International Journal of Innovation and Scientific Research
ISSN 2351-8014 Vol. 7 No. 1 Aug. 2014, pp. 50-56

http://www.ijisr.issr-journals.org/

A Comparison between the Self-Organizing Maps and the Support Vector Machines for Handwritten Latin Numerals Recognition

R. SALOUAN, S. SAFI, and B. BOUIKHALENE

Department of Mathematic and Informatic, Polydisciplinary Faculty,
Sultan Moulay Slimane University, Beni Mellal, Morocco

ABSTRACT: In this paper, we present a comparison between two methods for learning-classification; the first one is called Kohonen network or Self-Organizing Maps (SOM) which is characterized by an unsupervised learning. The second one is called Support Vector Machine (SVM) which is based on a supervised learning. These techniques are used for recognition of handwritten Latin numerals that's extracted from MNIST database. In the pre-processing phase we use the thresholding, centering and skeletization techniques in the features extraction we use the zoning method. The simulation result demonstrates that the SVM is more robust than the SOM method in the recognition of handwritten numerals Latin.

KEYWORDS: Handwritten Latin numerals, Thresholding, Centering, Skeletization techniques, Self-Organizing Maps (SOM), Support Vectors Machines (SVM).

1 INTRODUCTION

Handwritten Latin numeral recognition is considered as a one of the most active field in pattern recognition in reason of its usage uses in many applications including those in postal sorting, bank cheque processing and automatic data entry.

Many approaches have been proposed by the researchers towards recognition of handwritten characters or Latin numerals by using the zoning [1-5] method in the features extraction or the SOM [6-10] or the SVM [11-14] in the learning-classification phase.

In fact, a recognition system can be divided into three principal steps. The first one is a pre-processing which is used for enhancing the image quality. The second step is the features extraction from each numeral pattern for to extract a quantity of informations from each numeral. The third step is the learning-classification which is used to recognize the Latin numerals.

In this work, all numeral images are pre-processed by the thresholding and the centering techniques then a skeletization of them is carried. The features extraction is used by the zoning method which is used to convert each image of numeral to a vector that will used as an input vector of SOM then of SVM that are used to train the numeral images of the training database and to classify them in the test database. The last steps contain two methods which are described as follow:

1.1 DESCRIPTION OF THE SOM METHOD

In learning phase, each input vector that models a numeral should beards the label of a neuron called the winner neuron, it's that has a vector weights is very nearest to this vector after calculation of the Euclidean distance between this input vector and all vectors weights that binds each vector of the input layer of the SOM and each neuron of the output layer of the SOM. The vector weight that's closest must be learned be more near to the input vector that's nearest to it. After learning we will get for each numeral the very nearest vector, all these vectors weight learned should be stocked just for forming a learning base.

In the classification phases we calculate the Euclidean distance between a test vector (unknown numeral) and each of vectors weights that was saved in the learning base. The recognition will assigned to label of the neuron that its vector weight is nearest to test vector.

1.2 DESCRIPTION OF THE SVM METHOD

In the learning phase, we use the SVM method which is based on the strategy: one against all. A separation must be making for each numeral image that belongs in a class has a label equal to value 1 of the learning base to all other numeral images that are regrouped in another class labeled by -1. This separation (maximizing the margin between both classes) is therefore creating a decision function separating these both classes. We have ten numerals. So we will have ten decision functions each of them will separate a pair of classes (1 and -1) among the ten pairs.

In the classification phase, we calculate the image of an unknown vector by all ten decision functions. The recognition will be attributed to the numeral whose decision function separates its class to another class containing the rest of numerals which gives the biggest value.

2 THE RECOGNITION SYSTEM

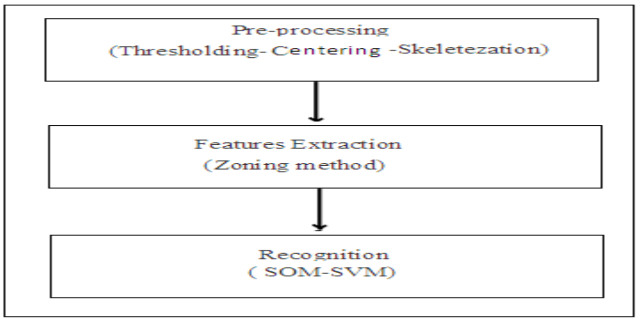

Fig. 1. System for handwritten Latin numerals recognition.

