

# Handwritten Digit Recognition based on Output-Independent Multi-Layer Perceptrons

Ismail M. Keshta

Department of Computer Engineering  
King Fahd University of Petroleum and Minerals  
Dhahran, Saudi Arabia

**Abstract**—With handwritten digit recognition being an established and significant problem that is facing computer vision and pattern recognition, there has been a great deal of research work that has been undertaken in this area. It is not a trivial task because of the big variation that exists in the writing styles that have been found in the available data. Therefore both, the features and the classifier need to be efficient. The core contribution of this research is the development of a new classification technique that is based on the MLP, which can be identified in handwritten documents as the binary digits ‘0’ and ‘1’. This technique maps the different sets of various input data onto the MLP output neurons. An experimental evaluation of the technique’s performance is provided. This evaluation is based on the well-known ‘Pen-Based Recognition of Handwritten Digits’ dataset, which is comprised of a total of 250 handwriting samples that are taken from 44 writers. The results obtained are very promising for such an approach in accurate handwriting recognition.

**Keywords**—Handwritten digit recognition; Pattern classification; Neural network mode; Two-class classification; Accuracy; Binary data

## I. INTRODUCTION

The method of handwritten digit recognition both recognizes and classifies handwritten digits from (0-9) without there being any human interaction at all. It is a computer’s ability to first receive and then interpret this handwritten input intelligently from a number of sources, including photographs, paper documents, touch screens and a number of other similar devices [1]. In fact, recognizing the handwritten digits of people has been an important field of progressive research for the last few decades and studied intensely as a challenging problem [1]-[3]. Researchers have achieved a number of results by using different algorithms, such as K-Nearest-Neighbors (KNNs), Neural Networks (NNs), etc. [2], [3].

Handwritten digit recognition remains a vital area because of its huge practical applications, as well as the important financial implications [1] [2]. It is promising in a wide range of application domains, including online handwriting recognition on computer tablets; zip code recognition to help sort posted mail, as well as the verification of signatures on cheques in order to thwart any attempts at bank fraud, etc. Handwritten digit recognition is also widely used in a number of academic institutions to process their examination papers [1], [3].

The handwriting recognition process faces certain challenges [4]-[7]. The biggest one is that the handwritten

images’ various dimensions have to be normalized and/or processed in order for them to fall within the system’s boundary specific requirements. Thus, the challenges of handwritten digit recognition do not just result from the many different ways in which a single digit can be written, but also arise from the various different requirements that are imposed by the applications that are specifically used [1], [2]. In addition, there are the varying degrees of thickness of people’s handwriting and the writing’s different positions when it comes to the sample’s margins [8]-[10]. Moreover, Akhtar *et al.* [1] indicated that as people’s various writing styles will depend on their age, qualifications, mode, background, etc. handwritten digit recognition is a relatively complex research task and the format of handwritten digits will have a big impact on classification and/or identification, as the various subjects will all use different styles of writing. In fact, even digits that have been written by the same person at different times were also found to vary [11], [12]. Therefore, it is almost impossible to develop a generic recognizer that is able to recognize an infinite number of writers’ handwritten digits.

As part of this research, we investigated a number of different handwritten digit recognition schemes and presented various existing methods after reviewing a number of recent researches related to this field. A classification technique that is based on MLP was also proposed. This was done in order to recognize both the binary digits of ‘0’ and ‘1’ in a series of handwritten documents. This technique was implemented in Java and then tested on the Pen-Based Recognition of Handwritten Digits dataset.

As part of the proposed approach, the number of hidden neurons depends on the number of input features (including the bias), which is in contrast to other approaches wherein the number of hidden layer units depends on the number of output neurons. As a result, the proposed approach provides for a more concise network for the recognition task. Furthermore, the ultimate goal is binary digit classification, which finds applications in digital system design and human subject identification.

The rest of this paper is organized as follows. Section II provides a literature survey of recent approaches proposed for handwritten digit recognition. Section III elaborates upon the proposed MLP-based classification technique. Section IV discusses the simulation results obtained. Finally, a conclusion and future directions for research are all summarized in Section V.

## II. HAND WRITING RECOGNITION APPROACHES

This section describes some of the existing approaches for handwritten digit recognition.

