Persian Handwritten Digit Recognition Using Particle Swarm Probabilistic Neural Network

MehranTaghipour-GorjiKolaie[1] Ismaeil Miri[2] Seyyed-Mohammad Razavi[3] Javad Sadri[4]

[1] Ph.D. Candidate, Faculty of Electrical & Computer Engineering, University of Birjand, Birjand, Iran.
mtaghipour@birjand.ac.ir
[2] Ph.D. Candidate, Faculty of Electrical & Computer Engineering, University of Birjand, Birjand, Iran.
miri.esmail@yahoo.com
[3] Associate Professor, Faculty of Electrical & Computer Engineering, University of Birjand, Birjand, Iran.
SMRazavi@birjand.ac.ir
[4] Assistant Professor, Department of Computer Science & Software Engineering Faculty of Engineering and Computer Science Concordia University, Montreal, Quebec, Canada.
[4] Assistant Professor Department of Computer Engineering, Faculty of Electrical and Computer Engineering, University of Birjand, Birjand, Iran.
javad.sadri@cs.mcgill.ca

Abstract:

Handwritten digit recognition can be categorized as a classification problem. Probabilistic Neural Network (PNN) is one of the most effective and useful classifiers, which works based on Bayesian rule. In this paper, in order to recognize Persian (Farsi) handwritten digit recognition, a combination of intelligent clustering method and PNN has been utilized. *Hoda* database, which includes 80000 Persian handwritten digit images, has been used to evaluate our proposed classifier. Obtained results show that PNN is a powerful classifier and excellent choice for classification of Persian handwritten digits. Correct recognition rate when training and testing data have been used directly (without clustering) for training data is 100% and for testing data is 96%, but when k-means has been used as cluster tool and clusters' center have been used as training data, in this case, correct recognition rate for training data is 100% and for testing data is 96.16%. In addition, when Particle Swarm Optimization (PSO) has been used to find optimum clusters for each class of Persian handwritten digits, correct recognition rate in training data is 100% and for the testing data it reaches to 98.18%.

Keywords: Probabilistic Neural Network (PNN), Classification, Persian handwritten digit recognition, Particle swarm optimization, clustering, K-means.

Submission date: 8, Jan., 2013

Conditionally acceptance date: 18, June, 2014

Acceptance date: 11, Oct., 2014

Corresponding author: S. M. Razavi

Corresponding author's address: Faculty of Electrical & Computer Engineering, University of Birjand, Birjand, Iran.

1. Introduction

Nowadays, in the formal and informal documents, bank check and financial contracts, even though electrical equipments are available and can be easily used to type digits, people use their hand to write digits yet. In this case, sometimes, electrical equipments cannot recognize handwritten digits, because of the similarity between two or more digits and may make a mistake. This mistake may create irretrievable problem in the future, therefore, using effective recognition systems and useful classifiers can reduce effects of such mistakes. Persian/Farsi handwritten digits like other languages' digits have its unique style. In this language, some digits have similarity with each other and may cause mistake in recognition process.

