International Journal of Computer Applications (0975 – 8887)
Volume 75– No.18, August 2013

Handwritten Tifinagh Text Recognition using Neural Networks and Hidden Markov Models

Badre-eddine El Kessab
Laboratory of modeling and calculation
Faculty of Science and Technology, Beni Mellal, Morocco

Cherki Daoui
Laboratory of modeling and calculation
Faculty of Science and Technology, Beni Mellal, Morocco

Belaid Bouikhalene
Team Information Processing
Faculty of Science and Technology, Beni Mellal, Morocco

ABSTRACT
This paper deals with a recognition system of character for handwritten Tifinagh Text. Here in this work a neural network (the multi-layer perceptron MLP) and Hidden Markov Models (HMM) are proposed for handwritten characters identification. The features of Tifinagh characters are abstracted by mathematical morphology. Acquisition, scanning, thinning and text segmentation are also done in preprocessing phase before the classification with MLP and HMM. This work has achieved approximately 80% of success rate for Tifinagh handwritten text identification.

General Terms
Hidden Markov Model HMM, Neural Network NN, Baum-Welch algorithm.

Keywords
Recognition handwritten Tifinagh Text; Segmentation; Neural network; Hidden Markov Models, Viterbi algorithm, Baum-Welch algorithm; Mathematical morphology; Classification.

1. INTRODUCTION
The pattern of recognition system has become one of the major domain in which works more and more researchers. In this work the offline handwritten characters are identified with a recognition system based on the Artificial Neural Network (ANN) and Hidden Markov Model (HMM). Two learning approaches have been introduced, namely gradient descent algorithm for ANN and Baum–Welch algorithm for HMM. The neural network is a system of calculation widely used in the pattern recognition [1, 2, 3, 4 and 5]. There are a number of standard classification methods in use. The neural network methods are most widely known. It is then necessary to have a set of data representative of the problem studied (in our case these examples consist of several written characters by different people). Our project is to classify each character in a text encountered from a set of characters identifiable by the network. In this work, the system of recognition (Figure 1) is used and implemented on the handwritten Tifinagh characters base. It is composed of four phases: the first phase is the pretreatment, the second phase is the segmentation of characters, the third phase is the extraction of features, and the fourth phase is the classification. The classifiers used are neural networks (Multi Layers Perceptron MLP) and Hidden Markov Models (HMM). In the last model, we use a vector of extraction as a suite of observation and we seek to maximize the model with the best probability. The Baum-Welch algorithm is used for learning. This paper is organized in the following sections. Section 2 defines Tifinagh characters. Section 3 explains preprocessing phase. Section 4 is devoted to feature extraction and section 5 describes the suggested Multi layers perceptron. Section 6 defines the Hidden Markov Models and lastly in section 7 the results are explained then followed by conclusions.

The pattern recognition system is illustrated in the following figure (see Figure 1).

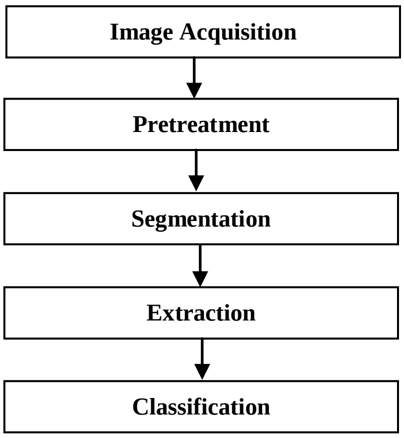


Fig 1: System of recognition

2. TIFINAGH CHARACTERS
The Tifinagh character is the Amazigh alphabet of North Africa. The number of Tifinagh characters by the Moroccan Royal Institute of Culture Amazigh (IRCAM) is 33 characters:

- Notable features
  - Type of writing system: alphabet.
  - Direction of writing: left to right in horizontal lines.
  - The characters are written in a way separated in the text (see Figure 2).

Fig 2: Tifinagh with Amazigh Alphabet

Fig 3: The Tifinagh Characters

3. RECOGNITION

The recognition text is intended to transform a written text in understandable representation by a machine and easily reproducible by a classifier. The words have an infinite number of representations that causing a problem, because each person produces their own written character. Our project is mainly interested on the recognition of handwritten characters.

