Handwritten Arabic Digits Recognition Using Bézier Curves

Aissa Kerkour El Miad and Azzeddine Mazroui

University Mohammed First, Faculty of Sciences, Oujda, Morocco

Abstract

In this paper we propose a new recognition approach for Arabic numerals. Given that the performance of recognition systems for Arabic numerals are closely linked to the choice of features and classification system used in the recognition phase, we seek to exploit the possibilities of the theory of Bézier curves that allows representing parametric curves from a limited number of data (some characteristic dots with their derivatives). Indeed, the characteristic dots of the Arabic digit that we have adopted are those such that their associated Bézier curve is close to the shape of the digit. The used classifier in this work is the k-nearest neighbour. The obtained results testify to the interest and the strength of our approach.

Keywords: Handwritten digits recognition, Image processing, Spline, Bézier curves, Skeletonization, Feature extraction, Training.

1. Introduction

The recognition of handwriting text is a topic widely studied by the scientific community. It has been in recent decades the subject of several research works. Some studies are devoted to the digit recognition given its growing interest in many applications such as postal mail sorting and bank check processing. The accuracy and speed of execution of these applications are widely characteristics that are sought by users. However, the diversity of writing styles of writers makes this subject difficult and stimulates many researchers to develop high performance applications. The performances of recognition systems depend strongly on the choice of approaches used in the feature extraction approach and the classification techniques relating to training and testing phases.

Among the most used feature extraction approaches, we quote statistical methods based on histograms [1-3], and those using local variations in digit shapes [4-5]. Our approach fits in line with the approaches of the second category. In fact, the features that we have adopted are the dots where the digit shape shows a strong variation (change of direction, inflection dots and cusps) accompanied by their tangents. The theory of Bézier curves explains the choice of these features [6]. Indeed, an appropriate selection of a limited number of dots with their tangents allows building very close shape to that of digit.

A broad family of classifiers have been used by several research teams. The artificial neural networks (ANN), the k-nearest neighbours (k-NN), the hidden Markov models (HMM) and the support vector machine (SVM) are among the most frequently used classifiers [7]. To improve the performance of recognition systems, some authors have used hybrid classifiers which consist in using two classifiers [8]. In this work we have used the 1-nearest neighbour classifier.

The paper is organized as follows. In Section 2 we give a state of the art related to Arabic recognition digit systems. Section 3 is devoted to the pre-processing steps. After, we recall in Section 4 the main properties of Bézier curve theory. We describe in Section 5 the proposed feature extraction approach. Then, we explain in Sections 6 and 7 respectively the training phase and the recognition phase. Finally, Section 8 addresses and analyzes the experimental results, and we end this paper by a conclusion and a brief description of future works.

2. State of the art

S. Mahmoud [9] presented a recognition system for handwritten Indian numerals. He used a set of 120 features computed from angle span, distance span, horizontal span and vertical span. The HMM and 1-NN were used as classifiers. He tested these classifiers with different sets of these 120 features in order to select the features and the classifier giving the highest recognition rate. He concludes that the results obtained with HMM classifier are better than the 1-NN classifier.

D. Sharma et al. [10] proposed a Zone feature extracted method for the recognition of handwritten numerals. It consists to divide firstly the whole image in 4×4 zones. In order to gain more accuracy these zones are divided into 6×6 zones. The division of zones carried out up to 8×8 zones. The features are the densities of object pixels in each zone. Finally, 116 of such features are extracted for classification and recognition. 1-NN classifier was used for classification and recognition.

S. Impedove et al. [2] developed a novel prototype generation technique to recognize handwritten digit. The features are computed from binary histograms of oriented gradients, and the k-NN classifier was used in two stage processes to reduce the classification time. The first step

used the adaptive resonance theory to determine the number of prototypes and select an effective initial solution, and an evolution strategy was used in the second step to generate the final solution.
In order to improve the performance of recognition systems, some approaches based on hybrid classifier have been developed in recent years [8, 11, 12]. In [8], X. Niu et al. used the convolutional neural network to extract the features and the SVM classifier to recognize the unknown pattern.
A state of the art of the main methods developed in the field of OCR for Arabic was made by L. M. Lorigo et al. [13], and more recently by A. M. AL-Shatnawi et al. [14] and by A. Mesleh et al. [15]. For digits, C. L. Liu et al. describes in [1] the different techniques used in digit recognition systems.