So far, different methods are presented in literatures to recognize handwritten digits in variety languages, for example; Cascade Ensemble Classifier [1], Adaptive Neural Network [2], Incremental Class Learning approach [3], Two-Stage Template and Model Based technique [4], Optimized Nearest Neighbor Classifier [5], Class-Specific Feature Polynomial (CFPC) [6], Generative Model [7] and a simple structure of neural network [8]. Also, some methods have been used to improve features in pre-processing step, such as; Deformable Templates [9] and Aspect Ratio Adaptive Normalization (ARAN) [10]. Recently, most of the literatures focused on preprocessing stage and uses K-Nearest Neighbor (KNN) [11] and Support Vector Machine (SVM) [12] as classifier for handwritten digit recognition. Optimal features with promising discrimination are extracted using Genetic Algorithm applied on local regions of digit's image. Then SVM is used to classified hand written digits in [12]. Biologically inspired features are used in [13] to recognize handwritten digits by SVM. Wavelet transform is used as a feature extraction method in [14] and then SVM use as classifier. A combination of Convolutional Neural Network (CNN) and Support Vector Machine (SVM) is used as a model to recognize handwritten digits in [15], his model automatically retrieves features based on the CNN architecture, and recognizes the unknown pattern using the SVM recognizer. Other methods which are proposed in literatures are: A kernel combined with Bayesian discriminant in the subspace spanned by the eigenvectors which are associated with the smaller eigenvalues in each class is adopted as the classification criterion in [16]. In [17], Bayesian supervised dimensionality reduction (BSDR) method that combines linear dimensionality reduction and linear supervised learning is used to recognize handwritten digits. Hidden Markov Model (HMM) [18] and Heterogeneous handwritten digits representation model based on multiple instance learning (MIL) based classifier [19] are other methods that introduce to solve recognizing handwritten digits. Also, some literatures focus their research on Farsi/Persian handwritten digits, like [20], where, Persian handwritten digits are recognized using combination of two multiple classifiers, which are based on statistic structure. Also, the characterization loci and neural network have been used as main feature and classifier, respectively [21]. In addition, variety structure of Multi Layer Perceptron (MLP) have been used to recognize Persian handwritten digits, some of them have been reported in [22-23]. The other tool that can be used in this field is Fuzzy logic, as it can be seen in [24] fuzzy logic and a clustering method along with MLP are utilized for recognizing Persian handwritten digits. Also, pre-processing on Persian handwritten digits database can improve recognition rate, like [25] and or in [26], the outer profiles of the digits' image which are calculated at multiple orientations has been used as main feature and then support vector machine has been used as classifier. SVM is used to recognize Persian handwritten digits in [27] and [28], Zoning features and projection histogram for feature extraction using Support Vector Machine (SVM) with three liner and non-linear kernels used to classify digits. Singular Value Decomposition (SVD) algorithm has been used in [29], this method uses several parts to solve problem, many preprocessing techniques, computational burden algorithm and not promising results in comparison of our method.

In this paper, probabilistic neural network has been introduced to classify Persian handwritten digits, even though some papers such as [30] that uses same method in recognition of digits in other languages. But it is noteworthy that each language has its unique style and calligraphy and perhaps one method which is presented for digits of one language (e.g. Farsi) is not proper and useful for digits of other languages (e.g. Urdu or Arabic).In this paper to improve validation process of proposed method *Hoda* database, which consists of 80000 samples of Persian handwritten digits, has been used. 40000 samples out of 80000 have been used as training dataset and 20000 have been used as validation dataset and 20000 samples have been used as testing dataset. In addition, using PNN in usual processors make memory problem and it is impossible to use large database for training PNN. Therefore, in order to remove memory problem, an intelligent clustering method using Particle Swarm Optimization (PSO) has been introduced.

In the rest of this paper, in the second section, overview of probabilistic neural network, k-means cluster and particle swarm optimization have been presented, respectively. In section three, proposed method has been discussed. Moreover, obtained results have been illustrated in section four.

2. Principle Concepts

2.1 Overview of Probabilistic Neural Network (PNN)

Probabilistic neural network is one of the artificial neural networks, which is based on Bayesian rule and concept [31]. PNN is also known as Bayesian network, belief network and knowledge map. As Fig. 1 shows, PNN uses feed forward topology with two hidden layer and one output layer. Hidden layers are called "*pattern layer*" and "*summation layer*", respectively. In this kind of ANN, each neuron is corresponded to each input data, which contains a specific knowledge of that input data. This knowledge is considered in Gaussian distribution, which is represented in (1).

$$f(I_1, I_2, ..., I_n) = e^{\left(\frac{-\sum_{L=1}^{n}(I-I_L)}{2\delta^2}\right)} \quad (1)$$

Where, $I_1, I_2, ..., I_n$ are the features of an input data, δ^2 is called "smoothing parameter or spread" which has same concept as variance.

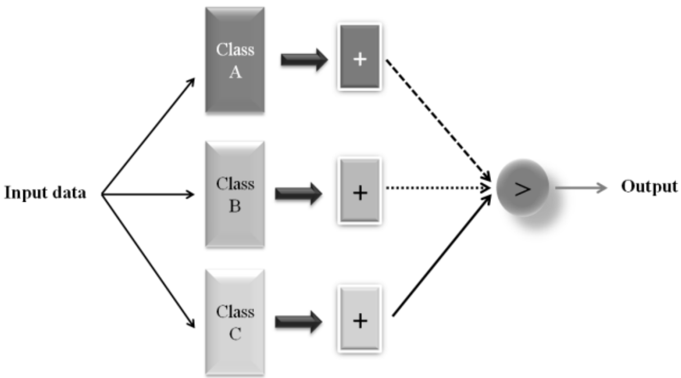


Fig. 1. A typical structure of PNN

As shown in Fig. 2, in the second hidden layer, distributions of i^{th} class are added together by (2):

$$F_{c_i} = \sum_{j=1}^{k} f_i \quad (2)$$

Where, f_j is corresponded Gaussian distribution to j^{th} input data of class C_i.

