International Journal of Innovation and Scientific Research
ISSN 2351-8014 Vol. 5 No. 1 Jul. 2014, pp. 16-24

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

A Comparative Study Between the Hidden Markov Models and the Support Vector Machines for Noisy Printed Numerals Latin 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 methods of learning-classification, the first one is called Hidden Model Markov (HMM) which is based on a unsupervised learning, and the second one called Support Vector Machine (SVM) which is based on a supervised learning. Those techniques are used for printed Latin numerals recognition, in different situation: rotated, resized and noisy. In the pre-processing phase we use the thresholding technic and in the features extraction we use the Hu invariants moments (HIM). The simulation results demonstrate that SVM is more robust than the HMM technic in the printed Latin numerals recognition.

KEYWORDS: The noisy printed Latin numerals, the thresholding technique, the Hu invariant moments, the Hidden Model Markov, the Support Vector Machine

1 INTRODUCTION

The optical character recognition (OCR) is considered as a one of the most successful and powerful applications of the automatic pattern recognition. It's really a very active field of research and development.

Several studies have been carried on Latin, Arabic numerals and characters by using: the hidden models Markov [1-4], the support vectors machines [5-10] or moments [11-14].

However, our study is focused on Latin numerals recognition.

A succession of operations in this recognition system can be divided into three principal phases. The first is a pre-processing which serves to clean the numeral image just for improving its quality. The second phase is the features extraction from pattern for avoiding data abundance and reducing its dimension. The third phase is the learning-classification. During this phase the images of learning base that are converted to vectors in the second phase should be to train with a learning process. After the images of the test database must be classified.

In this study the pre-processing numerals is carried by the thresholding technique. In the phase of extraction of the primitives from numeral image the Hu invariants moments (HIM) [15] that are used to transform each image of numeral to a vector that will used as an input vector of HMM and of SVM which are used to train the images of the training database and then to classify those of the test database. The last phase takes place as follows:

- By using the HMMs:

In the learning phase, each numeral image is converted to a vector by calculating the HIM; this vector will be used as an observation vector of an initial own HMM of this numeral ~~for~~ to determine the probability which have generated this observation. Then this model must be re-estimated in order to ~~reason for~~ maximize this probability ~~by~~ using the Baum-Welch algorithm. All the re-estimated models (optimal models) of all numerals are saved ~~for~~ to form the learning base.

In the classification phase, we will present an unknown numeral (test numeral) translated, rotated or resized and influenced by noise. Those numerals are used ~~noisy~~ like a vector of observation by calculating its HIM, then we calculate the probability generated by this observation by all the optimal models already recorded in the learning base by the forward algorithm. The recognition will be given to the numeral with highest probability.

- By using the SVMs:

In the learning phase, we have used the SVM whose the strategy is one against all. This separation was made for each image which modeled by a class labeled by the value 1 of the learning base to the rest of all the other images that are modeled by ~~a~~ another class that has a label equal to -1.This separation (maximizing the margin between two classes) is therefore creating a decision function separating these two classes. We have 10 numerals. So we will have 10 decision functions each of them will separate a pair of classes (1 and -1) among the 10 pairs.

In the classification phase, we calculate the image of the vector that models the test numeral preprocessed translated, rotated or resized and containing a noise by 10 decision functions. The recognition will be assigned to the numeral whose decision function separates its class to another class containing the rest of all others numerals which gives the biggest value among all the values calculated of the 10 images of the numeral ~~of~~ test.

2 THE RCOGNITION SYSTEM

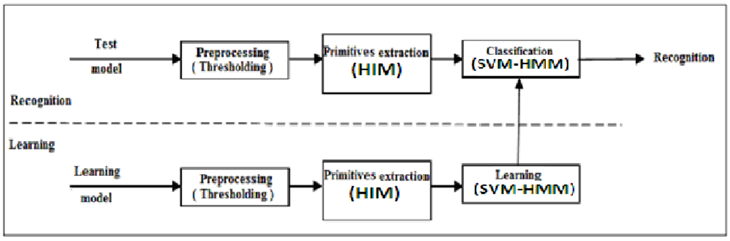

***Fig. 1. System for Latin numerals recognition*.**

3 PREPROCESSING

Pre-processing is the first part of Latin numerals recognition system. This is used to produce a cleaned up version of the original image so that it can be used efficiently by the feature extraction components of the OCR.