### A. Clustering approaches

The aim of the research done by the authors in [12] is to use the genetic algorithm for clustering digit for handwriting digit recognition. It uses the famous genetic algorithm clustering method based on the phenomenon of natural selection in genetic evolution, i.e., mating and mutation. Starting from an initial population of arbitrary samples of handwriting, a group of new and elite individuals are generated through random selection. There are three main operations or phases in the genetic algorithm, namely, selection, cross over and mutation. The proposed hand digit recognition includes three aspects: digital image pre-processing, feature extraction, and identification.

The authors found that the genetic algorithm has a higher clustering accuracy than the Backpropagation neural network; in addition, the experiment carried out shows that genetic algebra requires less training time than the BP NN. In [13], the scheme proposed uses the following procedure for conducting the clustering task. The procedure involves three steps: 1) Image pre-processing, 2) Feature extraction, and 3) Clustering of digits. In the image pre-processing phase, the following sub-tasks are needed, namely, binarization, smoothing and de-noising, image normalization and refinement.

In the binarization stage, the image is transformed into a grey scale version that contains only two kinds of values, 0 or 1. The smoothing and de-noising task is needed to fine-tune the binary output for noise removal/smoothing. The third task, which is image normalization, is needed because text sizes and their positions differ from person to person, thus the characters need to be normalized. The final sub task is refinement, which is mainly needed for reduction of the original binary matrix into a value that can be easily stored. This is done by obtaining an approximate graph formed by a simple curve with the original. Secondly, feature extraction; the method used in this paper is the principal component analysis (PCA), which is statistical feature extraction that uses the full data in the second order statistical information for data feature extraction and reduction of dimensionality. The authors use the k-means clustering algorithm to partition N observations into K clusters, in which each observation belongs to the cluster with the nearest mean.

### B. Feature extraction methods for Handwritten Digit Recognition

The authors in [14] propose an algorithm that uses two dimensional hidden Markov models for a handwritten digit recognition algorithm. Their main idea is based on the combination of tropical geometry and algebraic statistics in order to determine the Markov model parameters. The main task is to compute the distance between the unclassified images and prototypes in a certain way. For that purpose, they used the k-nearest neighbors technique.

The authors in [15] proposed a method to extract the features of one-number Persian images and subsequently used

a neural network with three layers for classification. Their method has the capability of extraction of the ideal features for one number in an image even if there is a rotation, changing in the size and noise. The proposed method has four main components, namely, frame finding, classification, feature extraction and finally recognition using neural network. The first step, which is frame finding, narrows the digits and extracts the features of that narrowed digit. Classification is used to classify the narrowed digits into two categories/classes. The two categories are digits 1, 2, 3, 4, 6 and 9 that have a 1-like base and the other digits that are 0, 5, 7 and 8. Feature extraction is used to extract the features based on direction and accumulation techniques. Finally, the recognition is done using the multi-layer perceptron neural network with a feed-forward algorithm used for the final recognition of the number.

### C. Neural Network for Digit Recognition

The authors in [16] present an intensive and complete representation for the two main types of neural networks, which are the backpropagation networks and radial basis function networks. They also attempt to compare the capabilities of these two types of neural networks based on a digit image recognition problem. The authors use 10 digit images, from "0" to "9". Each image consists of  $8 \times 7 = 56$  pixels. The authors highlight the architecture for the neural networks based on the digit image recognition problem. For backpropagation networks they have 56 inputs, 10 outputs and no hidden layer. Their architecture for the RBF neural network has 56 inputs, 10 outputs and an additional layer that consist of 10 units for each data image; this is because the 10 input image data can be classified into 10 clusters.

In [11] and [15]-[17], all schemes use the MLP backpropagation method in classification of the handwritten digits. In the propagation phase, data propagation involves the following steps: 1) forward propagation of a training pattern's input through the neural network in order to generate the propagation's output activations, and 2) backward propagation of the propagation's output activations through the neural network using the training pattern's target in order to generate the deltas of all output and hidden neurons.

The authors of [18] aim to analyze the effect of two backpropagation (BP) algorithms: simple BP and BP with momentum, on a handwritten digit recognition task. They analyze the relationship between the learning rate and the accuracy of the handwritten digit recognition task using two backpropagation schemes. It is very important to note that the choice of the learning rate is critical for the design of a BP neural network. The main distinction between the simple backpropagation and backpropagation with momentum is in the way the weights are updated; the latter (BP with momentum) uses an additional term, which is the momentum. They provided two results. The first one was by using the simple BP algorithm and the other using BP with momentum for different learning rates and different number of hidden neurons: 150 iterations are used in both BPs. The results show that the accuracy of the network increases as the learning rate increases from 0 and approaches 1, but the accuracy decreases as the learning rate decreases for any number of hidden neurons in the hidden layer.

