

Tifinagh Character Recognition using Harris Corner Detector and Graph Representation

Youssef Ouadid

Department of
Computer Science Sultan,
Laboratory of Information
Processing and Decision and
Support,
Moulay Slimane University
Faculty of Science and
Techniques
B.P 523 Beni - Mellal Morocco

Mehdi Boutaounte

Moulay Slimane University
Faculty of Science and
Techniques B.P 523
Beni - Mellal Morocco

Mohamed Fakir

Department of
Computer Science Sultan,
Laboratory of Information
Processing and Decision and
Support,
Moulay Slimane University
Faculty of Science and
Techniques B.P 523
Beni - Mellal Morocco.

Brahim Minaoui

Department of
Computer Science Sultan,
Laboratory of Information
Processing and Decision and Support,
Moulay Slimane University
Faculty of Science and Techniques
B.P 523 Beni - Mellal Morocco.

ABSTRACT

In this article, a printed Tifinagh character recognition system is presented. It is composed of three main steps: pre-processing, features extraction and classification. In the pre-processing step, binarization, normalization and thinning are applied to enhance the quality of the image. In the feature extraction step, corners are extracted using Harris Corner Detection method then represented by graph. In the classification step, graphs are compared using a graph matching method. Experimental results are obtained using IRCAM database to test the effectiveness. The system shows good results in terms of accuracy.

Keywords

Character Recognition, Graph Matching, Tifinagh Character, Harris Method

1. INTRODUCTION

The Amazigh language is spoken today by about 14 million people, mainly in the Maghreb. The Amazigh writing is normally written from left to right and vertically from top to bottom, it is a non cursive writing simplifying the segmentation of characters in a text image Amazigh. Figure 1 illustrates the Tifinagh characters adopted by IRCAM.



Fig 1: Tifinagh of the Royal Institute of Amazigh Culture, IRCAM

The OCR system is a tool to facilitate human-machine interaction. It is the process that converts a text image in a text document modifiable by the computer or similar equipment. It is used in many areas where the work is based on text documents, mainly in the office for the purpose of indexing or automatic document archiving for example. Many researches have been concentrated in the creation of an optical recognition system of Chinese characters [1] Arabic [2], particularly Latin characters. However, recognition of Tifinagh characters, are less explored.

Recent proposed approaches for the recognition of printed Tifinagh characters are generally based on the hybridization of several descriptors [3,4], classifiers [5,6], or both [7]; which gives very good results in terms of recognition rate. In return, combining several descriptors / classifiers increases significantly the complexity of the system. Other interesting systems have been proposed, Fakir et al. [8] presented a system based on the use of Riemannian manifolds and classification by SVM (Support Vector Machine). Gounane et al. [9] proposed two systems to compare two methods of

clustering, self-organizing map and k-nearest neighbors. To minimize the error rate, the bigram was used [10]. In a new approach, Es-Saady et al. [11] proposed a system based on the vertical and horizontal histogram to extract the feature vector which is classified by the Hidden Markov Models. A similar approach was proposed by Amrouch et al. [12] where they used a directional characteristic vector sequence which led to good results. El Ayachi et al. [13] proposed a recognition system based on Brele code as character feature, it is a fast system but accuracy need to be work on.

Most of works reviewed are concentrated on the accuracy of the recognition system while the complexity still needs to be work on. In this paper, a fast and accurate system for the recognition of Tifinagh characters is presented. It is based on graph theory for character feature representation and classification. It contains three main steps: Pre-processing, Extraction and Classification as shown in figure 2. Interest points and their relationship are used as features to represent the character, then the graph is classified using spectral graph matching.

The organization of the remainder of this paper is as follows: section II deals with the preprocessing which consist of normalization & thinning of the character. In section III, features extraction will be described. Section IV deals with the classification method. In the last section, experimental results and discussion are given.