In our study, we preprocess the images by a thresholding technic; in order to construct the images containing only the black and ~~the~~ white colors according a preset threshold.

4 FEATURES EXTRACTION

The second phase of the Latin numeral recognition system is features extraction. Several methods can be used to compute the features. In this recognition system, we use the Hu Invariant moments (HIM)

Firstly, we recall the definition of the geometric moment of order $(p+q)$ of an image function $f(x,y)$ of a size NxN is:

$$m_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} x^p y^q f(x,y) \qquad (1)$$

These moments are not invariant to geometric transformations: translation, rotation and scaling. For to make it invariant to translation, we present the central moment of order $(p+q)$:

$$\mu_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} (x-\bar{x})^p (y-\bar{y})^q f(x,y) \qquad (2)$$

$\bar{x}$ and $\bar{y}$ are the coordinates of the center of gravity of the image calculated by:

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad , \quad \bar{y} = \frac{m_{01}}{m_{00}} \tag{3}$$

The centered normalized moment of order $(p+q)$ which is at a time invariant to translation and scaling is:

$$\eta_{pq} = \frac{\mu_{pq}}{m_{00}^{\gamma}} \quad , \quad \gamma = \frac{p+q}{2} + 1 \quad , \quad (p+q) \geq 2 \tag{4}$$

Hu was established seven moments following which are invariant to translation, rotation and scaling:

$$\varphi_1 = \eta_{20} + \eta_{02}$$

$$\varphi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\varphi_3 = (\eta_{30} - 3\eta_{12}) + (3\eta_{21} - \eta_{03})^2$$

$$\varphi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\varphi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) * [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03}) * [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

$$\varphi_6 = (\eta_{20} - \eta_{02}) * [(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\varphi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) * [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + 3(\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) * [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

5 LEARNING PHASE

5.1 THE HIDDEN MARKOV MODELS

The Hidden Markov model (HMM) [16] bases on a doubly stochastic processes whose one of them is hidden. The transition of the process from the actual state to the next is based on this underlying process. The observable outputs or the observations are generated by other stochastic process which is given by probabilities. The HMM with a discrete observation symbol is defined by $\lambda = (A, B, \pi)$, where A is the matrix of the probabilities of transitions, B is the matrix of the probabilities of observations, and π is the vector probability of initial states. N : The number of states $s_1, s_2 \ldots\ldots s_N$.

T : The number of observations.

q_t The state of the process at the time t ($q_t = \{s_1, s_2, \ldots. s_N\}$).

o_t The observation at the time t ($o_t = \{v_1, v_2, \ldots., v_M\}$).

M : The size of observations $v_1, v_2 \ldots\ldots v_M$.

$$A = \{a_{ij} = \Pr ob(s_j / s_i)\} \; ; \; \sum_{j=1}^{N} a_{ij} = 1 \tag{5}$$

$$\pi = \{\pi_i = \Pr ob(s_i)\} \; ; \; \sum_{i=1}^{N} \pi_i = 1 \tag{6}$$

$$B = \{b_j(k) = \Pr ob(o_t = v_k / o_t = s_j)\} \; \sum_{k=1}^{M} b_j(k) = 1 \tag{7}$$

* The hidden Markov model with a continuous observation symbol is defined by $\lambda = (A, \pi, \mu_i, \sigma_i)$ where μ_i and σ_i are respectively the mean and the standard deviation of the state i of the Gaussian function that used for to generate the probability of observation:

$$b_j(k) = \mathrm{Prob}(o_t = v_k / o_t = s_j) = \frac{1}{\sigma_i \sqrt{2\pi}} e^{\frac{(o_t - \mu_i)^2}{2\sigma_i^2}} \quad (8)$$

In our work we used the HMM with the continuous observation. Each numeral is converted to a vector O_t of 7 components calculated by the HIM and has a particular initial model.

