Offline Handwritten Characters Recognition Using Moments Features and Neural Networks

Mohamed Abaynarh and Lahbib Zenkouar

Laboratory of Electronics and Communication, Faculty and Name of Institute, Ecole Mohammadia d'Ingenieurs, Avenue Ibn sina B.P. 765 Agdal Rabat, 10000, Morocco

Abstract: In this paper we revise the moment theory for pattern recognition designed, to extract patterns from the noisy character datas, and develop unconstrained handwritten. Amazigh character recognition method based upon orthogonal moments and neural networks classifier. We argue that, given the natural flexibility of neural network models and the extent of parallel processing that they allow, our algorithm is a step forward in character recognition. More importantly, following the approach proposed, we apply our system to two different databases, to examine the ability to recognize patterns under noise. We discover overwhelming support for different style of writing. Moreover, this basic conclusion appears to remain valid across different levels of smoothing and insensitive to the nuances of character patterns. Experiments tested the effect of set size on recognition accuracy which can reach 97.46%. The novelty of the proposed method is independence of size, slant, orientation, and translation. The performance of the proposed method is experimentally evaluated and the promising results and findings are presented. Our method is compared to K-NN (k-nearest neighbors) classifier algorithm; results show performances of our method.

Key words: Neural network, character recognition, orthogonal moments, pattern recognition.

1. Introduction

Handwritten character recognition is one of the most challenging topics in pattern recognition; this field of research is applied in various areas that aim at reducing human efforts like postal automation [1, 2], bank automation [3], form filling etc. Handwritten character recognition for Amazigh scripts [4] is quite a challenging task due to several reasons. The large category set, wide variability of writing styles, and the confusion between similar characters. Further, handwritten characters tend to show a large variation in the basic shape of the characters since the pen ink, pen width, accuracy of acquisition device, the stroke size and location in the character, physical and mental situation of the writer affect the writing style and in turn the recognition accuracy. Research in offline character recognition started with the recognition of

printed characters, and then extended to the recognition of handwritten numbers and characters in many languages scripts [3, 5-10].

ANNs (artificial neural networks) and HMMs (hidden markov models) are the most used, amongst the techniques which have been investigated for handwriting recognition. It has been observed that ANNs in general obtained best results than HMMs [11]. The most widely studied and used artificial neural network is the MLP (multi-layer perceptron) [12]. Such an architecture trained with back-propagation [13] is among the most popular and versatile forms of neural network classifiers and have shown its power and good performance for pattern recognition. However, the performance of those classifiers is strongly affected by the quality of the representation of the pattern i.e. features. Consequently, we present an efficient feature extraction method based on orthogonal Legendre moments which is rotation, scale and translation invariant.

Corresponding author: Mohamed Abaynarh, Ph.D. student, research fields: signal processing, pattern recognition, computer vision, image processing. E-mail: mohamed.abaynarh@gmail.com.

Hu [4] first introduced the use of seven invariants moments which are defined on moments of the image as features for pattern classification, after much kind of moments are used in literature. In this work we use Legendre moments, which are nonlinear and invariant under translation, scaling, and image reversal. If lower orders of moments are not enough to classify patterns, higher orders will be used, although, the higher orders resulted in higher sensitivity.

Indeed we suggest the maximum entropy principle MEP (maximum entropy principle) [14] as feature selection criterion which produces finite optimal moment orders carrying out only moments of low orders containing sufficient and pertinent information needed for classification.

Our classification method is used with different network topologies, compared to other methods as nearest neighbor.

In general, the overall recognition process can be divided into 3 main sections (Fig. 1), named segmentation, preprocessing, and classification. Segmentation requires isolating the characters individually before they are fed to the preprocessing unit where the important features of characters (feature extraction) are identified. Finally, classification process is done by determining the category or the group of each character used during the recognition process.

Experimental results show that the proposed method reduces the computational burden of the recognition system in terms of the total number of layers and nodes, while showing improved performances in terms of recognition rate and generalization ability.