3.1 Character writing

In pattern recognition system two modes of writing are used: static mode for previously written characters and dynamic mode for recognizing handwritten characters being written. In this project, we use the static mode. This mode uses the scanners. It analyzes the text line by line, offering the choice of different types of images binary, grayscale or color. For developing system to identifiy the character used in recognition the character pass by the classic steps of treatment, namely, the acquisition of the image, pretreatment, extraction, learning and classification.

3.2 Acquisition of image

After the image has been obtained, Acquisition method can be applied to the image to perform the tasks of treatment. Images acquisition begins with the writing of many examples on white paper, and with a scanner we get binary images in JPEG format. The obtained images have a size 64x24 with a marker and 24 x 24 with a pen.

Fig 4: Example of images acquired by a scanner with a marker

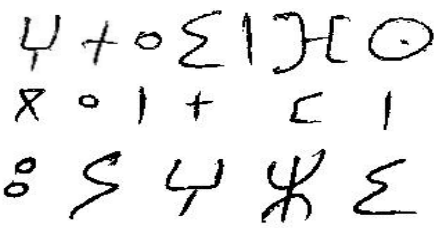


Fig 5: Example of images acquired by a scanner with a pen

3.3 Pretreatment

The pretreatment process is very important phase performed on information before the principal treatment in the recognition in order to improve their quality, or the extraction of information, the image of the character may undergo of any shortcomings. Should be corrected possibly these problems before the analysis step. The pretreatment techniques are schematically as follow:

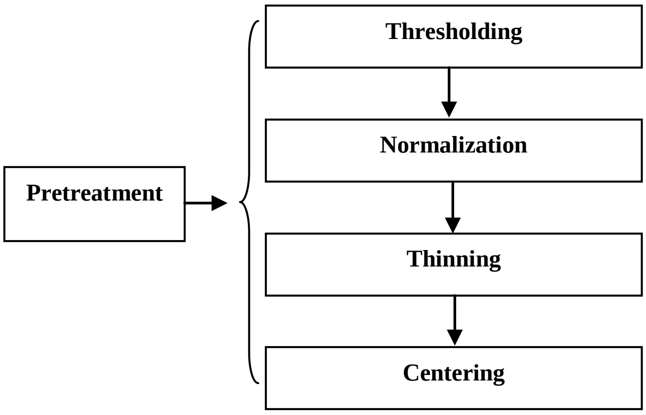


Fig 6: the pretreatment process

3.4 Learning

The learning processes are different and depending on characters form printed or written and following be it recognize one or more characters. In this step, we developed the system to learn the characters properties of the vocabulary used. The choice of the reference character by hand is a function of the application. The number of samples can vary from a few to some ten units by characters. This type of learning is called supervised learning.

4. SEGMENTATION OF CHARACTERS

The segmentation is a separation procedure of characters; its task is to analyze the raw image supplied to the computer by the digitalization peripheral to locate characters it contains. The segmentation is divided into two phases:

- The horizontal projection:
- The vertical projection:

4.1 The horizontal projection:

The horizontal projection consists a counting the number of white pixels in each line, then we trace the number of pixels found in each line a function of numbers of the lines (see Figure 7). Finally, we must isolate the lines of text from one another.

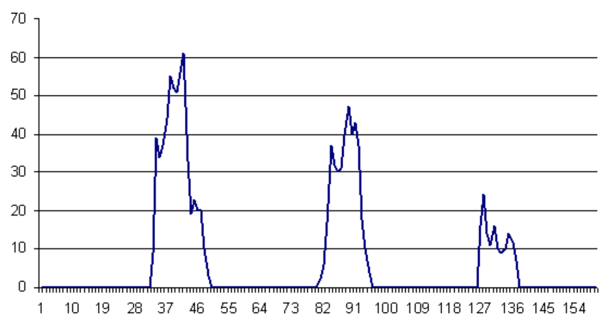


Fig 7: The horizontal projection histogram.