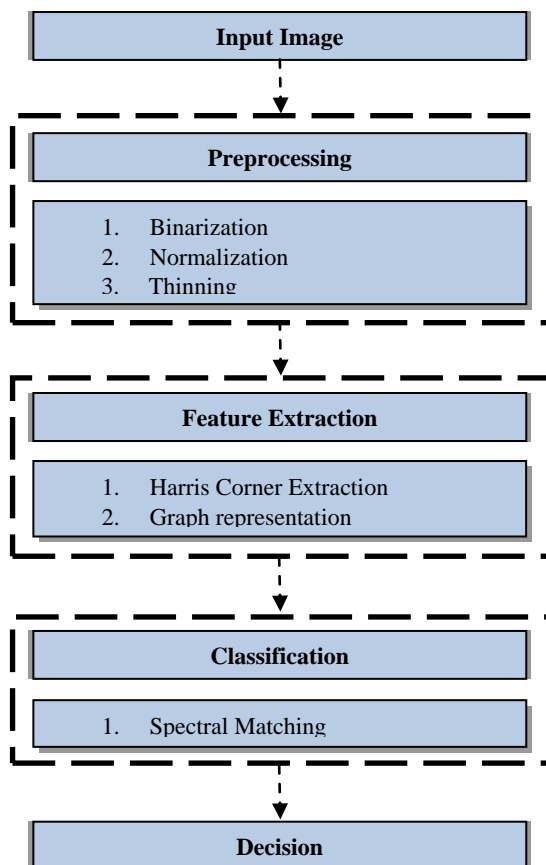


Fig 2: Tifinagh Character Recognition System

2. PREPROCESSING

The preprocessing aims to reduce the number of data to keep only relevant information. It consist of binarization, normalization and thinning. Since our database is composed of monochrome images. Authors considered using only normalization and thinning.

2.1 Normalization

Normalization is a first step of pre-processing to remove the unwanted area and put the character in the center of the image. To do it, the vertical and horizontal histogram of the image are calculated to find the border of the character. The horizontal histogram is scanned vertically and vertical histogram horizontally from both direction to find the first white pixel in top, bottom, left and right. Fig. 3-(b) illustrate an example of character image normalization.

2.2 Thinning

The skeleton of the character is extracted using the method of Zhang-Suen [14]. It is a parallel thinning algorithm in two iterations to eliminate pixels that do not belong to the skeleton of the character as shown in figure 3-(c). However, in some cases, the algorithm produces two pixel thickness skeletons in some places. Thus, in the case of the character number 25, the filled circle is eliminated entirely. To overcome these problems the following solutions are proposed:

- Apply thinning on connected components instead of applying it to the entire image. This allows us to treat the case of removed components.
- Smooth the skeleton by eliminating pixels that do not affect the connectivity and topology of the character as shown in figure 3-(d).

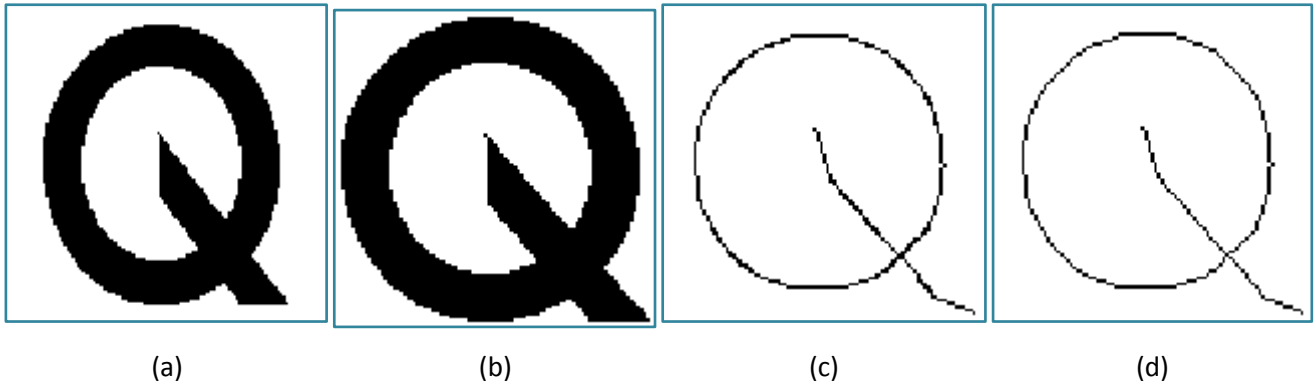


Fig 3 : Example of preprocessing functions applied on a Tifinagh character, (a) Original monochrome Image, (b) Normalization, (c) Thinning, (d) Skeleton smoothing