The paper is organized as follows: In the coming section, we describe briefly the database used in our system. Section 3 points out the proposed method of moment features extraction. Section 4 explains neural classifier. Section 5 is devoted to experimental results. Finally, Section 6 draws conclusion and summarizes the paper.

Fig. 1 Block diagram of the system.

2. Database Preparation

2.1 Motivation

Amazigh is the language of a population called “imazighen”, it is spoken by 50% of population in Morocco, in Egypt 27% of persons use it, 25% in Algeria, from 5% to 10% in Tunisia, Niger and Mali [15].

In Morocco while spoken Amazigh varies across regions, written Amazigh, has standardized version called Tifinaghe, which is adopted by IRCAM (“Institut Royal de la Culture Amazigh”) in Fig. 2, used for official communication and has been integrated in mass media and schools. The characters of Amazigh script are used by a much higher percentage of north Africa’s population. Thus, the ability to automate the interpretation of written Amazigh would have widespread benefits.

Amazigh handwriting recognition can also enable the automatic reading of ancient Amazigh manuscripts. Since written Amazigh has changed little over time, and the variations between regions are minor, the same techniques developed for Tifinaghe can be applied to many manuscripts, so automatic processing can greatly increase the availability of their content.

Fig. 2 Amazigh alphabet (Tifinaghe IRCAM).

Fig. 3 Printed pattern (a) and handwritten forms (b, c, d) of the character “YAZ” of different styles of writing.

The recognition task seems simple, because the writing in manuscripts is usually neater than free handwriting. However, image degradation, unexpected markings, and previously unseen writing styles provide many challenges.

2.2 Amazigh Writing

The Amazigh alphabet contains 33 characters, but Unicode codes only 31 letters plus a modifier letter to form the two phonetic units [16], written horizontally from left to right, and letters within a word are not joined. There is no connection between separate words, so word boundaries are always represented by a space.

2.3 Database Construction

Fifteen to twenty years ago, large databases were

developed for the recognition of handwriting in Latin scripts, and recently for Arabic ones, in this work we'll use a new database for Amazigh scripts developed in our laboratory. Descriptions of these components of the present database are given below.

2.3.1 Acquisition

Before analyzing the different processing steps, we should mention that we are especially interested at the off line processing. For our case, the acquisition is made with a numeric scanner of resolution 300 dpi, images are stored as grayscale BMP (Bitmap) images of 128×128 size, with 8 bits/pixels, the used samples are all possible classes of the handwritten Amazigh alphabet with variable sizes and variable thickness. The Fig. 4 shows some samples of the used database.

2.3.2 Preprocessing and Normalization

The handwritten character data samples were acquired from various persons both male and female of different age groups. Their handwriting was sampled on A4 size paper. They were scanned using flat-bed scanner at a resolution of 300 dpi and stored as 8-bit grey level images.

Since handwriting exhibits variation in slope, stretch, skew, relative size, and letter appearance, as shown in Fig. 3, we used a normalization process, which normalized the character into 128×128 pixel image.

Our system will be evaluated on two databases the first is composed with spaced discrete characters, the second with boxed text.

The first database contains forms of unconstrained handwritten characters including 7,524 isolated characters, gathered from 57 different and independent writers. The whole set of available isolated characters datas have been split into a training dataset consisting of 6,600 samples per character taken randomly for training the classifiers, and a testing set consisting of 924 samples (Fig. 4).

The second one contains a dataset of text of months and days of the year, with a total of 3 examples overall and 924 characters. Training set with 726 characters and testing set contains 198 characters (Fig. 4).

Before collection of datas, the following points were decided to make the database as much representative as possible. Common factors responsible for variations in handwriting styles include age, sex, education, profession, writing instrument, writing surface. No restriction was imposed on the writers except for specifying rectangular regions for writing isolated characters of different sizes for boxed spaced characters. Since such rectangular regions are large enough, the restriction may not be considered as a serious one.

Samples of isolated characters from the present database are shown in Fig. 3. In some cases a character may touch or cross the horizontal or vertical lines of the bounding box. Therefore two types of errors may happen. In the major error case, some character's dots or some complementary strokes of it were omitted and the result was not distinguishable, but in the minor error case, usually the last part of the

character was missed.