### III. THE PROPOSED MLP-BASED DIGIT RECOGNITION TECHNIQUE

After describing some of the existing approaches for handwritten digits recognition in Section II, this section describes the proposed classification technique based on the Multilayer Perceptron (MLP) algorithm, for identifying binary digits '0' and '1' in handwritten documents. Also, under the same section, the implemented algorithm is pointed out. In addition, simulation setups for the proposed scheme that are used for evaluating the performance of the implementation are described in this section.

#### A. Dataset

In this research work, the "Pen-Based Recognition of Handwritten Digits" dataset [19] is used. It is a digit database that was prepared from handwritten numeric digits. It is a collection of 250 samples from 44 different writers. The writers are advised to write 250 digits in a casual order inside boxes of 500 x 500 tablet pixel resolution. Each screen contains five boxes, with the digit to be written displayed above; subjects are told to write only inside the boxes. 10992 instances exist in this dataset with 16 input features [19], [20]. The calculation for these features was based on basis of (X, Y) coordinates of the pen at a particular instance. The dataset contains 10 classes. It is important to point out that all input attributes are integers in the range (0-100). The last attribute in this dataset is the class label that is an integer in the range (0-9) [21]. It is important to point out that the dataset was filtered based on the class label in order to include only digits 0 and 1 and exclude the other digits from (2-9).

From the literature, it can be concluded that the authors of [12] used genetic algorithm for clustering digit samples in order to implement handwriting digit recognition. The genetic algorithm has a faster convergence rate than the back propagation neural network methods; it involves complexities in its implementation and includes many randomization processes. Another method is the use of k-means clustering algorithm [13], which has the problem of deciding on the members of the initial cluster. The proposed approach focuses on the use of a classification technique based on the Multilayer Perceptron (MLP) for identifying binary digits '0' and '1' in handwritten documents, with neural structure being independent of the size of the output layer.

It is important to point out that the training algorithm for the MLP neural network employs a technique which makes the number of hidden units in the hidden layer equal to the number of the input neurons which includes nodes for the features and bias. In other words, the difference between the proposed approach and approaches proposed by Yu *et al.* in [13] and others for handwriting digit recognition is that in the approach, the number of the hidden units depends on the number of the elements in the input vector regardless of the number of the outputs, whereas the other techniques rely on the number of the outputs. For example, if the number of features is 4 then the number of hidden units will be 10 based on the previous approaches because of having ten digits (i.e., 0-9), while in the proposed technique the number of hidden units will be five corresponding to four features and one bias regardless of the number elements in the output vector. Therefore, for

identifying binary digits '0' and '1' in handwritings, the previous approaches require only two hidden units in the hidden layer regardless of how many features are used, while the proposed approach requires the number of the hidden units to be the same as the number of the features plus one for bias.

```
1) /**Initializing steps**/
2) Initialize all weights to small random values and Take the
   feature vector and bias as Input vector.
3) /**propagate the input features and bias of the training
   pattern k to determine the MLP's output [steps 4-14]**/
4) for i ← 0 to (number of features + 1 )
5)   hiddenU[i] ← 0
6)   for j ← 0 to (number of features + 1 )
7)     hiddenU[i] ← hiddenU[i]+ trainPattern[k][ j]*
      IH_Weights[j][i]
8)   end for
9)   hiddenU[i] ← tanh (hiddenU[i])
10) end for
11) predOut ← 0
12) for i ← 0 to (number of features + 1 )
13)   predOut ← predOut + hiddenU[i]+ HO_Weights[i]
14) end for
15) /**calculate the error for the pattern k, step 16**/
16) Error_inPatternK= predOut - actual [k]
17) /**update weights hidden-output, steps (18-21) **/
18) for i ← 0 to (number of features + 1 )
19)   delta = Error_inPatternK * hiddenU[i]
20)   HO_Weights[i] ← HO_Weights [i]- delta;
21) end for
22) /**update weights input-hidden, steps (23-31)**/
23) for i ← 0 to (number of features + 1 )
24)   for j ← 0 to (number of features + 1 )
25)     x ← 1-(hiddenU[i] * (hiddenU[i]))
26)     x ← x* HO_Weights [i] * Error_inPatternK
27)     x ← x* trainPattern[k][ j]
28)     change ← x
29)     IH_Weights[j][i]= IH_Weights[j][i]-change
30)   end for
31) end for
```

Fig. 1. Pseudo-code for the proposed binary digit recognizer.