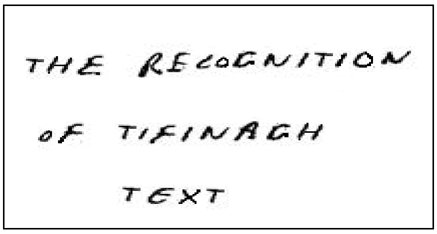


Fig 8: Text before the segmentation

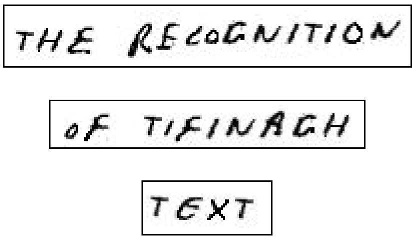


Fig 9: Text after the segmentation

4.2 The vertical projection:

The vertical projection consists a counting the number of white pixels in each column, then we trace the number of pixels found in each column a function of numbers of the columns (see Figure 10). After isolation of lines with the horizontal projection, we must separed the characters in each column.

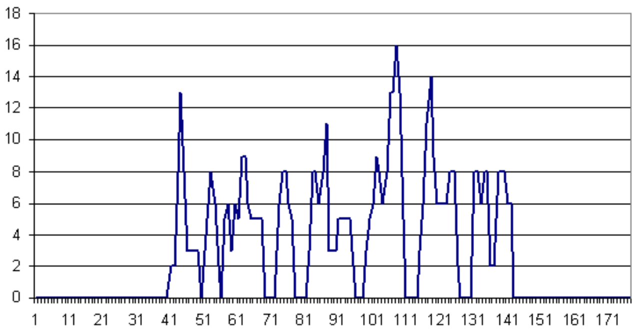


Fig 10: The vertical projection histogram.

Fig 11: Text before the segmentation

Fig 12: Text after the segmentation

4.3 The Overlap of Characters

Relatively frequent in practice, the overlap of two or more characters is a problem that is largely dependent on careful in writing. This phenomenon often due to writing too sloping or because of an alignment between the lines of written text. In this work Tifinagh characters does not pose a big problem that they are written in a way separate in a text , then the use of a threshold in the segmentation we are helping to find a good result.

5. FEATURES EXTRACTION

The feature extraction is based on mathematical morphology [6, 7 and 8]. The characteristic zones can be detected by the intersections of dilations found to the East, West, North and South. We define for each image five types of characteristic zones: East, West, North, South, and Central zone.

5.1 Dilatation of image:

The dilatation is based on the intersection between the object of the image A (white pixels) with a structuring element B. It is defined by the following formula.

$$\textbf{Dilatation (A, B)} = \{x \in \textbf{Image} \,/\, B_x \cap A \neq \emptyset\}$$

Where A is the object of the image (the white pixels), B the structuring element which is a particular set of Center x, known size and geometry (in this work is a right half). Example of the dilatation of Tifinagh character to the East

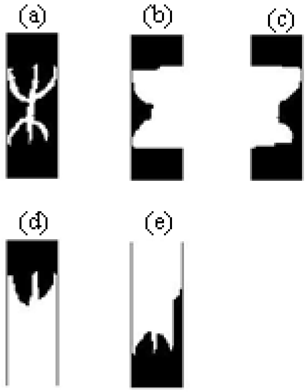


Fig 13: (a) : Image of character
(b) : Dilation of character to the West
(c) : Dilation of character to the East
(d) : Dilation of character to the South
(e) : Dilation of character to the North

5.2 Detection of the characteristic zones of the image

The characteristic zones can be detected by the intersections of dilations found to the East, West, North and South. We define for each image five types of characteristic zones: East, West, North, South, and Central zone.

5.1.1 Extraction of West characteristic zone:

A point of the image (Figure 14-a) belongs to the West characteristic zone (Figure 14-b) if and only if:
- This point does not belong to the object (the white pixels in image).
- From this point, moving in a straight line to the West, we do not cross the object.
- From this point, moving in a straight line to the South, North and East we crosses the object. The result of the extraction is illustrated in (Figure 14-b).