3. FEATURE EXTRACTION

The goal is to provide a structural description of the character. First, the key points of the character are extracted using Harris Corner Detector [15] (an example is illustrated in figure 5 in experimental results section). Then the character is represented by a graph, which will allow us to provide important information such as key points position and their relationships.

3.1 Harris Corner Detection

The Harris algorithm is an algorithm that allows a discrete representation of the image characteristics. Its objective is to extract the interest of points that give information of the image structure. This is an improvement of the Moravec corners detector [16]. The principle of Moravec algorithm is to consider a rectangular local window in the image to determine the average of the changes in the intensity in image by moving it to several directions. When the window undergoes a large sequence of change; the minimum of it, is the corner or the point of interest. Assume that I the intensity of the image, the changing E produced by the movement (x,y) is given by:

$$E_{x,y} = \sum_{u,v} w_{u,v} |I_{x+u,y+v} - I_{u,v}|^2 \quad (1)$$

Where $w_{u,v} = \exp - (u^2 + v^2)/2\sigma^2$.

$w_{u,v}$ is the local window in the image and movement $(x,y) \in \{(1,0), (1,1), (0,1), (-1,1)\}$. The process for finding points of interest is the same as looking for the local maximum of $\min\{E\}$. The Moravec algorithm suffers from some problems such as the limited movement of the window (four directions), Binary and square window. Harris algorithm remedy these problems by reading the layout of the window and improving the way it moves the changing of intensity E given by equation (2):

$$E(x,y) = Ax^2 + 2Cxy + By^2 \quad (2)$$

Where $A = \frac{\partial^2 I}{\partial x^2} \otimes w$, $B = \frac{\partial^2 I}{\partial y^2} \otimes w$ and $C = \left(\frac{\partial I}{\partial x} \frac{\partial I}{\partial y}\right) \otimes w$.

Harris has also introduced a measure of quality and sensitivity given respectively by equation (3) & (4).

$$\text{Tr}(M) = A + B \quad (3)$$

$$\text{Det}(M) = AB - C^2 \quad (4)$$

The determination of corner region is done by response measure R . given by equation (5).

$$R = \text{Det} - k\text{Tr}^2 \quad (5)$$

3.2 Graph Construction

A graph is a formal mathematical representation of a set of objects and their relationships. Each object is called a vertex. Relationships between objects are called edges. More formally, a graph G is defined as an ordered pair of $G = (V, E)$ where V is a set of vertices, E is a set of edges and each edge is a pair of vertices.

Adjacency matrix is a way to represent graphs. It is a binary square matrix M where the number of vertices $|V|$ is its size. The entry in row i and column j is 1 if and only if the edge (i, j) is in the graph. Which means:

$$\begin{cases} AM(i, j) = 1, & \text{nodes } i \text{ and } j \text{ are linked by an edge} \\ AM(i, j) = 0, & \text{nodes } i \text{ and } j \text{ are not linked by an edge} \end{cases} \quad (6)$$

Some graphs can be directed which means $AM(i, j) \neq AM(j, i)$, or undirected which means $AM(i, j) = AM(j, i)$. In our case the undirected weighted graphs representation is used, which means that the adjacency matrix AM is symmetric where the nonzero values represent the features of edges. The Euclidian distance is used as weight of edges.

Converting the image into a graph is critical because the graph should represent the characteristics of the letter to a large extent. The distance between two nodes must be high enough to represent the characteristics of the letter properly and also small enough to not take a lot of memory. To do it, based on interest points extracted, the algorithm is proceeded as follow:

1. Set interest points to the background of the character skeleton image.
2. Label connected component in the character skeleton image.
3. Return interest points to the foreground of the character skeleton image.
4. Build adjacency matrix based on the labeled image of skeletonized character.