The characters prepared as explained in previous section, are scanned using a scanner and these characters will be segregated according to their own character group. One example is shown below in Fig. 4.

3. Features Extraction

Features extraction is an important step in achieving high performance for an OCR (optical character recognition) system. Despite, the remaining steps which are not independent also need to be optimized to obtain the best possible performance. The selection of image features and corresponding extraction method limits or performs well the nature and output of the image-preprocessing step and the decision to use gray-scale versus binary image, filled representation or contour, thinned skeletons versus full-stroke images depends on the nature of the features

The figure displays two examples of handwritten forms. The left form is a registration card with fields for 'Responsable', 'Numéro', 'FRANCAIS', 'AMAZIGH', and a table of months and days with corresponding handwritten characters. The right form is a sample of isolated handwritten characters arranged in a grid.

FRANCAIS	AMAZIGH	Echantillon 1	Echantillon 2	Echantillon 3
Janvier	ⵏⵉⵙⵓⵏⵉ	ⵏⵉⵙⵓⵏⵉ	ⵏⵉⵙⵓⵏⵉ	ⵏⵉⵙⵓⵏⵉ
Février	ⵉⵎⵓⵏⵉ	ⵉⵎⵓⵏⵉ	ⵉⵎⵓⵏⵉ	ⵉⵎⵓⵏⵉ
Mars	ⵎⵓⵔⵉ	ⵎⵓⵔⵉ	ⵎⵓⵔⵉ	ⵎⵓⵔⵉ
Avril	ⵏⵉⵙⵓⵏⵉ	ⵏⵉⵙⵓⵏⵉ	ⵏⵉⵙⵓⵏⵉ	ⵏⵉⵙⵓⵏⵉ
Mai	ⵎⵓⵔⵉ	ⵎⵓⵔⵉ	ⵎⵓⵔⵉ	ⵎⵓⵔⵉ
Juin	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Juillet	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Août	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Septembre	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Octobre	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Novembre	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Décembre	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Lundi	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Mardi	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Mercredi	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Jeudi	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Vendredi	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Samedi	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ
Dimanche	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ	ⵙⵓⵏⵉ

Below the table, there are fields for 'Age', 'Profession', 'Nom', and 'Ville'. The 'Age' field has options: <20, 21-30, 31-40, >40. The 'Profession' field has options: Etudiant/élève, Enseignant, Administratif, Autre. The 'Nom' field contains 'Khalil' and the 'Ville' field contains 'Sofron'.

The right form shows a grid of isolated handwritten characters, including letters, numbers, and symbols, arranged in a grid.

Fig. 4 Samples of forms of the two databases.

to be extracted. Moreover, the format of the extracted features must match the requirements of the classifier [17].

Features extraction has been a topic of intensive research and we can find a large number of features extraction methods in the literature, but the real problem for a given application, is not only to select different features but which features extraction method is the best? This question led us to characterize the good features which should satisfy the following conditions:

(a) Robust to transformations—the image features should be as invariant as possible to image transformations including translation, rotation, and scaling, etc.;

(b) Robust to noise—the image features should be robust to noises and various degraded situations;

(c) Feature extraction efficiency—image features can be computed efficiently;

(d) Feature matching efficiency—the matching algorithms should only require a reasonable computational cost.

In this paper, we especially propose a method which can be used to binary and gray level images both, which is reasonably invariant with respect to shape variations caused by various writing styles. The major advantage of this approach stems from its robustness to small variation, ease of implementation and provides good recognition rate. Moments based feature extraction method provide good result even when certain preprocessing steps like filtering, smoothing and slant removing are not considered. Especially, the advantages of considering orthogonal moments are that they are shift, and scale invariants and are very robust in the presence of noise [18, 19]. The invariant properties of moments are utilized as pattern sensitive features in classification and recognition [20, 21].