In the output layer, PNN uses Bayesian decision rules to compare the outputs of previous layer and assigns the correct class for input data.

PNN has some advantages and disadvantages like all classifiers:

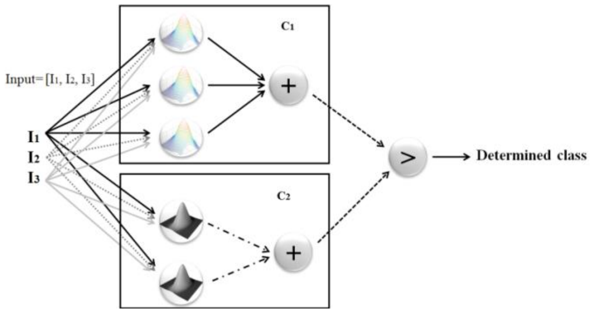


Fig. 2. An example of two-class PNN

- Advantages:
  - ✓ All process in this kind of ANN is done parallel;
  - ✓ It does not need to computational burden process for training, such as; Back Propagation and Gradient descant. Therefore, PNN is trained faster than other ANNs.
- Disadvantages
  - ✓ Most of the time, it is only used for classification applications;
  - ✓ It requires large memory;
  - ✓ It requires a representative training set.

2.2 Overview of K-means Cluster

K-means is one of the oldest and simplest unsupervised learning algorithms, which is used for clustering problems. Different topologies have been proposed for this algorithm; however, all of them have two main ideas:

- To define K-centers, for K clusters;
- To assign each input data to a cluster that has the nearest center to this input data.

In the simple format of K-means; first, corresponding to clusters, centers have been randomly determined, then, data will be assigned to a cluster based on their distance with primary random centers of clusters. This process is repeated in each iteration and new updated centers are determined by using moderating from all data points. In this process, the algorithm tries to minimize objective function expressed in (3):

$$Objective\,function = \sum_{j}^{k}\sum_{i}^{n} || I_i^j - c_j ||^2 \quad (3)$$

Where, $||.||^2$ calculates distance between a data point $x_i(j)$ and center of j^{th} cluster c_j.

2.3 Overview of Particle Swarm Optimization (PSO)

PSO simulates the behaviors of bird flocking. In PSO, each single solution is considered as ‘‘bird’’ in the search space, which is called as ‘‘particle’’. Fitness value has been calculated for all of the particles, which are evaluated by the fitness function to be optimized, and have velocities, which direct the flying of the particles [31]. The particles are ‘‘flown’’ through the problem space by following the current optimum particles [32-33]. In each iteration, each particle is updated by following two ‘‘best’’ values. The first one is the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called "pbest". Another ‘‘best’’ value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the population. This best value is a global best which is called "gbest". After finding the two best values, the particle updates its velocity and positions based on (4) and (5):

$$V_{new} = W * V_{old} + C_1 * r_1*(P_{pbest} - X_{cs}) + C_2 * r_2*(P_{gbest} - X_{cs}) \tag{4}$$

$$X_{new} = X_{old} + V_{new} \tag{5}$$

Where ,W is the inertia weight, V_{new} is the particle velocity, X_{cs} is the current particle (solution) of each particle, P_{pbest} and P_{gbest} are positions of pbest and gbest, coefficient r_1 and r_2 are random numbers which can be between 0 and 1. Coefficients C1, C2 are learning factors. Particle’s velocities on each dimension are clamped to a maximum velocity V_{max}. If the sum of accelerations would cause the velocity on that dimension to exceed V_{max}, which is a parameter specified by the user then the velocity on that dimension is limited to V_{max}. In Fig. 3, typical movement of one particle in solution space has been shown.

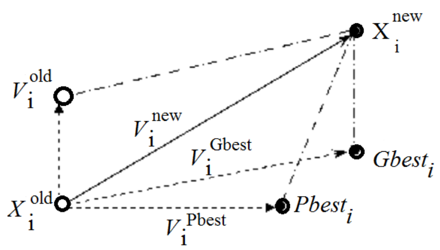


Fig. 3. Typical movement of one particle in solution space

3. Proposed Method

In this paper, Hoda database, which is gathered by Prof. Kabir and Dr. Khosravi has been used [32].This standard database includes binary images of 102352 digits which is obtained from 11942 registration form, in this paper 80000 samples out of 102352 for evaluation of proposed classifier, Fig. 4 shows some typical samples of Persian handwritten digits.