However, for the proposed approach, the network size depends on the number of features; thus, this approach gives a more concise network when the recognition task is extended to 0-9.

#### B. The Implemented MLP algorithm

The main steps of the algorithm in the training phase for the MLP network are as follows: given an input pattern X from the dataset, this value is forward propagated to the output of the MLP network, and is compared with the desired output; the error signal between the output of the network and the desired

response is then back propagated to the network, and adjustments are made on the synaptic weights. The same procedure is repeated for the next input vector until all the training patterns are passed through the neural network [22]-[24]. The pseudo-code for the implemented training MLP algorithm is provided in Fig. 1. It illustrates the main steps needed for propagating the input features and bias of a training pattern  $k$  as shown in [step lines 4-14]. Line 16 calculates the error in current pattern  $k$ . For updating the weight, we have two stages, namely updating the weights hidden-output as in steps 18-21, and updating the weights in the input to hidden layer interconnections, as in steps 23-31.

### C. Experimental work

**2-Feature Experiment:** The first experiment was performed using only two features, namely, Feature 6 and Feature 14 for digits 0 and 1. That means, the number of inputs is two for the two features and one input for the bias. So, the number of inputs is three in total for each input pattern. Therefore, the size of the input training array is (number of input Patterns X 3). It is considered that only one hidden layer in both experiments. In the first experiment, the number of inputs is three (two inputs for Features + one for bias), i.e., three units in the hidden layer as shown in Fig. 2. For the input weight to hidden layer interconnection, it is a 2-D array of size equal to number of input patterns times 3. The output weights are presented as a 1-D array of size three. Finally, it is also assumed that they have only one unit at the output layer.

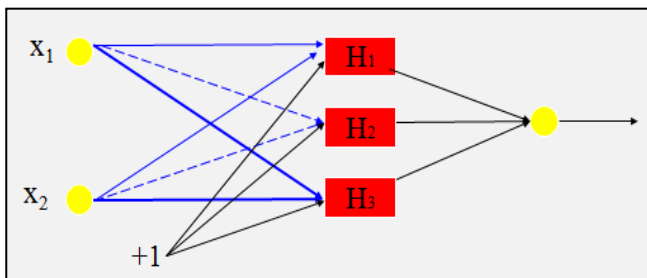


Fig. 2. MLP implemented structure for experiment 1.

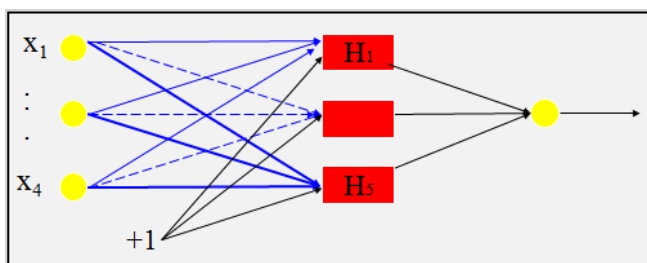


Fig. 3. MLP implemented structure for experiment 2.

**4-Feature Experiment:** For the second experiment, four features are used, namely, Feature 5, Feature 6, Feature 8 and Feature 14 for digits 0 and 1. Thus, the number of inputs is four

for each feature and one input for the bias. So, five inputs in total can be seen for each input pattern. Therefore, the size of the input training array is equal to number of input patterns times 5. Five units in the hidden layer are used as illustrated in Fig. 3. The size of the input weight to hidden layer array is equal to number of input patterns times 5. The size of the output weight array is five. Similar to the first experiment, only one unit is used at the output layer.

## IV. SUMMARY AND DISCUSSION OF RESULTS

Fig. 4, 5 and TABLE. I summarize the results obtained through simulation of both the setups. In TABLE. I, the error levels are shown, when applying the proposed approach on the Pen-Based Recognition of Handwritten Digits dataset. The salient aspects that require analysis are the numbers of correctly/incorrectly classified instances.