Fig. 4 illustrates some examples of application of the Harris algorithm and its graphs representation on the Tifinagh characters.

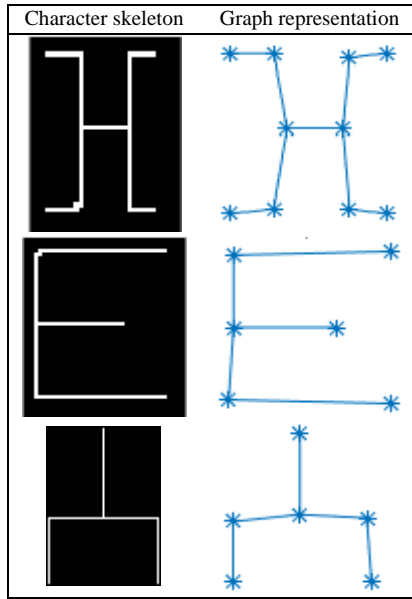


Fig 4: Examples of graph representation of Tifinagh Character based on interest points detector by Harris method

4. CLASSIFICATION

4.1 Spectral graph matching

The spectral method [17] aims to find coherent agreement between two sets of features P (P contain n_p data features) and Q (Q contain n_q data features). It is generally used to find the main cluster in a graph. This is done by constructing the adjacency matrix M , where vertices represent the potential assignment ($a = (i, i'), i \in P$ and $i' \in Q$) between two features and arcs design the level of resemblance between these assignments. This level is represented by a positive weight if two pairs of assignments are well agreed otherwise the link have zero weight. In our case, one-to-one correspondence mapping (instead of one-to-many) is used. Which means that one feature of the first sets of features can be assigned with only one feature from the second set. This representation takes into account the geometry of the character and the degree of similarity between features. By observing of the adjacency matrix M , the spectral properties are extracted. It can help us to determine how well a pair of assignments is connected to the main cluster. Algorithm keep rejecting correspondence with low association until the one-to-one correspondence mapping is reached. The proceeding to construct the matrix M is as follow: Given C a set of pairs where $a = (i, i') \in C$. It contains all possible assignments that respect the correspondence mapping one-to-one (one feature from P assigned at most to one feature from Q). To measure agreement between two features from P and Q , it is considered a list L of an associated score for each candidate assignment $a \in C$ and an associated affinity for each pairs of assignments (a, b) where $a, b \in C$. Based on this list the scores on the matrix M are stored as follow:

The diagonal entries of M contain the score of individual assignment $a \in C$.

Other entries contain the score of pairs assignment (a, b) where $a, b \in C$.

The matrix M is a square matrix of size $n = k * n_p$ where k is the average number of candidate of each data features $i \in P$. Since interest points are used as features, all pairs of

assignments are candidate assignment. Now the correspondence problem is reduced to find the cluster C that maximizes the inter cluster score.

$$S = \sum_{a,b \in C} M(a, b) = x^T M x \quad (7)$$

x is an indicator vector where:

$$\begin{cases} x(a) = 1 & \text{if } a \in C \\ x(a) = 0 & \text{if } a \notin C \end{cases} \quad (8)$$

To find the best cluster C^* we have to find the optimal binary vector x^* that maximize the inter cluster score S .

$$x^* = \operatorname{argmax}(x^T M x) \quad (9)$$

The value of this score depends on the number of assignments, the number of arcs adjacent to this assignments and the weight on arcs. The higher is the score S the bigger is the similarity between characters represented by M .

This method is applied by following the steps described in the algorithm below:

1. Build the symmetric non-negative $n \times n$ matrix M .
2. Initialize L by all possible candidate assignments and x a $1 \times n$ zeros vectors.
3. Find $a^* = \operatorname{argmax}_{a \in L}(x^*(a))$, where x^* is the principal eigenvector of M is.
4. If $x^*(a^*) = 0$ stop and return the solution x . Else, set $x(a^*) = 1$ and remove a^* from L .
5. Remove all assignments in conflict with a^* (one to one correspondence constraint).
6. If L is empty return the solution x . Else, return to step three & four.