3 PREPROCESSING

Pre-processing is the first part of Latin numerals recognition system that's used for producing a cleaned up version of the original image so that it can be used efficiently in the feature extraction step.
In our study, we preprocess the images by these techniques; the first is the thresholding that's used to make each numeral image contains only the black and white colors according to preset threshold. The second steps is the centering, used for numeral which is in center of image, the skeletonization is used for detecting the skeleton of each numeral image.

4 FEATURES EXTRACTION

It is very important to extract the features in such a way that the recognition of numerals becomes easier on the basis of individual features of the numerals. In this work we have used the Zoning method that can be explained as follow:
Given a black image that contains an numeral written in white, the zoning method consists to divide this image to a several zones then calculating in each of them the number of white pixels, all these numbers are stocked in a vector, that is to say: the image is converted to a vector has a number of components equal to that of zones.

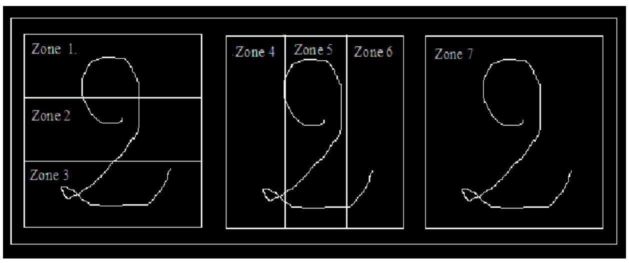

Fig. 2. Example of zoning method of handwritten Latin numeral 2

5 LEARNING PHASE

5.1 THE KOHONEN NETWORK

The Kohonen network [15] is composed of two layers one has I nodes that's the input of network, other has a J nodes that's its output. These layers are connected via IxJ coefficients called weights W .

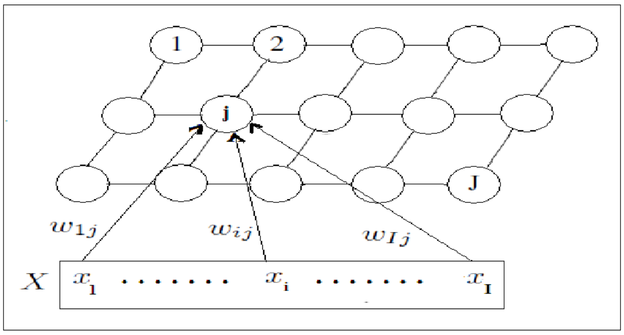

Fig. 3. The Kohonen network

The topological maps of Kohonen (Self-Organizing Maps)weighed a special structuring to its neurons (nodes).This structure binds the neurons and have forced them to respect a certain topology during the learning phase .Thus the near data in the input space have an very closest representations in topological Kohonen map.

5.1.1 LEARNING ALGORITHM OF TOPOLOGICAL KOHONEN MAPS:

It contains the following phases:

- Initialize the weights randomly $W_j^0 : j \in [1, J]$
- Presentation of data $X = (x_1, x_2, ..., x_I)^t$ in input to the current iteration n and calculating its distance from each of the vectors $W_j^n \quad j \in [1, J]$
- Selection of winner neuron j^* that is nearest to the input X by computing the distances:

$$d_j^2 = \sum_{i=1}^{I} (x_i(t) - W_{ij}(t)))^2 \tag{1}$$

- Update he weights W_j^n

The result obtained W^* after the training (learning) phase is a memory containing a set of an optimal weight vectors that are very nearest to each input vector $X = (x_1, x_2, ...x_I)^t$

5.2 THE SUPPORTS VECTORS MACHINES

An SVM [16] is basically defined for two-class problem separation, and it finds an optimal hyperplane which can maximize the distance, the margin, between the nearest examples of both classes, named support vectors (SVs). Given a training database of M data: $(X_i, i = 1,2.....M)$.