Using zoning method, required features have been extracted from these binary images. In this method, the frame of character is divided into overlap and non-overlap segments. Density of points or other features from segments are used to feature extraction from that character. These features are extracted by dividing image into diameter and rectangular segments and then chain codes histograms are calculated in each segment. In this paper, digits have too much zero points (i.e. zero points belongs to background that is empty from information).

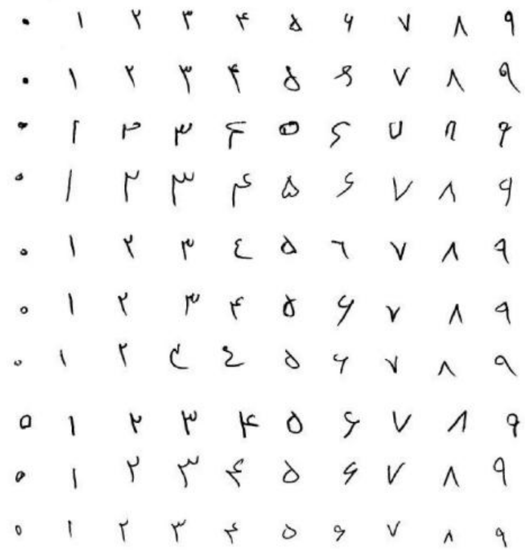
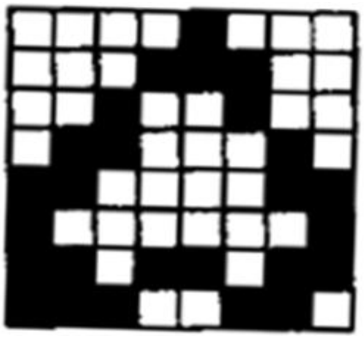


Fig. 4. Some typical samples of Persian handwritten digits

In order to extract zoning features, sub segments which contains redundant information is removed and each digit is limited to one quadrant. Here, each digit is normalized to 32 × 32 images, as dimension point of view (this normalization means that each image is converted to given m × n dimension image). Used method for image normalization is that 32 × 32 image is divided into sub segments, and then points are counted in each sub segment, and it is considered as feature. For example, 4 × 4 blocks are chosen, therefore, the image will contain 64 blocks that each block has 16 points. Black points counted as feature, thus, each digit contains 64 features that each feature can be a number between 0 to 16.

For example we want to extract zoning feature from digit five as shown in Fig. 5 (a). Fig. 5 (a) is normalized to 8 × 8 image. Image is divided into 4 × 4 blocks. Then by counting black points in each block figure 5(b) is calculated. According to Fig. 5 (b), feature matrix has been calculated, which can be seen in matrix (6), and then it can be converted to feature vector such as vector (7).

Since, using large database for PNN make memory problem, two conditions have been considered for using PNN. In the first condition, 10000 samples out of 80000 samples have been used to train PNN and 70000 samples have been used as testing data. For better validation, it is necessary to use more samples for training step, however PNN do not let us do that. Therefore, in second condition, 40000 samples have been used to train PNN, but not original data. In order to train PNN, centers of considered clusters for each class have been used for training step, as Fig. 5 shows block diagram of proposed method.



(a) Persian handwritten digit 5

$$\begin{array}{cccc|cccc}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
\hline
1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 1 & 1 & 0
\end{array}$$

(b) Extracted features from image

Fig. 4. Applying Zoning Method on Persian handwritten digit 5

$$\begin{bmatrix} 4 & 5 \\ 9 & 8 \end{bmatrix} \tag{6}$$

$$\begin{bmatrix} 4 \\ 5 \\ 9 \\ 8 \end{bmatrix} \tag{7}$$

Fig. 5. Block diagram of basic proposed method

In this step, in the first study, 60 clusters have considered for each class using K-means algorithm and in the second study, we let PSO to determine optimum number of clusters in each class, namely; the number of clusters in each class have been considered as particles of PSO and correct recognition rate has been considered as fitness function. As shown in Fig. 6, in each iteration, the proposed number of clusters by PSO is fed into clustering step for training dataset and after that PNN is trained by calculated centers, and then in the evaluation step, validation dataset have been tested by trained PNN, and this procedure continues until best recognition rate is obtained.