6 THE SUPPORTS VECTORS MACHINES

6.1 PRINCIPE OF FUNCTIONING BETWEEN TWO CLASSES OF SVM

6.1.1 THE LINEAR CASE

For a set of vectors $x_i \in \Re^n$ with n is the dimension of the vector space and given a 2 classes. the first class containing a party of these vectors and bears a label equal to 1, the second class contains the other party of vectors labeled by the value -1.The goal of the SVM[17] is to determine a classifier that well separates these 2 classes and maximizes as much as possible the distance between the 2 classes. This classifier called hyperplan (see figure 2)

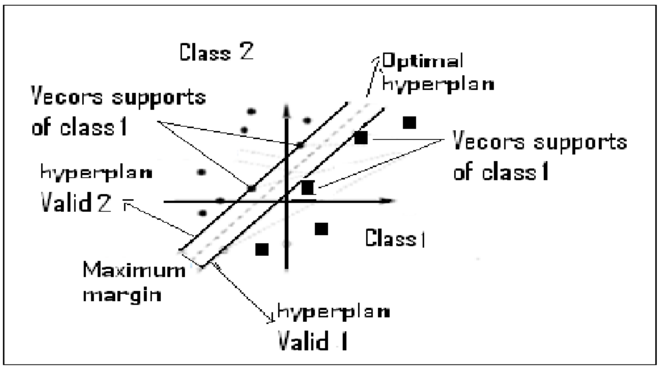

Fig. 2. The determination of optimal hyperplan, vectors supports, maximum Marge and valid hyperplans.

The nearest points that only are used for the determination of hyperplan are called the support vectors. The property of SVM is that this hyperplan must be optimal, that is to say it must maximize the distance between support vectors of a class and those of other. The classifier is represented by the function:

$$f(x, w, b) : x \rightarrow y \quad (9)$$

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

6.1.2 THE PRIMAL/DUAL PROBLEMS

6.1.2.1 THE PRIMAL PROBLEM

For to maxim the distance between the supports vectors of a class and those of other class, we must to solve a problem of a minimization under some constraints called the primal problem:

- To minimize

$$P(w, b) = \frac{1}{2} \|w\|^2 \quad (10)$$

- Such that

$$y_i(wx_i + b) \geq 1$$

6.1.2.2 The dual problem

For to simplify the calculations, it's necessary to introduce a formulation called problem dual by using the Lagrangian operator:

$$L(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i [y_i(wx_i + b) - 1] \tag{11}$$

The dual variables α_i intervening in the Lagrangian is called Lagrange multipliers. The dual problem is:

- To maximize

$$D(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{i=1}^{n}\alpha_i\alpha_j y_i y_j x_i x_j \tag{12}$$

- Such that

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

$$\alpha_i \geq 0 \ , \forall i = 1,2,\ldots,n$$

Only the α_i^* corresponding to the points which are nearest to hyperplane is nonzero, we speak of the support vectors. The decision function associated to this separation is:

$$f(x) = \sum_{i=1}^{n} \alpha_i^* y_i x_i . x + b \tag{13}$$

6.1.3 THE NON LINEAR CASE

In the linear case(see figure 2), the classification of the data is easy, but in the nonlinear case the optimal separation between the two classes is carried by virtue an special type of functions called the kernel functions:

$$K : \Re^n x \Re^n \rightarrow \Re^p \qquad p \succ n \tag{14}$$

$$(x_i, x_j) \rightarrow K(x_i, x_j)$$

We must solve therefore:

- To maximize

$$D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

- Such that

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{15}$$

$$0 \leq \alpha_i \leq C \ , \forall i = 1,2,\ldots\ldots, \ n$$

The parameter C which appears here is a positive constant fixed in advance; it's called the constant of penalty. The decision function has the form:

$$f(x) = \sum_{i=1}^{n} \alpha_i^* y_i K(x, x_i.) + b \tag{16}$$

Some examples of the kernel functions:

Table 1. Examples of kernel functions

Kernel linear	xy
Kernel polynomial of degree n	$(axy + b)^n$
Gaussian radial basis function (GRBF):	$e^{-\frac{\lVert x-y \rVert^2}{2\sigma^2}}$

6.1.4 PRINCIPE OF FUNCTIONING BETWEEN A SEVERAL CLASSES OF THE SVMS

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 of classes [18], 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 a output of SVM is the largest.

$$i = \arg \max_{i=1,2,\ldots,N} (f_i(X)) \quad (17)$$

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. In our study, we used the kernel function GRBF with the standard deviation σ = 1 and a penalty constant. C = 10^4

7 THE CLASSIFICATION PHASE

- By using the HMMs :

We present an unknown numeral (test numeral) translated rotated or resized and noisy as a vector of observation by calculating its HIM, then we will calculate the probability generated by this observation by all optimal models already saved in the learning base by the Forward algorithm, the recognition will be assigned to the numeral that the optimal model which gave the biggest probability.