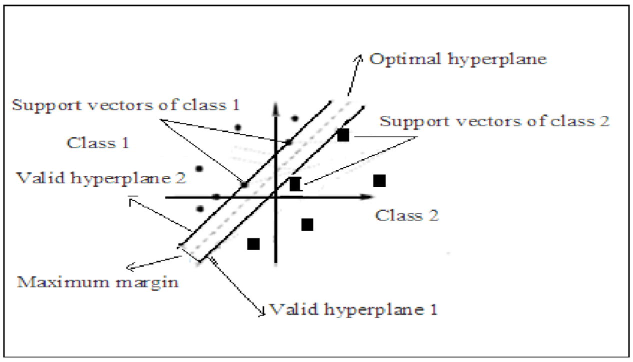

Fig. 4. The determination of optimal hyperplane, vectors supports, maximum Marge and valid hyperplanes.

The linear SVM classifier is then defined as:

$$f(x) = \omega x + b \tag{2}$$

Where w and b are the parameters of the classifier y is the label.

The linear SVM can be extended to a non-linear classifier by replacing the inner product between the input vectors x and the SVs, through a kernel function K defined as:

Kernel linear: xy

Kernel polynomial of degree n: $(xy + 1)^n$

Gaussian Radial Basis Function (GRBF): $e^{-\frac{\|x-y\|^2}{2\sigma^2}}$

The method described above is designed for a problem of two classes only, many studies treat a generalization of the SVM to a multi-classification [17] among these studies we cite the two strategies frequently used: the first approach is based to use N decision functions (one against all) allowing to make a discrimination of a class contains a one vector against all other vectors existed in a other class opposite. The decision rule used in this case is usually the maximum such that we will assign an unknown vector X into a class associated with an output of SVM is the largest.

The second method is called the one against one instead of learning N decision functions; each class is opposed against another. So $\frac{N(N-1)}{2}$ decision functions are learned and each of them performs a voting for the assignment of a new test (unknown) vector X. its class then becomes the majority class after the vote.

6 THE CLASSIFICATION PHASE

- By using the SOM :

During the classification phase, an unknown vector should be presented $X = (x_1, x_2, \ldots, x_I)^t$ and determining the very nearest node in terms of Euclidean distance between X and each optimal vector W^* the unknown vector X will bear the label of the class of the winner neuron.

- By using the SVM :

After having built the ten decision functions between the ten pairs of classes in the learning phase by the strategy of (one against all) we calculate all the values of the images of the vector that models the numeral test by the all the ten decision functions, the recognition will be assigned to the numeral whose an decision function separating its class to another class contains the rest of the other numerals that gives the largest value among all values calculated of the ten images of the numeral test.

7 EXPERIMENTS AND RESULTS

We choose the sizes of all numeral images 24x24 pixels. We have used in the learning and test databases 1200 numeral images. According the features extraction phase, we have divided each numeral image to 3 horizontal zones, and 3 vertical zones, the 7^{th} zone is all the size of image (see figure 2). In each zone we count the number of white pixels which allows converting each image to a vector of 7 components. We have used in the learning-classification phase the kernel function GRBF with a standard deviation σ = 12.Our goal is to compare between the performances of SOM and SVM in this recognition. We group in the following table the values of:

The rate τ_n and the time t_n of recognition for each numeral N by using SOM and SVM.

The global rate τ_g and the time t_g of recognition for all numerals by using SOM and SVM.

Table 1. The rates τ_n and times t_n of recognition the or each numeral, and the global rate τ_g and the global time t_g of recognition for all numerals by using SOM and SVM.

N	(SOM)		(SVM)	
	τ_n	$t_{n\,(s)}$	τ_n	$t_{n\,(s)}$
0	60.00%	6.885	90.00%	114.377
1	95.00%	6.678	100.0%	115.027
2	93.33%	6.802	96.66%	106.847
3	56.67%	6.996	77.33%	100.529
4	65.00%	6.388	78.33%	105.324
5	65.30%	6.489	70.00%	112.550
6	61.66%	6.356	73.30%	104.215
7	80.00%	6.392	81.66%	107.476
8	66.67%	6.207	78.67%	100.125
9	58.33%	6.202	76.66%	101.221
τ_g / t_g	70.19%	65.40	82.26%	103.314

The associated graph to table above is:

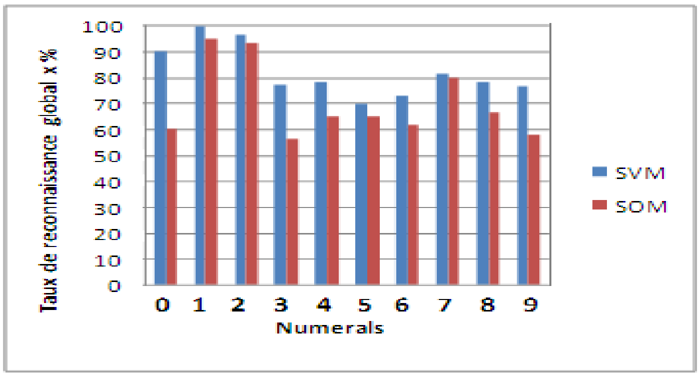

Fig. 5. The recognition rate of each numeral for SOM and SVM

- **Analysis of obtained results:**

These results show that generally, for each numeral, the recognition rate for each numeral also the global rate by using SVM is more than those that when using SOM in one hand, about time recognition the SOM is more fast than SVM in other hand. Now, we present firstly the matrix of confusion that is associated to SOM:

Table 2. The matrix of confusion of SOM

SOM	0	1	2	3	4	5	6	7	8	9
0	60.00 %	00.00%	00.00%	03.66%	00.00%	28.00%	00.00%	00.00%	08.34%	00.00%
1	00.00%	95.00%	00.00%	05.00%	00.00%	00.00%	00.00%	00.00%	00.00%	00.00%
2	04.00%	00.00%	93.33%	00.00%	00.00%	02.67%	00.00%	00.00%	00.00%	00.00%
3	15.33%	04.00%	00.00%	56.67%	00.00%	14.50%	00.00%	06.50%	03.00%	00.00%
4	03.50%	04.25%	00.00%	00.00%	65.00%	15.00%	00.00%	06.75%	05.50%	00.00%
5	11.00%	04.00%	05.00%	02.00%	00.00%	65.30%	00.00%	00.00%	12.70%	00.00%
6	12.00%	03.00%	05.50%	02.00%	03.50%	07.00%	61.66%	00.00%	05.34%	00.00%
7	00.00%	05.50%	00.00%	06.00%	00.00%	03.50%	00.00%	80.00%	00.00%	05.00%
8	06.00%	00.00%	04.00%	07.00%	06.00%	03.00%	03.00%	00.00%	66.67%	04.33%
9	00.00%	03.50%	00.00%	00.00%	02.00%	15.00%	00.00%	06.67%	14.50%	58.33%

Then, the matrix of confusion that is associated to SVM is presented in following table:

Table 3. The matrix of confusion of SVM

SVM	0	1	2	3	4	5	6	7	8	9
0	90.00 %	00.00%	00.00%	02.50%	04.50%	03.00%	00.00%	00.00%	00.00%	00.00%
1	00.00%	100.0%	00.00%	00.00%	00.00%	00.00%	00.00%	00.00%	00.00%	00.00%
2	00.00%	00.00%	96.66%	00.00%	00.00%	02.34%	00.00%	00.00%	00.00%	00.00%
3	11.67%	00.00%	04.00%	77.33%	00.00%	06.00%	00.00%	01.50%	03.50%	00.00%
4	00.00%	02.67%	00.00%	04.00%	78.33%	05.00%	00.00%	0.00%	05.50%	04.50%
5	05.50%	00.00%	07.50%	10.00%	00.00%	70.00%	00.00%	03.00%	00.00%	04.00%
6	14.00%	00.00%	00.00%	02.70%	05.00%	00.00%	73.30%	00.00%	05.00%	00.00%
7	00.00%	02.50%	04.50%	00.00%	00.00%	00.00%	00.00%	81.66%	00.00%	11.34%
8	00.00%	00.00%	02.00%	00.00%	05.33%	00.00%	08.00%	00.00%	78.67%	06.00%
9	00.00%	00.00%	05.34%	00.00%	00.00%	08.50%	00.00%	09.50%	00.00%	76.66%

8 CONCLUSION

In this paper, we have presented two methods for recognition of isolated Latin handwritten numerals. Those methods are Pre-processed by a thresholding, centring and then a skeletisation operation is carried. The features extraction is used by the zoning method, the learning-classification was done using the self-organization maps and the support vectors machines.