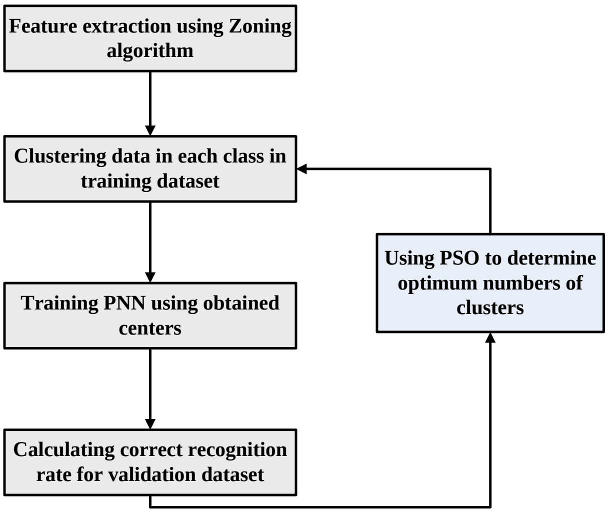


Fig. 6. Using PSO to determine optimum numbers of clusters for each class

3.1 Advantages of proposed method

Week performance in training steps and consequently not reliable results in testing step is one of the problems in introduced methods for handwritten digits recognition. Most of the proposed method maybe presents good results in limited dataset but when dataset become large, the performance of them is reduced. Hoda dataset is one of the largest dataset in Farsi handwritten digits, which include different type of Persian handwritten digits. Most of the proposed method for this dataset cannot deal with the computational burden, especially in training step.

Proposed method shows promising results that can be more improved yet. The advantages of proposed method can be mentioned as follow:

- Probabilistic Neural Network (PNN) is one of the most powerful classifier which map information and adapt itself by training data. Many applications using this classifier reports its ability to present high recognition rate for training data.
- The big problem of PNN is when it encounters a large number of training data, therefore feature vectors that can be extracted from clustering is proposed in this paper.
- Center of each cluster is an average of features in cluster and can be a good representation of cluster. Center of each cluster is used as training feature vectors.
- Each class of Persian handwritten digits has its own nature and it is impossible to use same number of clusters for them. In this paper, PSO is introduced to find optimum number of cluster for each class (or in other word, optimum feature vectors for training step).

4. Experimental Results

As mentioned before, this paper can be divided into two main parts; in the first part, the number of training data is low, because of memory problem. Performance of proposed method is evaluated by Correct Recognition Rate (CRR). It is worth noting that results for proposed method based on PSO clustering, is presented in minimum, maximum and average results. In order to reach validated results using PSO, 25 separated runs have been fulfilled in this paper.

Hoda dataset has been as one of the biggest Persian handwritten dataset by more than 100000 samples. According to other literatures, from these samples 60000 samples are used in training step and 20000 are used in testing step. In this paper, in order to present honest results, 20000 data out of 60000 are used for validation and clustering is applied on remained 40000 samples.

4.1. Persian handwritten recognition by using PNN without clustering

Table 1 (see Appendix) shows obtained results for recognition of Persian handwritten digits. Results show that by increasing training data, correct recognition rate for testing data will be increased. As mentioned in section 2, one of the PNN's works is mapping knowledge of input data to its hidden layer; therefore, increasing of training data helps PNN to classify testing data better. It is noteworthy that it is impossible to use more than 10000 samples for training PNN for our processor. Also, 20000 samples out of 80000 have been used as testing data. In following test in this part, spread is four.

4.2. Persian handwritten recognition by using PNN with clustering

In order to solve memory problem, we use K-means algorithm to cluster data in each class. Then centers of each class have been used as training data. In this part, 60 clusters have been considered for each class, namely; for example, sample data of class "0" are clustered into 60 clusters. As Table 2 (see Appendix) shows by increasing training data correct recognition rate for testing data is increased.

We do not know if these clusters for each class are optimum clusters or not? Therefore, PSO has been used to find the optimum clusters for each class. Table 3 (see Appendix) shows the optimum clusters which obtained by PSO. In addition, Fig. 7 shows the convergence curve of using PSO, which is explained in section 3. This typical curve shows the convergence of PSO on validation data. In order to use PSO considered parameters are as follow: C1=1.9 and C2=2.1, the number of particles= 40, maximum iteration = 50 and inertia weight (W) = 0.99.