5.1.2 Extraction of East characteristic zone:

A point of the image (Figure 14-a) belongs to the East characteristic zone (Figure 14-c) if and only if:
- This point does not belong to the object (the white pixels in image).
- From this point, moving in a straight line to the East, we do not cross the object.
- From this point, moving in a straight line to the South, North and West we crosses the object. The result of the extraction is illustrated in (Figure 14-c).

5.1.3 Extraction of South characteristic zone:

A point of the image (Figure 14-a) belongs to the South characteristic zone (Figure 14-d) if and only if:

- This point does not belong to the object (the white pixels in image).
- From this point, moving in a straight line to the South, we do not cross the object.
- From this point, moving in a straight line to the North, East and West we crosses the object. The result of the extraction is illustrated in (Figure 14-d).

5.1.4 Extraction of North characteristic zone:

A point of the image (Figure 14-a) belongs to the North characteristic zone (Figure 14-e) if and only if:

- This point does not belong to the object (the white pixels in image).
- From this point, moving in a straight line to the North, we do not cross the object.
- From this point, moving in a straight line to the East, West and South we crosses the object. The result of the extraction is illustrated in (Figure 14-e).

5.1.5 Extraction of Central characteristic zone:

A point of the image (Figure 14-a) belongs to the Central characteristic zone if and only if:

- This point does not belong to the limit of the object.
- From this point, moving in a straight line to the South, North, East and West we cross the object. The result of the extraction is illustrated in the (Figure 14-f).

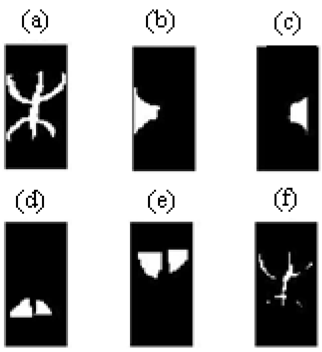


Fig 14: (a) : Image of character
(b) : West characteristic zone
(c) : East East characteristic zone
(d) : South characteristic zone
(e) : North characteristic zone
(f) : Central characteristic zone

But the problem in central zone, this three different characters (see Figure 15) they have the same central zone (see Figure 16).

Fig 15: Images of characters

Fig 16: Central characteristic zone for each Image

To remedy this problem, we changed the central area of the three characters to get the following form (see Figure 17):

Fig 17: Central characteristic zone for each Image

Each character is characterized by these five components: **NZW, NZE, NZN, NZS and NZC.**

With: **NZW, NZE, NZN, NZS, NZC**: The number of pixels of value 1 respectively in the characteristic areas West, East, North, South and Central. So the extraction vector is defined as follows:

Vext = [ZW, ZE, ZN, ZS, ZC]

Each component is defined as follows:

ZW = NZW / (Npixels).

ZE = NZE / (Npixels).

ZN = NZN / (Npixels).

ZS = NZS / (Npixels).

ZC = NZC / (Npixels).

Npixels is number of pixels in the processed image.

6. NEURAL NETWORK ARCHITECTURE (Multi layer perceptron MLP)

The neural networks [9, 10, 11 and 12] based on properties of the brain to build systems of calculation best able to resolve the type of problems as human beings live know resolve. They have several models. One of these models is the perceptron. In the classification phase we used the MLP [13 and 14], we proceed as follows: The number of neurons used in the network is:

- Five neurons in the input layer (the number five corresponds to the values found in the vector of extraction).
- Thirty neurons in the output layer (the number thirty corresponds to the numbers of characters using in recognition).
- The number of neurons in the hidden layer is chosen after its three conditions:

- Equal the number of neurons in the input layer.
- Equal 75% of number of neurons in the input layer.
- Equal the square root of the product of two layers of exit and entry.

We followed these three conditions; we varied the number of neurons in layer hidden between four and nine neurons. The method used for learning is gradient back-propagation algorithm [15 and 16].