3.1 Legendre Moments

Statistical moments represent average values of

processes (powered to order n) when a random variable is involved. Here, the original images were considered as two dimensional arrays of a random variable of dimension $N \times N$. The random variables took values from level 0 to 255, as the images were considered in gray levels quantized in 8 bytes

(Gray levels were obtained from BMP format). Moments were calculated for the random variable X , which was identified with the image block. In addition, X is a matrix of two coordinates (x, y) obtained from the image matrix $f(x, y)$. The definition of $(p + q)$ order invariant moment around the origin is given by:

The Legendre moments of order $(p + q)$ are defined for a given real image intensity function $f(x, y)$ as

$$\lambda_{p,q} = \frac{(2p+1)(2q+1)}{4} \int_R \int_R P_p(x) P_q(y) f(x, y) dx dy \quad (1)$$

where $f(x, y)$ is assumed to have bounded support

The Legendre polynomials $P_p(x)$ are a complete orthogonal basis set on the interval $[-1, 1]$ for an order p they are defined as

$$p_p(x) = \frac{1}{2^p p!} \frac{d^p}{dx^p} (x^2 - 1)^p \quad (2)$$

If only Legendre moments of order smaller than or equal to θ are given, then the function $f(x, y)$ can be approximated by a continuous function which is a truncated series:

$$f_\theta(x, y) = \sum_{p=0}^{\theta} \sum_{q=0}^{\theta} \lambda_{p-q,q} P_{p-q}(x) P_q(y) \quad (3)$$

The number of moments used in the reconstruction of image for a given θ is defined by

$$N_{total} = \frac{(\theta+1)(\theta+2)}{2} \quad (4)$$

In this paper, we determine the order of the truncated expansion of $f_\theta(x, y)$ which provides a good quality of the reconstructed object. The moments used in this reconstruction process will constitute the optimal subset for representing this object. Then, we introduce the MEP (maximum entropy principle) to extract relevant moments that uniquely represent the patterns [14, 22-23]. By applying the Maximum Entropy Principle the entropy function monotonically

increases up to a certain optimal order where sufficient image information is recreated and then becomes relatively constant [23].

3.2 Optimal Order Moments Selection Using MEP

In order to determine the expansion order, which gives a good quality of the estimated input image, we introduce the MEP for the search of this optimal order, this automatic technique can estimate the optimal number of moments directly from the available data and does not require any a priori image information specially for noisy images.

Let G_w be a set of estimated underlying probability density function for various Legendre moment orders θ :

$$G_w = \{\hat{p}_\theta / \theta = 1 \dots \omega\} \quad (5)$$

By applying the maximum entropy principle for noisy images, we deduce that among these estimates of the probability density function, there is one and only one probability density function denoted $\hat{p}_\theta^*(x_i, y_j)$ whose entropy is the maximum, and which represent the optimal probability density function, and then gives the optimal order of moments.

The Shannon entropy of $\hat{p}_\theta^*(x_i, y_j)$ is defined as:

$$S(\hat{p}_\theta) = -\sum_{x_i, y_j \in \Omega} \hat{p}_\theta(x_i, y_j) \log(\hat{p}_\theta(x_i, y_j)) \quad (6)$$

and the optimal $\hat{p}_\theta^*$ is such that

$$S(\hat{p}_\theta^*) = \max\{s(\hat{p}_\theta) / \hat{p}_\theta \in G_w\} \quad (7)$$

The process of determining the optimal order θ consists in estimating the p.d.f. for different orders and selecting the optimal one as the one for which the entropy reaches maximum. The following is basic algorithm which consists in an exhaustive search to determine the optimal order which maximizes $S(\hat{p}_\theta^*)$:

Initialize θ

Compute the p.d.f. $\hat{p}_\theta$

If $\hat{p}_\theta$ is maximum, then θ is optimal and $\hat{p}_\theta = \hat{p}_\theta^*$, else $\theta = \theta + 1$ and go to 2. Then having $\hat{p}_\theta^*$, we assign to each point of the optimal p.d.f. $\hat{p}_\theta^*(x_i, y_j)$ defined by (5). In this case, the “good data” are the set of points belonging to the mode of $\hat{p}_\theta^*$.

4. Classification and Recognition

Recognition of handwritten characters is a very complex problem. The characters could be written in different size, orientation, thickness, format and dimension. These will give infinity variations. The capability of neural network to generalize and be insensitive to the missing data would be very beneficial in recognizing handwritten characters, also it's a fast, parallel and compact approach for processing the extracted features of the isolated characters.