Table 4 (see Appendix) shows the maximum, average and minimum obtained results of correct recognition rate for training and testing data. In addition, confusion matrix of testing data for optimum result has been presented in Tables 5 (see Appendix). In this matrix, rows are actual output of PNN in testing step and columns are target or real labels, namely; the numbers in the diameter of this matrix are correct classified data.

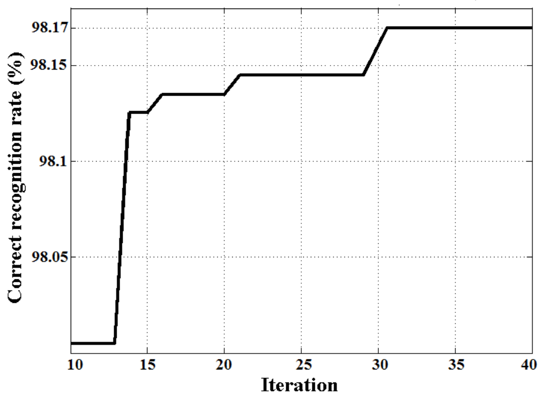


Fig. 7. A typical Convergence curve of correct recognition rate for validation data

4.3. Comparison with Other Methods

In order to evaluate the ability of proposed method for Persian handwritten recognition, obtained results have been compared with Feed Forward Back Propagation Neural Network (FFBPNN) (whit three layers; two hidden layer and on output layer 90-70-10) and other methods presented in literatures. As it can be seen in Table 6 (see Appendix), correct recognition rate for training data when FFBPNN is used is less than PNN, but correct recognition rate for testing data is bigger than PNN with simple clustering method. However, it is less than correct recognition rate when PSO has been used to find optimum clusters. It is worth noted that for testing FFBPNN, centers of clusters have been used as training data, as well.

5. Conclusion

In this paper, probabilistic neural network has been presented to classify Persian handwritten digits. Since, PNN maps the knowledge of input data in its hidden layer, it is predictable that it can presents very good performance for training data, however, memory problem limits the useable training data. Therefore, K-means has been introduced to solve as a clustering algorithm, and then, centers of each class have been used as training data. In addition, in order to improve proposed method PSO has been used to find optimum numbers of clusters in each class. Obtained results show that proposed method presents the best recognition rate for training data and very good and promising recognition rate for testing data. Moreover, it is possible to improve this recognition rate by using better controlling parameter for PSO or using more powerful optimization methods.

References

[1] P. Zhang, T. D. Bui and Ch. Y. Suen, “A novel cascade ensemble classifier system with a high recognition performance on handwritten digits” Pattern Recognition, Vol. 40, No. 12, pp. 3415-3429, 2007.

[2] M. Kang and D. Palmer-Brown, “A modal learning adaptive function neural net-work applied to handwritten digit recognition” Information Sciences, Vol. 178, No. 20, pp. 3802-3812, 2008.

[3] J. Mandziuk and L. Shastri, “Incremental class learning approach and its application to handwritten digit recognition” Information Sciences, Vol. 141, No. 3-4, pp. 193-217, 2002.

[4] P. Gader, B. Forester, M. Ganzberger, A. Gillies, B. Mitchell, M. Whalen and T. Yocum, “Recognition of handwritten digits using template and model matching” Pattern Recognition, Vol. 24, No. 5, pp. 421-431, 1991.

[5] H. Yan, “handwritten digit recognition using an optimized nearest neighbor classifier” Pattern Recognition Letters, Vol. 15, No. 2, pp. 207-211, 1994.

[6] Ch. Liu and H. Sako, “Class-specific feature polynomial classifier for pattern classification and its application to handwritten numeral recognition” Pattern Recognition, Vol. 39, No. 4, pp. 669-681, 2006.

[7] M. Revow, C. K. I. Williams and G. E. Hinton, “Using generative models for handwritten digit recognition” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 6, pp. 592- 606, 1996.

[8] S. Knerr and L. Personnaz and G. Dreyfus, “Handwritten digit recognition by neural networks with single-layer training” IEEE Transactions on Neural Networks, Vol. 3, No. 6, pp. 962 – 968, 1992.

[9] A. K. Jain and D. Zongker, “Representation and recognition of handwritten digits using deformable templates” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 12, pp. 1386 – 1390, 1997.

