Volume 4, Issue 3, March 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Tifinagh Character Recognition by Graphs Model Representation

M.BOUTAOUNTE*, A.ELBALAOUI, A.MERBUOHA, Y.OUADID

Faculty of Science and Technology,
Sultan Moulay Slimane University,
Beni-Mellal, Morocco

Abstract— Most methods of character recognition existing require lot of pre-processing. Also using the method's of artificial intelligent(neural network , SVM, ...) request much resources and time (time necessary for the learning phase and recognition) , and the scarcity of work aimed the Tifinagh language , we present a structural approach to recognition the Tifinagh characters using the graphs representation as a model for the offline recognition of Tifinagh characters. This representation allows extracting the basic form of each character and recognizes it by employing a pre-classification of the characters in order to earn time. The proposed approach shows its effectiveness against problems.

Keywords— Tifinagh character, Pattern recognition, graph theory, graph matching

I. INTRODUCTION

In question of the development of technology and the appearance of intelligent systems, optical character recognition is one of the most future leading by researchers around the world, which is retained their research for Latin characters negligent other language characters as (Arabic Chinese, Tifinagh ...) which are features that reduces the performance of existing systems, so in this part we tried to carry out improvement to exceed these problems, which we mentioned, low recognition rate follows the similarity of characters, and the processing time due to the high number of characters (33 characters) In this article we focus on the Amazigh language, which is a Hamito-Semitic language derived from ancient Berber. It includes a variety of dialects present from Morocco to Egypt, passing through Algeria, Tunisia, Mali, Niger and Libya. Research for the recognition of Tifinagh characters use recognition systems based on learning, artificial (neural networks, SVM ...)[1], that is why we have adopted a structural approach in which graphs [2]are used as model of representation and matching of graphs [3]as recognition method with a few tricks to improve the recognition rate and reduce the execution time, which will be explained by the following.

II. RECOGNITION SYSTEM

Our recognition system is built on three main parts: the pre-treatment phase for the PREPARING of the characters, extracting the feature and the last part the classification. In order to improve the recognition of characters, prepare images traits. Les preliminary steps that are required for this are.

A. Binarisation

Binarisation is the first pre-processing step it involves converting the scanned image into a binary image. However, the binarisation is an operation that produces two classes of black pixels and white pixels. There are so many binarisation methods that give good results in the binarisation of characters as Otsu method, ISODATA method, Method Kapur [4] in this article we have voted for the otsu method [5]

The method selected is the one adopted by "OTSU" based on the calculation of an automatic threshold by calculating the histogram given by Equation (1).

$$h(i) = \frac{n_i}{\sum n_i} \quad (1)$$

Where n_i represents the number of pixels of level i in the image.

Let q represent the estimate of class probabilities defined as:

$$q(k) = \sum_{i=1}^{k} p(k) \quad (2)$$

The separation takes place from the mean and variance given respectively by Equation (3) and (4).

$$mean = \sum_{i=1}^{k} i * h(i) \quad (3)$$

And

$$var = \sum_{i=1}^{k} (i - mean)^2 \frac{p(i)}{q(i)} \quad (4)$$

With $mean_T = mean(255)$.

For each value of k =1….255, we calculate the square of S given by Equation (5)
where 255 is the total number of gray levels.

$$s^2(k) = var_T{}^2 - var^2(k) \quad = q(k) * \big(1 - q(k)\big) * \big(mean_T - mean(k)\big)^2 \tag{5}$$

The maximum of Equation (6) is considered as the threshold .

$$S = \sqrt{s^2}(K) \tag{6}$$

The maximum is calculate by using the following equation (7)

$$s^2(k) = \max\big(s^2(i)\big) \tag{7}$$

B. Skeletisation

We have adopted the algorithm of Zhang and Wang [6] due to its robustness and speed. This is a parallel algorithm in a single iteration that produces perfectly skeletons 8-connected and which operates the collisions.
Figure 1 represent a result of image skeletisation using this method.