The proposed Amazigh handwritten character recognition system uses a neural network approach to recognize the characters, based on feed forward MLP network with one hidden layer trained using back-propagation algorithm. Our method is compared to KNN algorithm

4.1 K-Nearest Neighbor

The nearest NN (neighbor rule) is described in several places in the literature [24, 25]. It's known as a conventional nonparametric statistical classifier. In training stage, it stores all training samples in a table. Then in testing stage, it assigns an unknown input pattern to which class has minimum distance to a training sample of that class. Just such as minimum mean distance classifier.

For background a simple statement will be included here. First, assume there are M pattern classes, numbered $1, \dots, M$. Let each pattern be defined in an N -dimensional feature space and let there be K training patterns. Each training pattern is a pair (x^i, θ_i) , $1 \leq i \leq K$, where $\theta_i \in \{1, 2, \dots, M\}$ denotes the correct pattern class and

$$x^i = (x_1^i, x_2^i, \dots, x_N^i) \quad (8)$$

is the set of feature values for the pattern. Let $T_{NN} = \{(x^1, \theta_1), (x^2, \theta_2), \dots, (x^K, \theta_K)\}$ be the nearest neighbor training set. Given an unknown pattern x , the decision rule is to decide x is in class θ_j , if

$$d(x, x^j) \leq d(x, x^i) \quad 1 \leq i \leq K, \quad (9)$$

where $d(.,.)$ is some N-dimensional distance metric. Actually, the preceding rule is more properly called the 1-NN rule, since it uses only one nearest neighbor. An obvious generalization of this is the k-NN rule, which takes the k nearest patterns $i_1, i_2, \dots, i_k$ and decides upon the patterns class that appears most frequently in the set $\theta_{i_1}, \theta_{i_2}, \dots, \theta_{i_k}$.

The similarity between two patterns x and y is given by:

$$S_{xy} = \sum_{i=1}^K (x_i - y_i)^2, (x_i, y_i, i = 1, \dots, K) \quad (10)$$

where x_i is the i^{th} feature of the pattern x and K is the total number of features.

The classification is carried using K-NN as follows:

Algorithm

Input : Isolated gray scale Amazigh character Images

Output: Recognition of the Character

Method: Legendre Moments, and K-NN Classifier

1. Fit the bounding box on an input image and crop the image, then resize it to 128×128 pixels;
2. Extract the moment based features stored in a feature vector;
3. A Euclidian distance criterion and K-NN classifier used to classify the test samples;
4. Stop.

4.2 Multi Layer Perceptron

Multi layer perceptron is a feed-forward neural network with one or more layers of nodes between the input and output layers, these in-between layers are called hidden layers, all neurons have a single sigmoid output. Each node in a layer is connected to the all nodes in the next layer. Using MLP in the context of a classifier requires all output nodes to be set to 0 except for the node that is marked to correspond to the class the input is from. That desired output is 1 and the weights are updated on the basis of a single sample.

We have designed a simple unconstrained character recognizer based on a MLP with one hidden layer. The number of hidden neurons was determined by a rule of thumb and some exploratory experiments

where the error rates on the training sets were used as criteria. Network output estimates a posteriori probabilities and the value of each output necessary remains between zero and one because of the sigmoid function used. The networks are trained and tested on different datas. Before training, a few of the weights—connection strengths, in each network were set to values that were expected a priori to be approximately correct, but the most majority of weights were independently set to random initial values, we train a number of neural network architectures with the same algorithm but with varying a number of hidden nodes and choose a model which produces the minimal training error with the character data. The inputs of the MFNN (multiple feedforward neural network) are feature vectors derived from the proposed feature extraction method described in the previous section. The number of nodes in the output layer is set to the number of Amazigh characters classes [26].

Experiments were conducted using the initial weight vectors that have been randomly chosen from a uniform distribution in $(-1, 1)$, this weight range has been used in Ref. [27, 28].

Structure of MLP network for Amazigh character recognition is shown in Fig. 3.