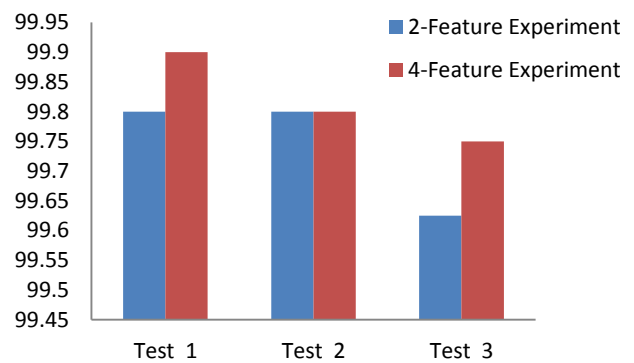


Fig. 4. Accuracy of the classifier for all the tested scenarios.

As shown on the obtained results, the implemented MLP performs better when four features are included instead of having just two features. In the case that the four features are included and the network is trained on the whole dataset and then tested again on the whole dataset, we got the highest identification rate of 99.9%. However, under the same test option, if two features are included, the percentage of correctly classified instances was around 99.8%.

In the second test option where the network is trained on half of the dataset and test on the other half, there is no clear difference between including four features or just having two, with the implementation yielding similar results for both cases. For the last test, test 3, including four features yields better performance than having just two features by merely 0.15%. It is important to highlight that the value of back propagation learning rate for all the tests was set as 0.7. Several other values for the learning rate were tried but because they yielded degraded performance; it is selected to be 0.7 as the optimal value for the conducted experiments.

TABLE. I. ACCURACY OF THE IMPLEMENTED MLP BASED ON THE FEATURE SIZES

Feature sizes	Experiment no.	Number of instances used for training	Number of instances used for testing	Correctly classified	Incorrectly classified	Accuracy%
With only two features F6 and F14	Experiment 1: Test 1	1000	1000 (same dataset used for training)	998	2	99.8
	Experiment 1: Test 2	500 (train in the first 500 instances)	500 (Test on the remaining ones)	499	1	99.8
	Experiment 1: Test 3	200 (train in the first 200 instances)	800 (Test on the remaining ones)	797	3	99.6
With four features F5,F6,F8 and F14	Experiment 2: Test 1	1000	1000 (same dataset used for training)	999	1	99.9
	Experiment 2: Test 2	500 (train in the first 500 instances)	500 (Test on the remaining ones)	499	1	99.8
	Experiment 2: Test 3	200 (train in the first 200 instances)	800 (Test on the remaining ones)	798	2	99.75

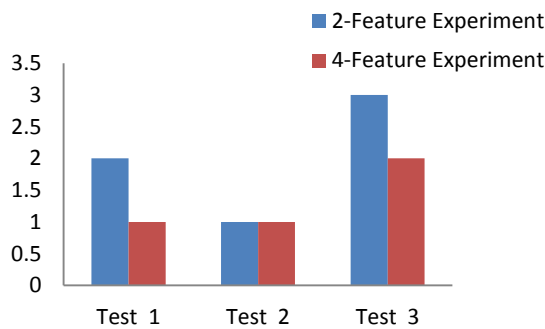


Fig. 5. Number of false classifications for the three tests and the two experiments.

### V. CONCLUSIONS & FUTURE WORK

In this paper, an MLP-based approach for identifying human handwriting samples is proposed. The proposed scheme provides an inner setup of the neural network different from existing research work done in the recent past.

The approach is output-independent, and provides for a hidden layer neuron structure, which is dependent only on the input layer. Two sets of experiments were conducted for testing the effectiveness of the proposed technique. The first approach accepts only two features as inputs and the second includes four features as inputs. In addition, the first sets of experiments were conducted using two features only and training and testing repeated on the same dataset. The second set of experiments bifurcated the data set into two sets of size 500 each. Training was done on one half, with testing done on the other. From the experiments conducted above, it can be concluded that the MLP back-propagation technique based on

output-independence is an excellent classifier for handwritten digit classification; furthermore, it is evident from the above results that selection of higher number of features gives more classification accuracy as opposed to using fewer features.

As part of future work, we plan to evaluate the performance of the proposed classification technique that is based on the MLP under other benchmarking datasets. In addition, we intend to study the effect of attribute selection and transfer function selection on the outcome of the classifier.