Fig. 1 : Example of skeletisation character

C. Singular points

We rely on the idea of HEBB [7] that any form is apprehended by the human visual perception, not as a whole, but in parts, and that the organization and the relative spatial positions of its parts play an important role in learning and recognition, structural descriptions are based on a decomposition of forms into simple elements. Then soon as the skeleton of a character obtained, we need to extract the simplest possible geometric elements of a grapheme is schematically represented by a skeleton that we casually refer to the primitive segments. The aim is to extract the singular points.
Singular points are classified into three types [8] the following types are represented in the figure 2:

- •Points end.
- •Points intersections.
- •Points inflections.

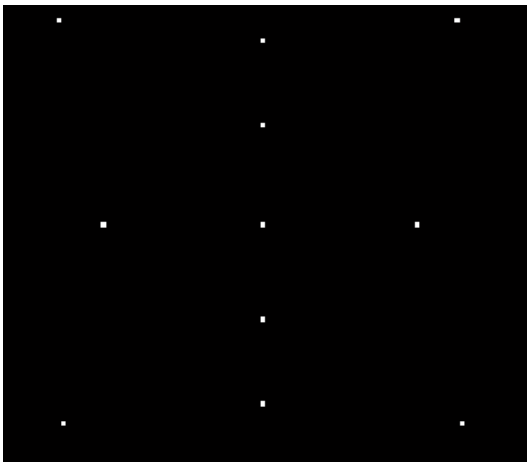
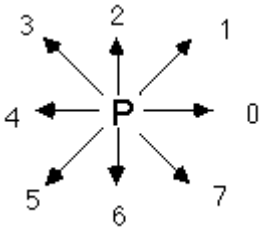


Fig. 2 : Singular points

The method adopted to detect singular points is to apply a translation on each pixel P of the character after the Skeletisation in eight directions .Figure 3 illustrate the eight directions mentioned in this paragraph.



P_3	P_2	P_1
P_4	**P**	P_0
P_5	P_6	P_7

Fig. 3 : the eight direction of translation

In our case this is done at the level of the matrix that is to say, on add to the original image pixel value the value of the pixel displaces (fig). The following figure shows the results obtained by applying this method on a character the following figure (fig. 4) represent the matrix resultant of applying the translation

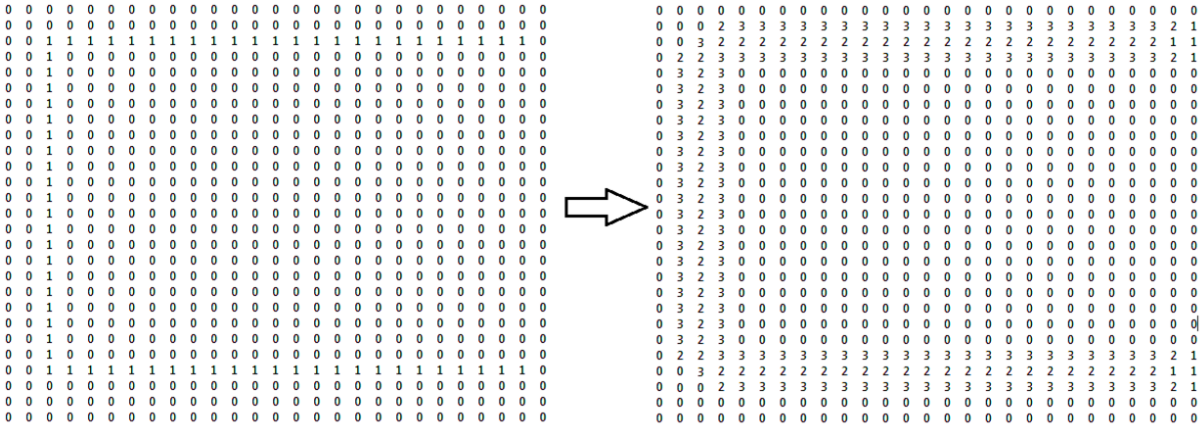


Fig. 4 : Matrix result of the translation

We perform an intersection between the original image after Skeletisation, and the result after translation in eight directions which gives us an image that has the same shape as the original image, but with different values as follows:
- The pixels having a value of 1 represent the ends
- Values strictly upper to 2 represent the inflection and intersection points.
- While equal to 2 pixels represents the primitive segments