5. RESULTS AND DISCUSSION

In this work, authors objective is to produce a fast and accurate System of Tifinagh Character Recognition. In preprocessing phase, the histogram method for normalization and Zhang-Suen algorithm for thinning are used, with produced a thinned version of the skeleton with the unwanted information removed. Fig. 6 illustrates the computing time of preprocessing phase. In Feature extraction phase, Harris corner detection method is applied on the thinned version of the character. Then the adjacency matrix that represent the character is constructed. Since the work is done on binary images the interest point extraction is not accurate as wanted, especially in characters that contain curves and branch points that contain more than three neighbor. This problem is solved, by clustering close interest points into one point as shown in fig. 5. By analyzing fig. 7&8, it is understandable that number of corners do not affect the CPU-Time of extraction. Actually, it is the complexity of the character and the diversity of writing style. In order to evaluate the overall performance of our system, IRCAM database is used. It is composed of 3300 binary image, each character have 100 samples. Based on 33 images references, 3267 images are used to test our system. The proposed system is compared with some of the best systems of printed Tifinagh character recognition. The table below shows the recognition rate, error rate and CPU-Time obtained for each system. The system have some confusion problem between which character number 01 & 22. But when looking into the database, it is noted that there is some images of the two characters that are very similar even for the human eyes. Fig. 9 shows two similar image of character number 01 & 22. As shown in the table 2, our method outperforms other methods in term of recognition rate/CPU-time.