[10] Ch. Liu, K. Nakashima, H. Sako and H. Fujisawa, “Handwritten digit recognition: Investigation of normalization and feature extraction techniques” Pattern Recognition, Vol. 37, No. 2, pp. 265-279, 2004.

[11] U. R. Babu, Y. Venkateswarlu, and A. K. Chintha “Handwritten Digit Recognition Using K-Nearest Neighbor Classifier” World Congress on Computing and Communication Technologies (IEEE Conference), pp. 60-65, 2014.

[12] N. Da's, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, “A genetic algorithm based region sampling for selection of local features in handwritten digit recognition application” Applied Soft Computing, Vol. 12, pp. 1592-1606, 2012.

[13] A. Cardoso , and A.Wichert, “Handwritten digit recognition using biologically inspired features” Neurocomputing, Vol. 99, pp. 575-580, 2013.

[14] M. S. Akhtar, and H. A. Qureshi, “Handwritten Digit Recognition Through Wavelet Decomposition and Wavelet Packet Decomposition” Eighth International Conference on Digital Information Management (ICDIM) (IEEE Conference), pp. 143-148, 2013.

[15] X. X. Niu, and Ch. Y. Suen, “A novel hybrid CNN–SVM classifier for recognizing handwritten digits” Pattern Recognition, Vol. 45, pp. 1318-1325, 2012.

[16] Y. Wen, and L. He, “A classifier for Bangla handwritten numeral recognition” Expert Systems with Applications, Vol. 39, pp. 489-953, 2012.

[17] M. Gönen, “Bayesian Supervised Dimensionality Reduction” IEEE TRANSACTIONS ON CYBERNETICS, VOL. 43, NO. 6, pp. 2179-2189, 2013.

[18] A. H. Ko, P. R. Cavalin, R. Sabourin, and A. de Souza Britto Jr, “Leave-One-Out-Training and Leave-One-Out-Testing Hidden Markov Models for a Handwritten Numeral Recognizer: The Implications of a Single Classifier and Multiple Classifications” IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 31, No. 12, pp. 2168-2178, 2009.

[19] Y. Hanning, and W. Peng, “Handwritten Digits Recognition Using Multiple Instance Learning” IEEE International Conference on Granular Computing (GrC), pp. 408-411, 2013.

[20] R. Ebrahimpour, and F. Shari_zadeh, “Persian Handwritten Digit Recognition with Classifier Fusion: Class Conscious versus Class in different Approaches” World Academy of Science, Engineering and Technology, Vol. 57, pp. 552-559, 2009.

[21] R. Ebrahimpour, M. R. Moradian ,A. Esmkhani and F. M. Jafarlou, “Recognition of Persian handwritten digits using Characterization Loci and Mixture of Experts” International Journal of Digital Content Technology and its Applications, vol. 3, No. 3, pp. 42-46, 2009.

[22] S. Moza_ari and H. Soltanizadeh, “ICDAR 2009 Handwritten Farsi/Arabic Character Recognition Competition” 10th International Conference on Document Analysis and Recognition, pp. 1413-1417, 2009.

[23] M. Moradi, M. A. Pourmina and F. Razzazi, “A New Method of FPGA Implementation of Farsi Handwritten Digit Recognition” European Journal of Scientific Research, vol. 39, No. 3, pp. 309-315, 2010.

[24] S. M. Razavi, M. Taghipour and E. Kabir, “Improvement in Performance of Neural Network for Persian Handwritten Digits Recognition Using FCM Clustering” World Applied Sciences Journal, Vol. 9, No. 8, pp. 898-906, 2010.

[25] Ch. Liu and Ch. Y. Suen, “A new benchmark on the recognition of handwritten Bangla and Farsi numeral characters” Pattern Recognition, Vol. 42, No. 12, pp. 3287-3295, 2009.

[26] H. Soltanzadeh and M. Rahmati, “Recognition of Persian handwritten digits using image profiles of multiple orientations” Pattern Recognition Letters, Vol. 25, No. 14, pp. 1569-1576, 2004.

[27] G. Mirsharif, M. Badami, B. Salehi, and Z. Azimifar, “Recognition of Farsi Handwritten Digits Using a Small Feature Set” International Journal of Computer and Electrical Engineering, Vol. 4, No. 4, pp. 588-591, 2012.

