javed, M. Sultana, and S. K. Jung, "Deep neural network concepts for background subtraction: A systematic review and comparative evaluation," *Neural Networks*, vol. 117, pp. 8–66, 2019.

[10] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, and V. K. Asari, "A state-of-the-art survey on deep learning theory and architectures," *Electronics*, vol. 8, no. 3, p. 292, 2019.

[11] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier and A. Maida, "Deep learning in spiking neural networks," *Neural Networks*, vol. 111, pp. 47–63, 2019.

[12] M. Pfeiffer and T. Pfeil, "Deep Learning With Spiking Neurons: Opportunities and Challenges," *Frontiers in Neuroscience*, vol. 12, p. 774, 2018.

[13] T. H. Rafi, "A Brief Review on Spiking Neural Network - a Biological Inspiration," Preprints 2021, 2021040202 (doi: 10.20944/preprints202104.0202.v1).

[14] S. A. Mohamed, M. Othman and M. Hafizul Afifi, "A review on data clustering using spiking neural network (SNN) models," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 15, no. 3, pp. 1392–1400, September 2019.

[15] A. Taherkhani, A. Belatreche, Y. Li, G. Cosma, L. P. Maguire, and T. M. McGinnity, "A review of learning in biologically plausible spiking neural networks," *Neural Networks*, vol. 122, pp. 253–272, February 2020.

[16] A. E. Hassanien, A. Abraham, and C. Grosan, "Spiking neural network and wavelets for hiding iris

data in digital images," *Soft Computing*, Vol. 13, No. 4, pp. 401-416, February 2009.

[17] Z. Zhang, Q. Wu, Z. Zhuo, X. Wang, and L. Huang, "Wavelet transform and texture recognition based on spiking neural network for visual images," *Neurocomputing*, vol. 151, pp. 985-995, 2015.

[18] M. Mozafari, S. R. Kheradpisheh, T. Masquelier, A. Nowzari-Dalini, and M. Ganjtabesh, "First-spike-based visual categorization using reward-modulated STDP," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 12, pp. 6178-6190, May 2018.

[19] M. Mozafari, M. Ganjtabesh, A. Nowzari-Dalini, S. J. Thorpe, and T. Masquelier, "Bio-inspired digit recognition using reward-modulated spike-timing-dependent plasticity in deep convolutional networks," *Pattern Recognition*, vol. 94, pp. 87-95, October 2019.

[20] M. Mozafari, M. Ganjtabesh, A. Nowzari-Dalini, and T. Masquelier, "SpykeTorch: Efficient Simulation of Convolutional Spiking Neural Networks With at Most One Spike per Neuron," *Frontiers in Neuroscience*, vol. 13, pp. 1–12, July 2019.

[21] S. R. Kheradpisheh, M. Ganjtabesh and T. Masquelier, "Bio-inspired unsupervised learning of visual features leads to robust invariant object recognition," *Neurocomputing*, vol. 205, pp. 382–392, September 2016.

[22] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, "STDP-based spiking deep convolutional neural networks for object recognition," *Neural Networks*, Vol. 99, pp. 56–67, March 2018.

[23] R. Vaila, J. Chiasson, and V. Saxena, "A Deep Unsupervised Feature Learning Spiking Neural Network With Binarized Classification Layers for the EMNIST Classification," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 6, no. 1, pp. 124–135, 2022.

[24] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *The Journal of Physiology*, vol. 117, no. 4, p. 500, 1952.

[25] E. M. Izhikevich, "Which Model to Use for Cortical Spiking Neurons?," *IEEE Transactions on Neural Networks*, vol. 15, no. 5, pp. 1063–1070. September 2004.

[26] J. P. Keener, F. C. Hoppensteadt, and J. Rinzel, "Integrate-and-Fire Models of Nerve Membrane Response to Oscillatory Input," *SIAM Journal on Applied Mathematics*, vol. 41, no. 3, pp. 503–517, 1981.

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

[28] A. N. Burkitt, "A review of the integrate-and-fire neuron model: II. Inhomogeneous synaptic input and network properties," *Biological cybernetics*, vol. 95, no. 2, pp. 97-112, 2006.

[29] R. A. Vazquez and A. Cachon, "Integrate and Fire neurons and their application in pattern recognition," in *2010 7th International Conference on Electrical Engineering Computing Science and Automatic Control*, CCE 2010, 2010, pp. 424–428.

[30] L. Long and G. Fang, "A review of biologically plausible neuron models for spiking neural networks," *AIAA Infotech @ Aerospace 2010*, p. 3540.

[31] E. Hunsberger and C. Eliasmith, "Spiking deep networks with LIF neurons," *arXiv preprint arXiv:1510.08829*, 2015.

[32] G. Beylkin, "On the representation of operators in bases of compactly supported wavelets," *SIAM Journal on Numerical Analysis*, vol. 6, no. 6, pp. 1716-1740, December 1992.