3. **Pre-processing**

Before extracting the features of each digit image, a pre-processing step is required. It consists in removing the unnecessary information in the image and keep only useful information.

3.1 Removing noise

The approach that we have adopted in the feature extraction phase is to use the skeleton of the digit instead of its original form. Following the phase of skeletonization (see paragraph 2 below), some branches appear in the skeleton of the digit in the form of noise. During the feature extraction phase, these branches can be detected as primitives. So, we conducted a filtering before skeletonization phase to prevent the appearance of these branches (see fig 1).

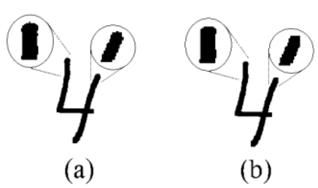

Fig. 1 - (a): before filtering ; (b): after filtering.

3.2 Skeletonization

In many cases, the treatment of skeleton of the digit instead of its raw form is less expensive in terms of time and more interesting in terms of accuracy. Thus, we chose to analyze the skeleton of the digit instead of its initial shape. The skeletonization algorithm that we used is that developed by Zhang-Wang [16]. It is known by its speed and its adaptation to Arabic numerals (see Fig. 2(b)).

3.3 Straightening of shapes

Sometimes we noticed, after the skeletonization phase, the onset of a quirky pixel in straight parts of the skeleton. To avoid being selected as feature in the next step, we proceed to the straightening of these pixels (see Fig. 2(c)).

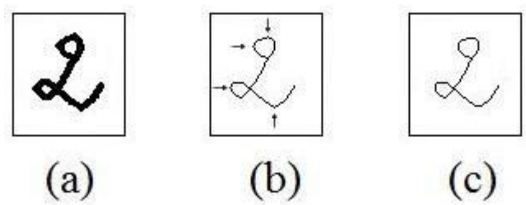

Fig. 2 (a) The initial digit ; (b) The digit after skeletonization ; (c) The skeleton after adjustment of shape

3.4 Resizing

A standardization phase of image sizes is necessary in order to compare the features of the digit to recognize to those of learned digit. To do this, we first frame the digit (i.e. identify the smallest rectangle containing the digit (see Fig. 2 (d)), then we put the framed image at the center of a 128 × 128 window (see fig. 2 (e)).

4. **Bézier Model**

Bézier curves are parametric piecewise polynomial curves. They were introduced for the first time by Pierre Bézier [2]. They are easy to build and they have interesting properties for graphic design. Indeed, for each four distinct dots P_0, P_1, P_2 and P_3, there exists a unique cubic Bézier curve which starts at P_0 and arrives at P_3, and has the vectors P_0P_1 and P_2P_3 as tangent vectors respectively at the dots P_0 and P_3. The shape of this cubic curve is controlled by the envelope P_0P_1, P_1P_2 and P_2P_3 (see Fig. 3).
Furthermore, if we move only one dot P_i, we obtain variations of the initial curve. This is used by typographers to refine the plotted curves (see Fig. 3).

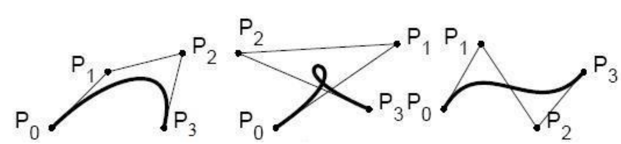

Fig.3 According to positions of four dots P_i, we obtain various forms of Bézier curves.

The continuous splines of degree 3 are obtained by connecting these curves. So, the skeleton of the digit can be seen as a continuous spline (see Fig. 3(d)). The features of the digit are the points and the derivatives which the associated spline is close to the shape of the skeleton. To illustrate this feature extraction method, we treat the case of the digit 2. We partition the shape of this digit into two curves. The first curve starts at Q_1 and arrives at Q_2 and the second starts at Q_2 and arrives at Q_3. After building a spline from dots Q_i and adapted tangents at these dots, we obtain a shape close to that of the digit 2 (see Fig. 4).

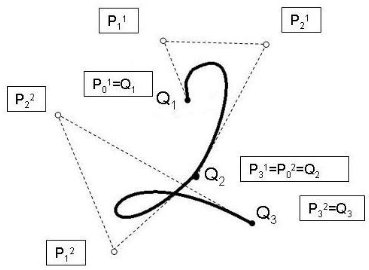

Fig. 4 Bézier data required to reconstruct the digit 2.