Table 1. Details of the neural networks

Layers	Neurons
Input	5
Hidden	9
Output	30
Constant of learning	
$\alpha = 0.9$	
Squared error	
$E = \frac{1}{2} * (t - o)^2$ t: the theoretical output. o: the desired output.	
Activation function	
$F(x)=\frac{1}{(1+e^{-x})}$	

7. HIDDEN MARKOV MODELS

Hidden Markov models (HMMs), is a statistical model in which the system being modeled is assumed to be a Markov process with unknown parameters. Hidden Markov models are widely used especially in pattern recognition [17, 18 and 19], artificial intelligence or automatic natural language processing. This HMM model can be defined as:

$$\boldsymbol{\lambda = (\pi, A, B)}$$

Where $\boldsymbol{\pi}$ is initial state probability vector, **A** is a final state transition probability matrix and **B** is a final observation probability matrix. In this paper we use a closed left to right chain HMM for handwritten English characters recognition. A sketch of 5 states HMM is shown in Fig.18.

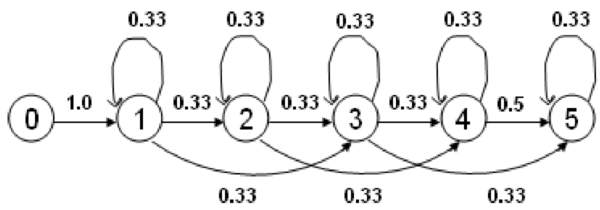


Fig 18: Left to Right Chain HMM Model with 5 States.

7.1 Training *of* HMM *Model*

We have used Baum-Welch algorithm [20 and 21], to train the HMM using observation sequence obtained from the feature vectors. At the end of training process we obtain the final value of A and B which is used for recognition purpose.

7.2 Recognition*:*

Knowing the class to which belongs the character, it is compared to models $\boldsymbol{\lambda_k}$, k = 1,…, L of its class. The selected model will be the one to provide the best probability corresponding to the evaluation of its suite of primitive i.e.:

$$\mathbf{max\ (P(O/\ \lambda_k))}$$

With,

O: The suite of observation in this work is the vector of extraction.

λk: Is the Markov model consisting of the transition matrix A, the observation matrix B and boot matrix $\boldsymbol{\Pi_i}$.

8. EXPERIMENTAL RESULTS

After feature extraction of the processed image, we identify the result by the neural network and hidden Markov model. The values of the feature vectors obtained from the extraction phase:

- Is introduced to the input of the neural network and the network is forced to converge to a specific end-state (supervised learning). Each character is characterized by a vector of extraction of five components. For training the network (the multilayer perceptron MLP) a validation base (2400 characters) is used to confirm the results (22000 characters in test base) and to provide the final rate of recognition.
- With hidden Markov model, we considering the values of the characteristic vectors obtained in recognition of the characters as a sequence of observations, it initializes the model and is sought with the Baum-Welch algorithm to find the best probability that maximizes the parameters of the model (A, B, Πi). The experimental results are illustrated in the following Table (see Table 1 and 2).

Table 2. The recognition rate for NN

Type of Base	Pen	Maker
Base of Validation	83%	85%
Base of Test	73.28%	74.39%
Numbers of validation images	1200 images	1200 Images
Numbers of Test images	11000 images	11000 Images

Table 3.The recognition rate for HMM

Type of Base	Pen	Maker
Base of Validation	77.50%	77.73%
Base of Test	71.38%	71.14%
Numbers of validation images	1200 images	1200 Images
Numbers of Test images	11000 images	11000 Images