The simulation results demonstrate that the SVM is more performing but more slow than the SOM in the recognition of handwritten Latin numeral.

REFERENCES

[1] Ravindra S. Hegadi, Recognition of Printed Kannada Numerals based on Zoning Method, International Journal of Computer Applications (0975 – 8878) on National Conference on Advanced Computing and Communications - NCACC, April 2012

[2] Basappa B. Kodada and Shivakumar K. M. Unconstrained Handwritten Kannada Numeral Recognition , International Journal of Information and Electronics Engineering, Vol. 3, No. 2, March 2013

[3] Efficient zone based feature extraction algorithm handwritten numeral recognition of four popular south Indian scripts, Journal of Theoretical and Applied Information Technology

[4] Rajashekararadhya S.V. and Vanaja Ranjan P., “Efficient zone based feature extraction algorithm for handwritten numeral recognition of four popular south Indian scripts”, Journal of Theoretical and Applied Information Technology, 2005, pp 1171- 1181.

[5] Sikkim Manipal ,Manjaiah D.H, Rabindranath Bera, Ashoka H.N. et al , Zone Based Feature Extraction and Statistical Classification Technique for Kannada Handwritten Numeral Recognition , International Journal of Computer Science & Engineering Technology (IJCSET).

[6] Neila Mezghani1 and Amar Mitiche2, A Gibbsian Kohonen Network for Online Arabic Character Recognition, G. Bebis et al. (Eds.): ISVC 2008, Part II, LNCS 5359, pp. 493–500, 2008.c Springer-Verlag Berlin Heidelberg 2008

[7] Dr. Pankaj Agarwal, hand-written character recognition using Kohonen Network, IJCSt Vol. 2, ISSue 3, September 2011

[8] R.Indra Gandhi, ADr.K.Iyakutti, n Attempt to Recognize Handwritten Tamil Character Using Kohonen SOM , Int. J. of Advance d Networking and Applications Volume: 01 Issue: 03 Pages: 188-192 (2009)

[9] Prof. Sheetal A. Nirve , Dr. U. B. Shinde, HINDI CHARECTER RECOGNITION USING KOHONEN NETWORK, International Journal of Scientific & Engineering Research, Volume 4, Issue 5, May-2013, ISSN 2229-5518

[10] S. Peyarajan R. Indra Gandhi, ON-LINE TAMIL HAND WRITTEN CHARACTER RECOGNITION USING KOHONEN NEURAL NETWORK, Research Journal of Computer Systems Engineering- An International Journal, Vol 02, Issue 02, July, 2011

[11] Arora, S., Bhattacharjee, D., Nasipuri, M., Malik, L., Kundu, M. and Basu, D. K., Performance Comparison of SVM and ANN for Handwritten Devnagari Character Recognition, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 6, May 2010.

[12] H Byun and S. Lee. “Applications of Support Vector Machines for Pattern Recognition: A Survey”, In the Proceedings of the First International Workshop: Pattern Recognition with Support Vector Machines, pps 213-236, Niagara Falls, Canada 2002.

[13] K. Vijay Kumar, R.Rajeshwara Rao, Online Handwritten Character Recognition for Telugu Language Using Support Vector Machines, International Journal of Engineering and Advanced Technology (IJEAT), ISSN: 2249 – 8958, Volume-3, Issue-2, December 2013

[14] Luiz S. Oliveira and Robert Sabourin, Support Vector Machines for Handwritten Numerical StringRecognition, Proceedings of the 9th Int’l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004) 0-7695-2187-8/04 $20.00 © 2004 IEEE

[16] Kohonen T," Self-Organizing Maps " 3rd edition, Springer, Berlin, 2001.

[17] V.N.Vapnik, "An overview of statistical learning theory", IEEE Trans. Neural Networks., vol. 10, pp. 988– 999, Sep. 1999.

[18] C. Cortes and V.Vapnik. Support vector networks. Machine Learning, 20: pp 1–25, 1995.