### REFERENCES

- [1] Akhtar, Muhammad Suhail, and Hammad A. Qureshi, "Handwritten digit recognition through wavelet decomposition and wavelet packet decomposition." Digital Information Management (ICDIM), 2013 Eighth International Conference on. IEEE, 2013.
- [2] Niu, Xiao-Xiao, and Ching Y. Suen. "A novel hybrid CNN-SVM classifier for recognizing handwritten digits." Pattern Recognition 45.4 (2012): 1318-1325.
- [3] Ali, Syed Salman, and Muhammad Usman Ghani. "Handwritten Digit Recognition using DCT and HMMs." Frontiers of Information Technology (FIT), 2014 12th International Conference on. IEEE, 2014.
- [4] Jackson, J.E. (1991). A User's Guide to Principal Components (Wiley).
- [5] Jolliffe, Ian. "Principal component analysis." New York: Springer-Verlag., 2002.
- [6] Mohebi, Ehsan, and Adil Bagirov. "A convolutional recursive modified Self Organizing Map for handwritten digits recognition." Neural Networks 60 (2014): 104-118.
- [7] Khatri, Sunil Kumar, Shivali Dutta, and Prashant Johri. "Recognizing images of handwritten digits using learning vector quantization artificial neural network." Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), 2015 4th International Conference on. IEEE, 2015.
- [8] Wu, J. (2012). "Advances in K-means clustering: a data mining thinking." Springer Science & Business Media..
- [9] Khan, M. E. (2015). "K-means clustering."
- [10] Juanying Xie, Shuai Jiang 2010, "A simple and fast algorithm for global K-means clustering", 2010 Second International Workshop on Education Technology and Computer Science,

- [11] Decong Yu, Lihong, Ma. "Digit Recognition Based on Multi-features", IEEE Systems and Design Engineering Symposium, 2007.
- [12] Ye Xu, Dezhan Liu: Clustering research of handwritten digit recognition by genetic algorithm. BIC-TA 2010: 291-297.
- [13] Xu Ye, Zhang Wei, "On a Clustering Method for Handwritten Digit Recognition" Third International Conference on Intelligent Networks and Intelligent Systems 2010.
- [14] Wierer, J., Boston, N., "A Handwritten Digit Recognition Algorithm using Two- Dimensional Hidden Markov Models for Feature Extraction," Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on, Volume 2, 15-20 April 2007 Page(s): II-505 - II-508.
- [15] Rasul Enayatifar and Mehdi Alirezanejad 2011 "Off Line Handwriting Digit Recognition by Using Direction and Accumulation of Pixels" International Conference on Computer and Software Modeling IPCSIT vol.14 (2011) IACSIT Press, Singapore.
- [16] Hao Yu, Tiantian Xie, Michael Hamilton, and Bogdan Wilamowski "Comparison of Different Neural Network Architectures for Digit Image Recognition" 4th Conference on Human System Interactions HSI-2011, Yokohama, Japan, May19-21, pp. 98-103, 2011
- [17] Marzieh Moradi, Mohammad Ali Poormina, Farbod Razzazi: FPGA Implementation of Feature Extraction and MLP Neural Network Classifier for Farsi Handwritten Digit Recognition.EMS 2009: 231-234
- [18] Abbas, Q.; Ahmad, J.; Bangyal, W.H., "Analysis of learning rate using BP algorithm for hand written digit recognition application" Information and Emerging Technologies (ICIET), 2010 International Conference on Communication, Networking & Broadcasting.
- [19] Alpaydin, E., and Fevzi Alimoglu. "Pen-based recognition of handwritten digits data set." University of California, Irvine, Machine Learning Repository. Irvine: University of California (1998).
- [20] Suebsing, Anirut, and Nualsawat Hiransakolwong. "A novel technique for feature subset selection based on cosine similarity." Applied Mathematical Sciences 6.133 (2012): 6627-6655. APA
- [21] Ji Qian, Dongping Qian, Mengjie Zhang —A Digit Recognition System for Paper Currency Identification Based on Virtual InstrumentsI IEEE Transactions, 1-4244-0555-6/06,2006
- [22] Gurney, Kevin. "An introduction to neural networks." CRC press, 1997.
- [23] Khan, Y., "Partial discharge pattern analysis using PCA and back-propagation artificial neural network for the estimation of size and position of metallic particle adhering to spacer in GIS." Electr. Eng. 98(1):29-42, 2016
- [24] Lin, B. S., Wu, H. D., and Chen, S. J., "Automatic wheezing detection based on signal processing of spectrogram and back-propagation neural network." J. Healthcare Eng. 6(4):649-672, 2015.