Table 4. Experimental results with a marker for each character

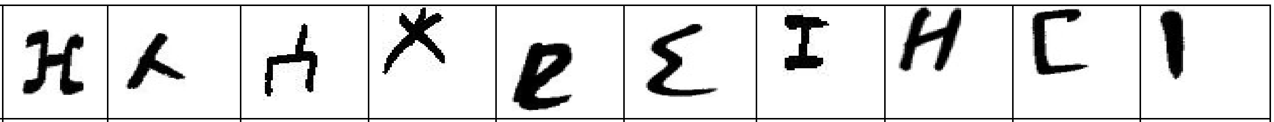
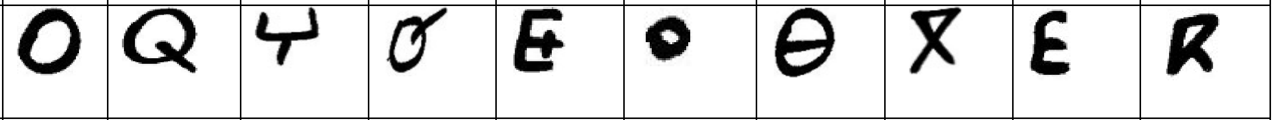
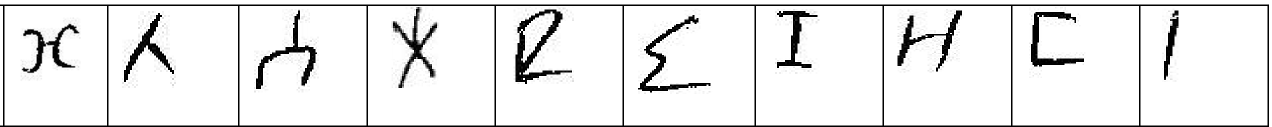
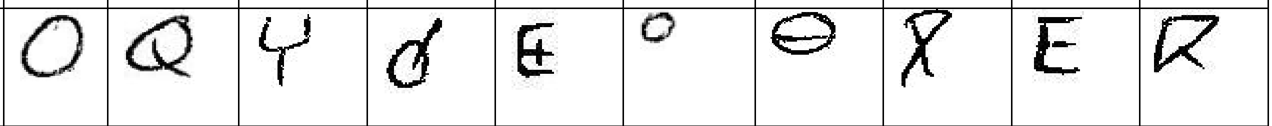
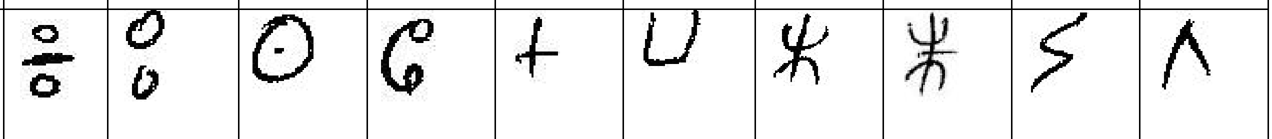
Characters										
HMM (%)	67.57	86.40	42.74	28.48	84.00	95.85	42.97	87.93	41.71	79.40
NN (%)	62.38	82.28	79.06	70.16	76.00	77.51	75.63	73.72	74.21	77.00
Characters										
HMM (%)	91.43	64.84	72.69	79.29	66.04	88.79	42.30	79.53	77.59	57.40
NN (%)	80.13	65.34	85.79	74.62	80.21	74.62	71.02	73.51	82.15	65.00
Characters										
HMM (%)	92.46	41.21	74.24	95.68	67.52	87.51	72.81	74.20	91.59	59.22
NN (%)	70.00	72.53	70.93	70.00	80.60	73.68	64.44	70.00	85.94	73.24

Table 5. Experimental results with a pen for each character

Characters										
HMM (%)	81.91	70.56	58.60	64.38	74.69	82.84	56.94	67.86	55.91	81.78
NN (%)	70.33	70.59	73.33	70.95	73.06	70.00	76.00	79.00	69.43	72.19
Characters										
HMM (%)	68.78	57.92	78.00	72.83	76.85	79.45	52.45	71.87	78.48	57.76
NN (%)	72.10	61.47	84.23	67.34	71.00	70.38	79.00	77.00	71.74	71.00
Characters										
HMM (%)	68.37	54.80	61.60	79.17	69.36	87.06	80.00	71.60	96.10	83.55
NN (%)	73.25	74.72	74.08	73.19	78.56	70.82	69.44	74.14	82.39	77.96

9. CONCLUSION

In this work a neural networks and hidden Markov models are proposed for the classification of Handwritten Tifinagh Text. Different techniques are used in the preprocessing phase before implementing identification. The segmentation phase does not pose a big problem because the characters are separated in the text but until now there is not a standard base of handwritten Tifinagh characters which facilitates the work. Experimental results have shown that the NN yield better recognition accuracy compared to HMM. But the learning with NN takes some time as the HMM. This model offers a satisfactory success rate but it is subject to further improvement.