This is a flow diagram of the active nodes used in the hidden and output layers of the neural network. Each input is multiplied by a weight (the w_N values), and then summed. This produces a single value that is passed through sigmoid function.

The output of each node is a pondered sum of its inputs:

$$y_i = \varphi(a_i) = \varphi(\sum_{k=1}^N (w_{ik} x_k)) \quad (11)$$

With x_k the k^{th} composant of sample vector.

w_{ik} is the weight of the connection which rely unit k and unit i.

a_i is the activation of the unit i.

φ is the activation function of the units which is a threshold function with the following expression :

$$\varphi(x) = \begin{cases} -1, & x < \theta \\ +1, & x \geq \theta \end{cases} \quad (12)$$

5. Results and Discussion

Our procedure of handwritten Amazigh character recognition is given below:

Capture the scanned characters into 128x128 pixels;

Apply our proposed Feature Extraction method without any image preprocessing except normalization;

Implement the Neural Network Classifier with the subset already extracted;

Get the recognized character.

A complete flowchart of handwritten Amazigh character recognition is given below in Fig. 4.

There is not enough work on Handwritten Amazigh character recognition, and especially few works which are based neural networks. Recognition scheme proposed in Ref. [29], has an accuracy of 93.63% tested on a data set of 124 characters. The scheme proposed by B. El Kessab et al. [30] is a hybrid model MLP/HMM has an accuracy of 92.33%. They tested their scheme on 7,200 samples of Amazigh character.

Error rates on both sets usually go up and down simultaneously. However, if the neural network is trained many times, which more than the needed

information is provided, then the error rate on the training set continues to decrease will revert on the testing set. This situation is called over-training. The relationship between error rate on training and testing sets is shown in Fig. 7.

Experiments show the effect of set size on character recognition and accuracy, while the number of hidden nodes increases the number of epochs taken to recognize the handwritten character should also increase (Fig. 7).

Fig. 5 Multilayer perceptron with one hidden layer.

Fig. 6 System for Amazigh character recognition.

Fig. 7 Error rate of the first database of 924 samples.

Fig. 8 Training and testing rates of the ANN versus 2000 iterations.

First, accuracy is tested by exposing the network to a set of 924 characters, 726 characters for training set and 198 for testing set. Second, the system is evaluated with the second database of 7,524 characters

split into a training set consisting of 6,600 samples and a testing set consisting of 924 samples.

We should mention that the training and testing data were different; even more the data used for testing were outside training set. Back propagation algorithm is used for the training of the Neural Network. At the training time, weight and bias will be updated on each iteration, if there is a difference between the computed output and the target.

Table 1 indicates network results for different states. For MLP network with 100 to 700 neurons in middle layer and with equal iteration, you will observe different quantities for predicting precision, and we see that network with 400 neurons gives us response equal with 96 in test series, which is the most desirable answer than the others.

It is important to note here that the system performs extremely well with recognition rates ranging between 93% and 99% on different nodes and the overall recognition is 93.88%. This is a very good performance taking into account the fact that we have a limited number of samples in each class and that we have not used any noise filtering techniques. The recognition on the training data is also extremely high, 99.4% which represents very good training.

By analyzing the results, it shows that insufficient hidden nodes will cause under fitting where the network cannot recognize the characters because there are not sufficient adjustable weights to model the input-output relationship. Excessive hidden nodes will cause over fitting where the network fails to generalize. There is no theoretical development to

Table 1 Results of handwritten recognition accuracy using MLPN.

Input of the MLPN	No. of hidden units	No. of epochs	Recognition accuracy (%)	
			Training data	Testing data
231x231	100	2000	99.2	91.2
	200	2000	98.7	94.7
	300	2000	99.5	95.91
	400	2000	99.8	97.46
	500	2000	99.7	96.67
	600	2000	99.7	92.07
	700	2000	99.7	89.15
Average			99.47	93.88

Fig. 9 Error rate of the same architecture (one hidden layer) and different number of nodes

Table 2 Result of Recognition rate by number of epochs on the training and test set.