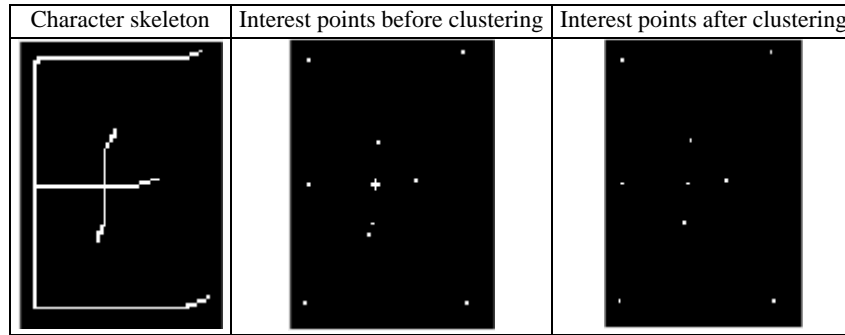


Fig 5: Example of interest points clustering

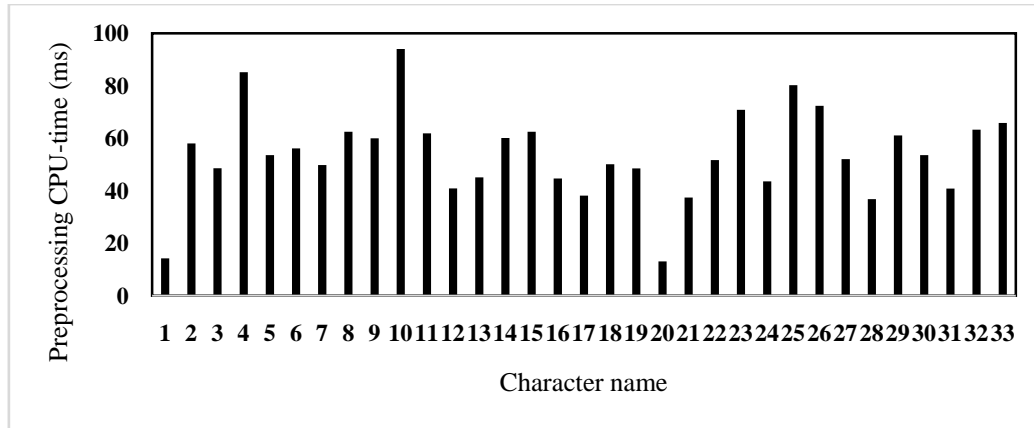


Fig 6: Preprocessing CPU-Time/Character id

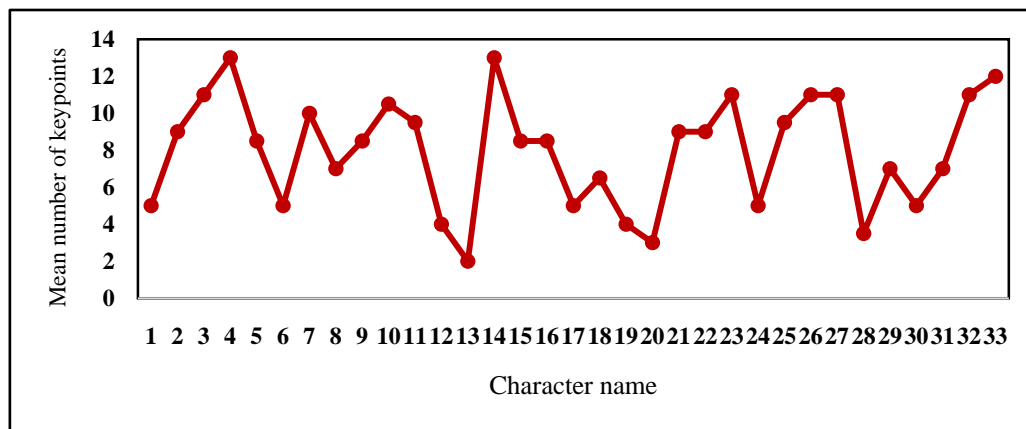


Fig 7: Mean number of interest points per character

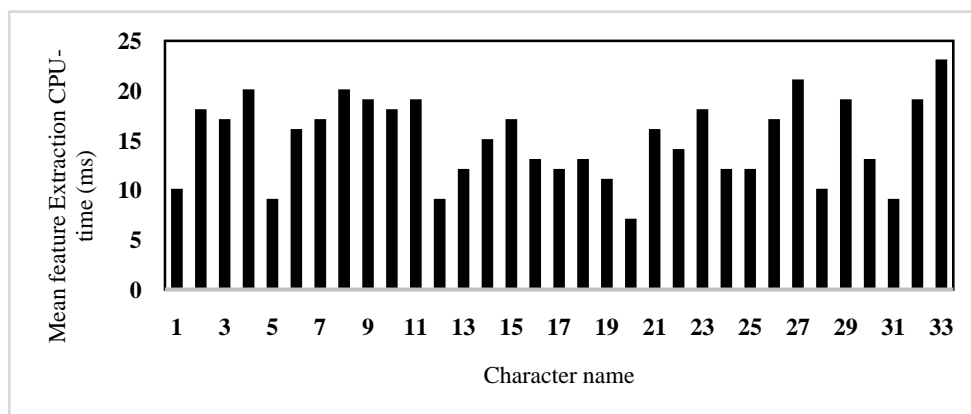


Fig 8: CPU time of corner points extraction by Harris method

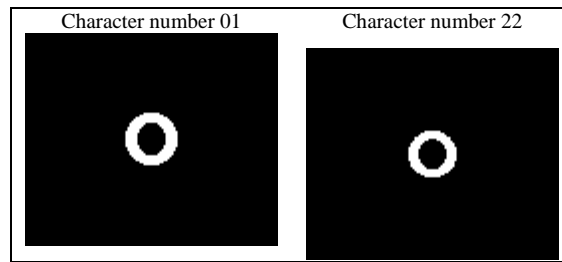


Fig 9: Example of two similar character in the database

Table 1: Recognition System comparison

Recognition system	Descriptor	Harris method & Graph representation	Gist Descriptor [3]	Moments of Legendre [17]
	Classifier	Spectral graph matching	Bayesian Network classifier	Multilayer Neural Networks
Recognition rate (%)		99	98.18	81.21
Error rate (%)		1	1.82	18.79
CPU Time (sec)		3550	5912.676	28 154.28

6. CONCLUSION

In this work, a fast and accurate system for the recognition of Tifinagh characters is presented. It is based on the matching of graphs via an efficient spectral matching method. It uses the spectral properties of the affinity matrix to calculate the score of similarity between graphs. Each graph represents the structural features of the character that are the key points and their connectivity. The outcomes of the experiments show that most characters are recognized without compromising the execution time. In future work, Author consider applying the system on tiffinagh handwriting with a larger database and higher level of noise to make the system more reliable.