10. REFERENCES

[1] Apurva A. Desai, Gujarati handwritten numeral optical character reorganization through neural network. Pattern Recognition 43, 2010, 2582–2589.

[2] T. Baidyk, E. Kussul, Application of neural classifier for flat image recognition in the process of microdevice assembly, Proceedings of the International Joint Conference on Neural Networks, Hawaii, USA 1 , 2002, 160–164.

[3] G. Ososkov, Effective neural network approach to image recognition and control, Proceedings of International Conference on Physics and Control 1 , 2003, 242–246.

[4] S. Vitabile, A. Gentile, F. Sorbello, A neural network based automatic road signs recognizer, Proceedings of the International Joint Conference on Neural Networks 3 , 2002, 2315–2320.

[5] J. Pradeep, E. Srinivasan and S. Himavathi, "Diagonal Based Feature Extraction for Handwritten Alphabets Recognition System Using Neural Network", International Journal of Computer Science and Information Technology (IJCSIT),, vol. 3, no. 1, pp. 27-38, Feb 2011.

[6] Kimori Y, Katayama E, Morone N, Kodama T: Fractal dimension analysis and mathematical morphology of structural changes in actin filaments imaged by electron microscopy. J Struct Biol 2011, 176:1-8

[7] Oh J, Hwang H: Feature enhancement of medical images using morphology-based homomorphic filter and differential evolution algorithm. International Journal of Control, Automation and Systems 2010, 8:857-861

[8] P. Soille, Morphological Image Analysis : Principles and Applications. 2nd edition, 2003.

[9] Anuja P. Nagare. "License Plate Character Recognition System using Neural Network. " International Journal of Computer Applications, Volume 25, No. 10, July 2011, pp. 36-39.

[10] Neural Network Classroom Approach by Satish Kumar by Tata McGraw-Hill, 2004.

[11] Yingqiao Shi, Wenbing Fan, Guodong Shi. "The Research of Printed Character Recognition based on Neural Network", Fourth International Symposium on Parallel Architectures, Algorithms and Programming. 2011.

[12] Sameeksha Barve. "Optical Character Recognition Using Artificial Neural Network. " International Journal of Advanced Research in Computer Engineering & Technology", Vol. 1, No. 4, 2012.

[13] E.M. Kussul, T.N. Baidyk, et al., Rosenblatt perceptrons for handwritten digit recognition, Proceedings of International Joint Conference on Neural Networks IJCNN 2 , 2001, 1516–1520.

[14] Hailong Liu and Xiaoqing Ding "Handwritten Character recognition Using Gradient feature and Quadratic

[15] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11), 1998, 2278–2344.

[16] Jan A. Snyman. Practical Mathematical Optimization : An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms. Springer Publishing, 2005

[17] R. El-Hajj, L. Likforman-Sulem, and C. Mokbel, "Arabic Hand- writing Recognition Using Baseline Dependent Features and Hidden Markov Modeling, Proc. Eighth Intl Conf. Document Analysis and Recognition, pp. 893-897, 2005.

[18] P. Natarajan, S. Saleem, R. Prasad, E. MacRostie, and K. Subramanian,"Multilingual Off-line Handwriting Recognition Using Hidden Markov Models: A script independent Approach", Springer Book Chapter on Arabic and Chinese Handwriting Recognition, ISSN:0302-9743, VOL. 4768, pp. 235-250, March 2008.

[19] M.-P. Schambach, "Recurrent HMMs and Cursive Handwriting Recognition Graphs," Proc. 10th Int'l Conf. Document Analysis and Recognition, pp. 1146-1150, July 2009.

[20] Press, WH; Teukolsky, SA; Vetterling, WT; Flannery, BP (2007). "Section 16.2. Viterbi Decoding". Numerical Recipes: The Art of Scientific Computing (3rd ed.). New York: Cambridge University Press. ISBN 978-0-521-88068-8. 2007

[21] Hidden Markov Models and the Baum–Welch Algorithm, IEEE Information Theory Society Newsletter, Dec. 2003.