[28] A. Nooraliei, “Persian Handwritten Digits Recognition by Using Zoning and Histogram Projection” 3rd Joint

Conference of AI & Robotics and 5th RoboCup Iran Open International Symposium (RIOS), pp. 1-5, 2013.
[29] H. Salimi, and D. Giveki, “Farsi/Arabic handwritten digit recognition based on ensemble of SVD classifiers and reliable multi-phase PSO combination rule” International Journal on Document Analysis and Recognition (IJDAR), Vol. 16, No. 4, pp. 371-386, 2013.
[30] F. A. Al-Omari and O. Al-Jarrah, “Handwritten Indian numerals recognition system using probabilistic neural networks” Advanced Engineering Informatics, Vol. 18, No. 1, pp. 9-16, 2004.
[31] H. Yaghobi, H. Rajabi Mashhadi,and K. Ansari, “Application of Radial Basis NeuralNetworks in Fault Diagnosis of Synchronous Generator’ Journal of Iranian Association of Electrical and Electronics Engineers, Vol.10, No.2, pp. 23-36, 2013.
[32] A. Arefi, M. Haghifam, S. H. Fathi, “Short-Term Nodal Congestion Price Forecasting in Large-Scale Power Market Using ANN with Genetic Optimization Training” Journal of Iranian Association of Electrical and Electronics Engineers, Vol. 11, No. 1, pp. 37-48, 2012.
[33] M. Joorabian, and E. Afzalan, “Optimal Power Flow under Both Normal and Contingent Operation Using Hybrid Particle Swarm Optimization and Nelder Mead Algorithms (PSO-NM)”Journal of Iranian Association of Electrical and Electronics Engineers, Vol. 12, No. 1, pp. 111-124, 2015.
[34] H. Khosravi and E. Kabir, “Introducing a very large dataset of handwritten Farsi digits and a study on their varieties” Pattern Recognition Letters, Vol. 28, No. 10, pp. 1133-1141, 2007.

Appendix

In this section tables of this paper can be seen.

Table. 1. Obtained results for Persian handwritten recognition by using PNN without clustering

The number of training data	Misclassified training data	Correct recognition rate for training data (%)	Correct classified testing data	Correct recognition rate for testing data (%)
5000	0	100	19080	95.4
8000	0	100	19186	95.93
10000	0	100	19200	96

Table. 2. Obtained results for Persian handwritten recognition by using PNN with clustering

Classification method	Misclassified training data	Correct recognition rate for training data (%)	Correct classified testing data	Correct recognition rate for testing data (%)
PNN	0	100	19238	96.16

Table. 3. Optimum clusters for each class by using PSO

Class (digit)	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
Optimum clusters	141	136	150	159	182	156	178	159	168	197

Table. 4. Obtained results for Persian handwritten recognition by using PNN with clusters for 25 separated runs

Classification method	Correct classified training data			Correct recognition rate for training data (%)			Correct classified testing data			Correct recognition rate for testing data (%)		
	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.	Min.	Avg.	Max.
PNN	20000	20000	20000	100	100	100	19468	19553.88	19636	97.34	97.76	98.18

Table. 5. Confusion matrix for testing data with optimum clusters

PNN's output \ Real outputs (labels)	0	1	2	3	4	5	6	7	8	9
Zero (۰)	1949	33	0	0	0	15	0	3	0	0
One (۱)	37	1952	6	0	0	0	2	0	0	3
Two (۲)	0	8	1926	51	12	0	0	0	0	3
Three (۳)	0	0	34	1944	16	2	3	0	1	0
Four (۴)	0	3	23	21	1953	0	0	0	0	0
Five (۵)	10	4	3	0	0	1980	0	1	1	1
Six (۶)	0	9	1	0	1	2	1978	0	1	8
Seven (۷)	0	1	5	0	3	0	2	1987	0	1
Eight (۸)	0	4	0	0	0	1	0	0	1991	4
Nine (۹)	1	10	3	0	2	0	8	0	1	1976

Table 6: Comparison proposed method with other methods

Classification method	Correct recognition rate for training data (%)	Correct recognition rate for testing data (%)
PNN without clustering	**100**	96
PNN with simple clustering	**100**	96.16
PNN with clustering using PSO	**100**	**Max.: 98.18** **Avg.: 97.76**
FFBPNN without clustering	97.6	94.96
FFBPNN with simple clustering	97.28	96.63
FPGA [23]	97.28	96