Most recognition programs using graph theory, this base only singular point, but with the characteristics of the language in Tifinagh there are several points of similarity between the characters so we must bring some improvement, to get the best possible result by reducing the distance between the singular points by increasing number of those points[9] (fig)

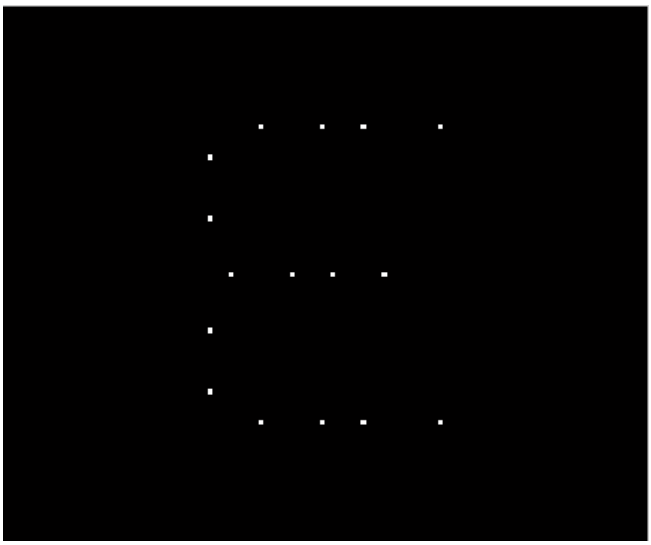


Fig. 5: Extra points add to image

D. Graph construction

A graph G is a pair G = (N, A), which N represents nodes or vertices and A represents the arcs or edges. The graphs of the most common structures are but not topologically complete.

Graphs are considered as a set of very powerful structural data for the representation of objects. In a graph representation, peaks represent objects or parts of objects. The arcs represent the relationship between the parts of objects. In our case the object is a character

Mathematically, the model are a set of nodes in form of tree connected by distance between each pair of nodes and have a single parent nodes that is the origin of tree.

Using this tree we can have the impact matrix [9] as follow:

An impact matrix M is a matrix of size n * m, such that

$$\begin{cases} n = |N| the\ number\ of\ nodes \\ m = |A|\ the\ number\ of\ edges \end{cases}$$

An element $e_{i,j}$ (i= {1… n} and j= {1… m}) of the matrix M may have two values:

$$\begin{cases} e_{i,j} = if\ arc\ j\ is\ incident\ to\ node\ i \\ e_{i,j} = 0\ if\ not\ non \end{cases}$$

Convert the image into a graph is critical because the graph should represent the characteristics of the letter exactly [10]. So in order to earn time we have proceeded by pre-classification of character, we have combined the characters that have a similar structure [11]. So it was possible to reduce the number of characters from 33 to 16. Figure 6 represent the 16 class selected to be the first step of recognition.

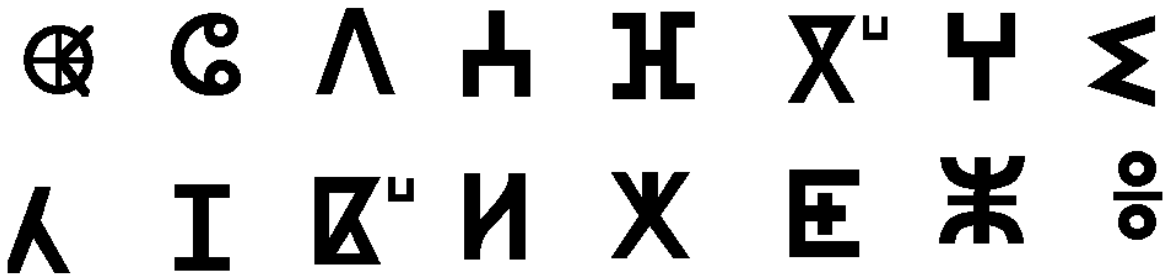


Fig. 6 : The 16 class chosen to reduce the number of characters

So the first step is to find which classes are the characters by comparing the impact matrix of the characters to matrix of the class and verify if it is include in it, after finding the class we will use the same matrix to compare the each character this class.