[33] I. Daubechies, *Ten lectures on wavelets*. Society for industrial and applied mathematics, 1992.

[34] S. G. Mallat, "Multifrequency channel decompositions of images and wavelet models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 2091-2110, 1989.

[35] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 14, no. 7, pp. 710-732, July 1992.

[36] M. Sifuzzaman, M. R. Islam, and M. Z. Ali, "Application of wavelet transform and its advantages compared to Fourier transform," *Journal of Physical Sciences*, vol. 13, no. 1, pp. 121-134, 2009.

[37] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Transactions on image processing*, vol. 1, no. 2, pp. 205-220, April 1992.

[38] R. Van Rullen and S. J. Thorpe, "Rate Coding Versus Temporal Order Coding: What the Retinal Ganglion Cells Tell the Visual Cortex," *Neural Computation*, vol. 13, no. 6, pp. 1255–1283, 2001.

[39] N. Abderrahmane, E. Lemaire, and B. Miramond, "Design Space Exploration of Hardware Spiking Neurons for Embedded Artificial Intelligence," *Neural Networks*, vol. 121, pp. 366-386, January 2020.

[40] C. Pehle and J. E. Pedersen, "Norse: A library to do deep learning with spiking neural networks," GitHub, 2019. Available: https://github.com/norse.

[41] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[42] L. Deng, "The mnist database of handwritten digit images for machine learning research [best of the web]," *IEEE signal processing magazine*, vol. 29, no.

6, pp. 141-142, November 2012.

[43] M. R. Shamsuddin, S. Abdul-Rahman, and A. Mohamed, "Exploratory analysis of MNIST handwritten digit for machine learning modelling," in *International Conference on Soft Computing in Data Science*, Springer, Singapore, Aug 2018, pp. 134-145.

[44] C. Lee, P. Panda, G. Srinivasan, and K. Roy, "Training Deep Spiking Convolutional Neural Networks With STDP-based Unsupervised Pre-training Followed by Supervised Fine-Tuning," *Frontiers in Neuroscience*, vol. 12, pp. 1–13, August 2018.

[45] G. Srinivasan and K. Roy, "ReStoCNet: Residual Stochastic Binary Convolutional Spiking Neural Network for Memory-Efficient Neuromorphic Computing," *Frontiers in Neuroscience*, vol. 13, pp. 1–18, March 2019.

[46] S. R. Kheradpisheh and T. Masquelier, "Temporal backpropagation for spiking neural networks with one spike per neuron," *International Journal of Neural Systems*, vol. 30, no. 6, p. 2050027, 2020.

بازشناسی رقم در شبکه‌های عصبی اسپایکی با استفاده از تبدیل موجک

هدیه آقابرار، کوروش کیانی* و پرویز کشاورزی

دانشکده مهندسی برق و کامپیوتر، دانشگاه سمنان، سمنان، ایران.

ارسال ۲۰۲۳/۰۱/۱۵؛ بازنگری ۲۰۲۳/۰۴/۰۴؛ پذیرش ۲۰۲۳/۰۵/۰۱

چکیده:

امروزه با توجه به پیشرفت سریع بازشناسی الگو، می‌توان از ایده‌های جدیدی مانند ریاضیات نظری برای بهبود بازدهی این موضوعات بهره برد. در این مقاله، تبدیل موجک گسسته به عنوان یک چارچوب ریاضی برای اجرای بازشناسی ارقام دست‌نویس در شبکه‌های عصبی اسپایکی مورد استفاده قرار می‌گیرد. انگیزه بکارگیری این روش این است که تبدیل موجک می‌تواند اطلاعات اسپایک و نویز را به زیر باندهای فرکانسی جداگانه تقسیم نماید و همچنین اطلاعات زمانی را ذخیره کند. نتایج شبیه‌سازی‌ها نشان می‌دهد که تبدیل موجک گسسته یک انتخاب مؤثر و شایسته است و شبکه با استفاده از آن، به بازدهی قابل مقایسه با شبکه‌های اسپایکی قبلی خواهد رسید. ابتدا تبدیل موجک گسسته به تصاویر MNIST در ورودی شبکه اعمال می‌شود. متعاقباً یک نوع کدگذاری زمانی به نام کدگذاری جریان ثابت-ادغام نشتی و آتش (LIF) روی داده‌های تبدیل یافته اجرا می‌شود. سپس تصاویر کدگذاری شده به شبکه اسپایکی کانولوشنی چند لایه وارد می‌شوند. در این معماری، موجک‌های مختلف بررسی شده‌اند و بالاترین دقت طبقه-بندی ۹۹,۲۵ درصد به دست آمده است.

کلمات کلیدی: تبدیل موجک، بازشناسی رقم، شبکه‌های عصبی اسپایکی کانولوشنی، کدگذاری جریان ثابت-LIF.