- By using the SVMs :

After having built the10 decision functions between the 10 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 10decision 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 10 images of the numeral test.

8 EXPEREMENT RESULTS

We choose the sizes of all images 30x30.Each numeral was converted to a vector of 7 components which is the HIM values For to take all the image size, we must varying x and y into[-1 1](as a case of Legendre moment)instead of [0 30] Therefore we will to perform the following change of variable:

$$u = \frac{x - \frac{30}{2}}{\frac{30}{2}}, \quad v = \frac{y - \frac{30}{2}}{\frac{30}{2}} \quad (18)$$

The Hu moments values is very small, so we have used log (φ_i) , i = 1,2…7. First we present a test numeral translated, rotated or resized and not noisy, then we add increasingly a quantity of noise of type 'salt & pepper' for to know the effect of noise added on the rate recognition of each numeral. The noise values are:

[0,0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09,0.1,0.11,0.12,0.13,.14,0.15,0.16,017,0.18,0.19,0.2,0.21,0.22,0.23,0.24,0.25, 0.26,0.27,0.28,0.29,0.30].

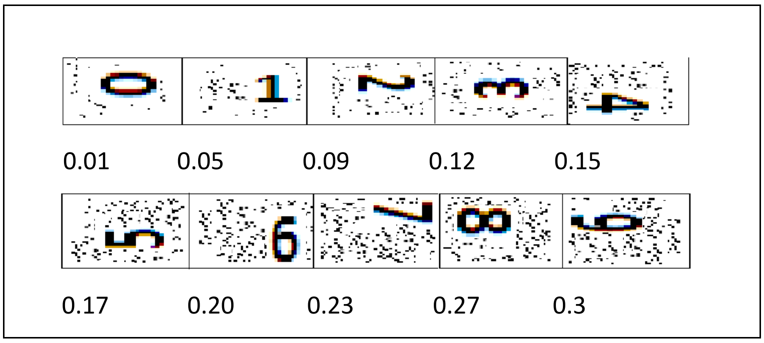

Fig. 3. The Latin numerals noisy by different values of noise of type 'salt & pepper'.

We used 30 numerals in the learning base and 990 in the test base.

We group the values of the recognition rate τ_n for each numeral that we obtained in the following table

Table 1. The recognition rate for each numeral.

Numeral	τ_n (HMM)	τ_n (SVM)
0	100%	100%
1	100%	100%
2	96,77%	100%
3	32,26%	96,77%
4	100%	100%
5	87,10%	93,55%
6	100%	100%
7	100%	100%
8	100%	100%
9	100%	100%

We present the evolution of the global recognition rate τ_g in function of noise added to numerals:

Table 2. The global recognition rate in function of noise added of HMM and SVM.

Noise	τ_g (HMM)	τ_g (SVM)
0.00	100.00%	100%
0.01	100%	100%
0.02	100%	100%
0.03	100%	100%
0.04	100%	100%
0.05	100%	100%
0.06	100%	100%
0.07	100%	100%
0.08	96,67%	100%
0.09	93,34%	100%
0.10	90%	100%
0.11	90%	100%
0.12	90%	100%
0.13	90%	100%
0.14	90%	100%
0.15	90%	100%
0.16	90%	100%
0.17	90%	100%
0.18	90%	100%
0.19	90%	100%
0.20	90%	100%
0.21	90%	100%
0.22	90%	100%
0.23	90%	100%
0.24	90%	100%
0.25	86,67%	100%
0.26	83,34%	96,67%
0.27	80%	93,34%
0.28	76,67%	90%
0.29	73,34%	80%
0.30	73,34	80%

And the graph associated:

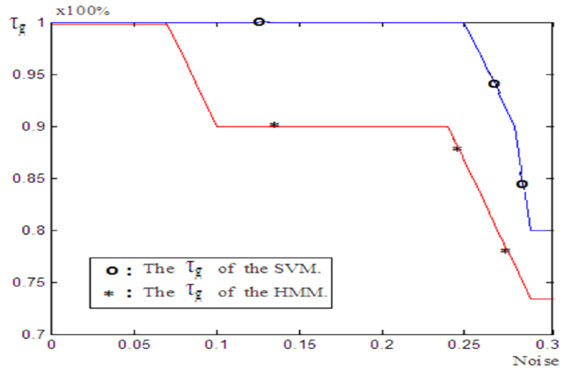

Fig. 4. The variation global recognition rate in function of noise added of HMM and SVM

Explication of obtained results:

The global rate recognition τ_g is a decreasing in function of noise added to numerals, but the important remark is that the falling of this rate of HMM is greater than the rate of SVM, this shows that the SVM is more performing than the HMM in recognition of noisy numerals.