Number of hidden nodes	Number of epochs	Recognition rate	
		Training set	Test set
400	500	100	78
400	1000	100	91
400	1500	100	96,5
400	2000	100	97,46

Table 3 Recognition rate, error rate versus hidden layers.

Number of hidden layers	Recognition rate	Error rates
1	97.62%	2.38%
2	94.5%	5.5%
3	91.3%	8.7%

determine, the optimal number of the hidden layer nodes.

There are several rules of thumb for deciding the number of neurons in the hidden layer [27].

- The number of hidden neuron should be less than twice the input layer size.
- The number of hidden neuron should be in the range between the size of the input layer and the size of the output layer.
- Finding the minimum number of epochs taken to recognize a character and recognition efficiency of training as well as testing samples.
- The number of hidden neurons should be 2/3 of the input layer size, plus the size of the output layer.

A close inspection of the Table 3, shows that the recognition rate using one hidden layer is higher than

those obtained by two and three hidden layers.

The results show that increasing the number of hidden nodes, improves performance considerably, but over a number of nodes we have overtraining which decrease considerably performances.

The recognition converges faster when the number of samples is great, due to the very small number of training examples.

We believe it is because there is a great level of consistency in how a user draws shape (character). Of course, the more examples, the better is to train the recognizer.

6. Conclusions

An improved method of construction for handwritten character recognition has been presented. The Legendre moments features used for character recognition are shown to be effective for developing training and testing sets which have improved generalization capability without any preprocessing of characters images set.

Further improvements can be made by using more realistic training data and by modifying the hidden layers of the ANN to be sensitive to shifts of characters. The system showed good performance (97%) on a database of 7,524 handwritten Amazigh characters.

The results of structure analysis show that if the number of hidden nodes increases the number of epochs (iterations) taken to recognize the handwritten character also increases. A lot of efforts have been made to get higher accuracy but still there are tremendous scopes of improving recognition accuracy by developing new feature extraction techniques or modifying the existing feature extraction technique.

References

- [1] Roy, K., Vaidya, S., Pal, U., Chaudhuri, B. B., and Belaid, A. 2005. "A System for Indian Postal Automation." In *Proceedings of the 8th Int. Conf. Document Analysis and Recognition*, 1060-4.
- [2] Pal, U., Roy, R. K., Roy, K., and Kimura, F. 2009.