III. CLASSIFICATION AND RESULTS

Database used is the IRCAM Database composed of 3300 images, 100 by characters. For each character we extract the impact matrix as it is clarify in the previous section .In this experience, 2000 character images are used in the training phase and the rest for testing the performance method.

These results demonstrate interesting in terms of CPU time compared to the works carried on this kind of character. This come from the fact that we use a pre classification that reduce the number of operation needed to compared the impact matrix for each character and the way used to create the graph using the distance between points

Table .1
Recognition and error rate

training data Size	1100	2200
The test database Size	550	1100
Recognition rate (%)	92.2%	95%
Error rate (%)	7.8%	5%

IV. CONCLUSION

Using a graph representation model for recognition of Tifinagh characters allows as having a height rate of recognition with less time than the other method of recognition and also this result can be combined with others methods like SVM, NN to ameliorate the recognition rate.

The work done is a first step for several perspectives. We try to improve our recognition rate by changing the graph model for a better representation keeping the speed of the system. We try to extend our approach to the recognition of printed handwriting recognition characters, the recognition of words, sentences and texts.

REFERENCES

[1] s Saady, Y., Rachidi, A., El Yassa, M., & Mammass, D. (2011). Amazigh handwritten character recognition based on horizontal and vertical centerline of character. International Journal of Advanced Science and Technology, 33.

[2] LING-HWE! CHEN and JUNE-RONG LIEH HANDWRITTEN CHARACTER RECOGNITION USING A 2-LAYER RANDOM GRAPH MODEL BY RELAXATION MATCHING, Pattern Recognition. Vol, 23. No 11. pp. 1189-12o5. 1990 Printed m Great Britain

[3] Horst Bunke Graph Matching: Theoretical Foundations, Algorithms, and Applications , Department of Computer Science University of Bern, Neubr¨uckstr. 10, CH-3012 Bern, Switzerland

[4] M. R. Gupta, N. P. Jacobson, E. K. Garcia, « OCR binarization and image preprocessing

[5] for searching historical documents », Pattern Recognition, 40, p. 389-397, 2007.

[6] Thi, N., Oanh & Tabbone, S. (2004). Binarisation d'images de documents graphiques. Technical report Laboratoire Lorrain de Recherche en Informatique et ses Applications à Université de Nancy. Bellman,

[7] *Y.Y. Zhang, P.S.P.Wang,«A new parallel thinning methodology». International Journalof Pattern Recognition and Artificial Intelligence, 1994.*

[8] *D. Arrivault, «Apport des Graphes dans la ReconnaissanceNon-Contrainte de Caractères Manuscrits Anciens», pp 61, 2002.*

[9] *D. Arrivault, «Apport des Graphes dans la ReconnaissanceNon-Contrainte de Caractères Manuscrits Anciens», pp 74-77, 2002.*

[10] *Ho-Yon Kim*, Jin H. Kim Hierarchical random graph representation of handwritten characters and its application to Hangul recognition Pattern Recognition 34 (2001) 187}201*

[11] *N. Shobha Rani,«Optical Character Recognition In Application With Graph Theory AndMatrix Processing», IJCSRT, 2013*

[12] *P.N. Suganthan, H. Yanb Recognition of handprinted Chinese characters by constrained graph matching Image and Vision Computing 16 (1998) 191-201*

[13] *Jinsoo Lee,Wook-Shin Han,Romans Kasperovics,Jeong-Hoon Lee An In depth Comparison of Subgraph Isomorphism Algorithms in Graph Databases School of Computer Science and Engineering Kyungpook National University, Korea*