7. REFERENCES

- [1] S Mansi, G B Jethava. (2013). A Literature Review On Hand Written character Recognition. Indian Streams Research Journal. Vol-3, issue –2.
- [2] M L Lorigo, V Govindaraju. (2006). Offline Arabic handwriting recognition: a survey. IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [3] M Oujaoura, B Minaoui and M Fakir. (2013). Walsh, Texture and GIST Descriptors with Bayesian Networks for Recognition of Tifinagh Characters. International Journal of Computer Applications. Volume 81, No.12.
- [4] R El Ayachi, M Fakir, B Bouikhalene. (2012). Transformation de Fourier et moments invariants appliqués à la reconnaissance des caractères Tifinaghe. e-TI - la revue électronique des technologies d'information. Number 6.
- [5] O Bencharef, M Fakir, B Minaoui and B Bouikhalene. (2011). Tifinagh Character Recognition Using Geodesic Distances, Decision Trees & Neural Networks. (IJACSA) International Journal of Advanced Computer Science and Applications, Special Issue on Artificial Intelligence.
- [6] B El Kessab, C Daoui, K Moro, B Bouikhalene and M Fakir. (2011). Recognition of Handwritten Tifinagh Characters Using a Multilayer Neural Networks and Hidden Markov Model. Global Journal of Computer Science and Technology. Volume 11, Issue 15, Version 1.0.
- [7] M Oujaoura, B Minaoui, M Fakir and O Bencharef. (2014). Recognition of Isolated Printed Tifinagh Characters. International Journal of Computer Applications. Volume 85, No 1.
- [8] M Fakir, O Bencharef, B Bouikhalene and B Minaoui. (2011). Tifinagh Character Recognition Using Riemannian Metric, SVM & Neural Networks. International Journal of Advances in Science and Technology. Vol 2, No 6.
- [9] S Gounane, M Fakir and B Bouikhalene. (2011). Recognition of Tifinagh Characters Using Self Organizing Map and Fuzzy K-Nearest Neighbor. Global Journal of Computer Science and Technology. Volume 11, Issue 15.
- [10] S Gounane, M Fakir and B Bouikhalene. (2013). Handwritten Tifinagh Text Recognition Using Fuzzy K-NN and Bi-gram Language Model. IJACSA Special Issue on Selected Papers from Third international symposium on Automatic Amazighe processing (SITACAM' 13).
- [11] Y Es-Saady, M Amrouch, A Rachidi, M El Yassa and D Mammass. (2012). Réalisation d'un OCR pour l'écriture Amazighe imprimée. 5th International Conference on ICT for Amazigh.
- [12] M Amrouch, Y Es-saady, A Rachidi and M El Yassa and D Mammass. (2012). Handwritten Amazighe Character Recognition System Based on Continuous HMMs and Directional Features. International Journal of Modern Engineering Research (IJMER). Vol 2, Issue 2. Pages 436-441.
- [13] R El Ayachi, M Oujaoura, M Fakir and B Minaoui. (2014). Code Braille et la reconnaissance d'un document écrit en Tifinagh, The International Conference on

Information and Communication Technologies for the Amazigh (TICAM).

- [14] T Y Zhang, C Y Suen. (1984). A Fast Parallel Algorithm for Thinning Digital Patterns. Communications of the ACM, Volume-27, Number-3.
- [15] Chris Harris, Mike Stephens. (1988). A Combined Corner And Edge Detector. Plessey Research Roke Manor, United Kingdom, The Plessey Company pic.
- [16] H Moravec. (1980). Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover. Tech Report CMU-RI-TR-3, Carnegie-Mellon University, Robotics Institute.
- [17] M Leordeanu, M Hebert. (2005). A spectral technique for correspondence problems using pairwise constraints, International Conference of Computer Vision.