- "Indian Multi Script Full Pin-Code String Recognition for Postal Automation." In *Proceedings of the 10th Int. Conf. Document Analysis and Recognition*, 456-60.
- [3] Chaudhuri, B. B., and Pal, U. 1998. "A Complete Printed Bangla OCR system." *Pattern Recognition* 31: 531-49.
- [4] Hu, M. K. 1962. "Visual Pattern Recognition by Moment Invariant." *IRE Trans. Info. Theory* 8 (Feb): 179-87.
- [5] Pal, U., and Chaudhari, B. B. 2004. "Indian Script Character Recognition: A Survey." *Pattern Recognition* 37: 1887-99.
- [6] Rahman, A. F. R., Rahman, R., Fairhurst, M. 2002. "Recognition of Handwritten Bengali Characters: A Novel Approach." *Pattern Recognition* 35: 997-1006.
- [7] Bhattacharya, U., Ghosh, S. K., and Parui, S. K. 2007. "A Two Stage Recognition Scheme for Handwritten Tamil Characters." In *Proceedings of the 9th Int. Conf. Document Analysis and Recognition*, 511-15.
- [8] Jayadevan, R., Kolhe, S. R., Patil, P. M., and Pal, U. 2011. "Offline Recognition of Devanagari Script: A Survey." *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews* 41 (6): 782-96.
- [9] Dehghan, M. 1997. "Farsi Handwritten Character Recognition with Moment Invariants." In *Proceedings of the 13th International Conference on Digital Signal Processing*, 507-10.
- [10] Amin, A. 1998. "Off-Line Arabic Characters Recognition: The State Of the Art." *Pattern Recognition* 31 (5): 517-30.
- [11] Kapp, M. N., Reitas, C. F., and Sabourin, R. 2004. "Handwritten Brazilian Month Recognition: An Analysis of Two NN Architectures and a Rejection Mechanism." Presented at the 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR-9), Tokyo, Japan.
- [12] Bishop, M. 1995. *Neural Networks for Pattern Recognition*. Oxford, UK: Oxford Univ. Press.
- [13] LeCun, Y., Bottou, L., Orr, G. B., and Muller, K. R. 1998. "Efficient BackProp." In *Neural Networks: Tricks of the Trade*, edited by Orr, G., and Muller, K. Springer.
- [14] Qjidaa, H., and Redouane, L. 1999. "Robust Line Fitting in a Noisy Image by the Method of Moments." *IEEE Trans. on Pattern Analysis and Machine Intelligence* 21 (11): 1216-23.
- [15] Chaker, S. 2003. "The Berber" Acts of Languages of France. 215-27.
- [16] Outahajala, M., Zenkour, L., Rosso, P., and Martí, A. 2010. "Tagging Amazigh with AncoraPipe." In *Proceedings of the Workshop on LR & HLT for Semitic Languages, 7th International Conference on Language Resources and Evaluation, LREC-2010*, 52-6.
- [17] Fu, K. S. 1982. *Syntactic Pattern Recognition and Application*. Englewood Cliffs, N.J: Prentice Hall.
- [18] Mukundan, R. 2004. "Some Computational Aspects of Discrete Orthonormal Moments." *IEEE Trans. on Image Processing* 138 (Aug): 1055-9.
- [19] Haddadnia, J., Faez, K., and Moallem, P. 2001. "Neural Network Based Face Recognition with Moments Invariant." In *Proceedings of the IEEE International Conference on Image Processing*, 1018-21.
- [20] Tech, C. H., and Chin, R. T. 1988. "On Image Analysis by the Methods of Moments." *IEEE Trans. on Pattern Analysis and Machine Intelligence* 10 (4): 496-513.
- [21] Belkasim, S. O., Shridhar, M., and Ahmadi, M. 1991. "Pattern Recognition with Moment Invariants: A Comparative Study and New Results." *Pattern Recognition* 24 (12): 1117-38.
- [22] El Fadili, H., Zenkour, K., and Qjidaa, H. 2003. "Lapped Block Image Analysis via the Method of Legendre Moments." *EURASIP Journal on Applied Signal Processing* 9 (August): 902-13.
- [23] Zhunang, X., Haralick, R. M., and Zhao, Y. 1991. "Maximum Entropy Image Reconstruction." *IEEE Trans. Signal Processing* 39 (6): 1478-80.
- [24] Hart, P. E. 1966. *An Asymptotic Analysis, of the Nearest-Neighbor Decision Rule*. Tech. Rep. Stanford Electron. Lab., Stanford, Calif.
- [25] Cover, T. H., and Hart, P. E. 1967. "Nearest Neighbor Pattern Classification." *IEEE Trans Inform. Theory* 13 (January): 21-7.
- [26] Abaynarh, M., Elfadili, H., Zenkour, K., Zenkour, L. 2012. "Neural Network Classifiers for Off-line Optical Handwritten Amazigh Character Recognition." *International Journal of computer Science and Network Security* 12 (June): 28-36.
- [27] Hirose, Y., Yamashita, K., and Hijita, S. 1991. "Back-Propagation Algorithm Which Varies the Number of Hidden Units." *Neural Network* 4: 61-6.
- [28] Hoehfeld, M., Fahlman, S. E. 1992. "Learning with Limited Numerical Precision Using the Cascade-Correlation Algorithm." *IEEE Tran. On Neural Networks* 3: 602-11.
- [29] El Ayachi, R., Fakir, M., and Bouikhalene, B. 2011. "Recognition of Tifinaghe Characters Using Dynamic Programming & Neural Network." INTECH Open Access Publisher.
- [30] El Kessab, B., Daoui, C., Moro, K., Bouikhalene, B., and Fakir, M. 2011. "Recognition of Handwritten Tifinaghe Characters Using a Multilayer Neural Networks and Hidden Markov Model." *Global Journal of Computer Science and Technology* 11 (September): 12-20.