9 CONCLUSION

The results obtained in the recognition of noisy Latin numerals show that reliable recognition is possible by using the thresholding technique in the preprocessing phase and the HIM in the primitives extraction phase. The simulation results demonstrate that the SVM method is more precise than the HMM technic in the recognition of printed Latin numeral.

REFERENCES

[1] M.Pechwitz and V. Maergner, “HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT-Database”, Proceedings of the Seventh International Conference on Document Analysis and Recognition - Volume 2, IEEE Computer Society, 2003, p. 890.

[2] Sabri A. Mahmoud and Sameh M. Awaida, “recognition of off-line handwritten Arabic (Indian) numerals using multi-scale features and supports vectors machines vs. hidden Markov models”, The Arabian Journal for Science and Engineering, Volume 34, Number 2B October 2009

[3] M. Mohamed, P. Gader, Handwritten word recognition using segmentation-free hidden Markov modeling and segmentation-based dynamic programming techniques, IEEE Trans. Pattern Anal. Mach. Intell. 18 (5) (May 1996) 548–554.

[4] A.H. Hassin, X. Tang, J. Liu, W. Zhao, Printed Arabic character recognition using HMM, J. Comput. Sci. Technol.19 (4), July 2004, pp 538–543.

[5] Sabri A. Mahmoud and Sameh M. Awaida,recognition of off-line handwritten Arabic (Indian) numerals using multi-scale features and supports vectors machines vs. hidden Markov models, The Arabian Journal for Science and Engineering, Volume 34, Number 2B October 2009

[6] 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.

[7] Ahmad, A. R.,Viard-Gaudin, C., Khalid, M. and Yusof, R., Online Handwriting Recognition using Support Vector Machine,Proceedings of the Second International Conference on Artificial Intelligence in Engineering & Technology, Kota Kinabalu, Sabah, Malaysia, August 3-5 2004.

[8] Gita Sinha, Dr. Jitendra kumar, Arabic numeral recognition using SVM classifier, International Journal of Emerging Research in Management &Technology ISSN: 2278-9359 (Volume-2, Issue-5), May 2013.

[9] Thiago C.Mota and Antonio C.G.Thomé, One-Against-All-Based Multiclass SVM Strategies Applied to Vehicle Plate Character Recognition, IJCNN, 2009.

[10] J. Sadri, C. Y. Suen, and T. D. Bui, “Application of Support Vector Machines for recognition of handwritten Arabic/Persian digits,” in Second Conference on Machine Vision and Image Processing & Applications (MVIP 2003), 2003, pp. 300–307.

[11] Ibrahim A. El rube, Mohamed T. El Sonni and Soha S. Saleh, “printed Arabic sub-word recognition using moments”, World Academy of Science, Engineering and Technology 42 2010

[12] Gheith Abandah, Nasser Anssari: novel moment features Extraction for Recognizing Handwritten Arabic Letters”, Journal of Computer Science 5 (3): 226-232, 2009, ISSN 1549-3636.

[13] R.Muralidharan1, C. Chandrasekar: Object Recognition using SVM -KNN based on Geometric Moment Invariant, International Journal of Computer Trends and Technology- July to Aug Issue 2011, ISSN: 2231-2803, Page 215-220.

[14] P.Nagabhushan ,S. A. Angadi ,B .S. Anami, a fuzzy statistical approach of Kannada vowel recognition based on invariant moments, NCDAR-2003, Mandy, India , pp275-285, 2003

[15] M. K. Hu, ‘’Visuel pattern recognition by moment invariants’’ , IRE Transactions On Information Theory, 1962, pp. 179-187.

[16] Przemyslaw Dymarski ,’’ Hidden Markov models, theory and applications’’, InTech Janeza Trdine 9, 51000 Rijeka, Croatia Copyright © 2011 InTech

[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.