Thus, we can characterize each digit by a limited number of dots equipped with their tangents. The Features are the extremity dots of the digit, in addition to some dots where the shape of the digit presents a variation (changes of direction, inflexion dots and cusps).

5. Features extraction

After the pre-processing step, the resulting shape is a sufficiently smooth skeleton. A study on local variations in the skeleton shape will allow us to identify the features.

5.1 Digit classes

We distinguish two classes of the Arabic digits:

- the class LD of digits that have a loop,
- the class NLD of digits without a loop.

By analyzing the shapes of digits written by different writers, we noticed that some writers write the numeral 2 with a loop and 4 without loop, so:

- LD class is composed of digits 0, 4, 6, 8, 9 and the digit 2 written with a loop.
- NLD class is composed of remaining digits and the digit 4 written without loop.

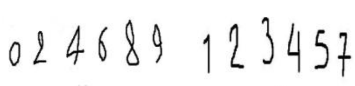

Fig. 5 (a): class LD ; (b): class NLD

A digit is an element of the LD class if browsing its skeleton in a given direction we meet a pixel twice.

5.2 Extremity dots

As the extremity pixels have only one neighbour, their identification will be through one of the following eight masks:

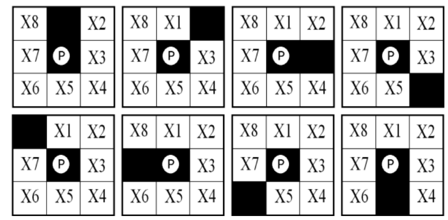

Fig. 6 detection masks of extremity pixels

5.3 Singular dots

For each pixel, we consider the eight possible derivative directions d_k:

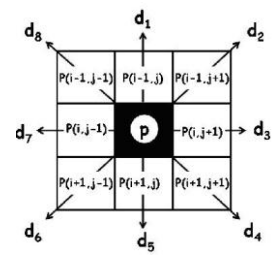

Fig. 7 Eight derivative directions of the pixel P

Definition: A pixel P is non-singular if it is aligned with its two neighbour pixels, i.e. the two associate directions d_i and d_k of its derivatives are collinear, which is equivalent to $|i-k|=4$.

The two first masks below are examples of used masks to detect singular dots and the last one is an example of used masks to detect non-singular dots.

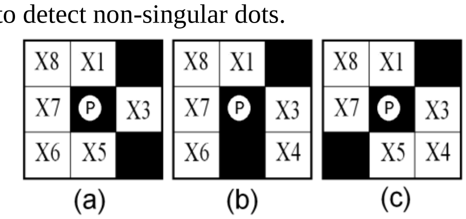

Fig. 8 (a) and (b): singular pixels ; (c): non-singular pixel

The first step of our algorithm consists in removing the non-singular pixels.

5.4 Characteristic dots

The change in directions of derivatives for some singular dots is not necessary due to a strong variation in the digit shape, but can be a result of the digital image processing or the skeletonization algorithm. In this case, these dots should not be considered as features. Hence, we use a second filter to remove them. It consists to eliminate any singular dot P that is almost aligned with its nearest neighbours P_b and P_a, where P_b (resp. P_a) is the singular dot detected just before (resp. just after) P. In the case where P is the first detected singular dot, the dot P_b will be the first extremity dot and where P is the last detected singular pixel, P_a will be the last extremity dot.

For a given digit D, we denote by P_1 and P_m the two extremity dots and $(P_i)_{2\leq i\leq m-1}$, the singular dots of D. Let (u_i,v_i) the two derivative directions of the dot P_i, with the extension $u_0 = v_m = 0$.

Lemma: for each $2\leq i \leq m-1$, the directions v_{i-1} and u_i are parallel and the directions v_{i-1} and v_i are not parallel.

Proof: as P_i is the first singular dot detected after the singular dot P_{i-1}, it is clear that the directions v_{i-1} and u_i are parallel. Similarly, since the directions u_i and v_i are not parallel (because P_i is a singular dot), we conclude that the directions v_{i-1} and v_i are also not parallel. ▪

Thus, for a given singular dot P_i, if $P_{\hat{\imath}}$ is the singular dot judged as characteristic dot just before P_i, then we are in front of one of the two following situations:

- First case: the angle between the direction v_i and one of the above directions v_j , $\hat{\imath} \leq j \leq i-1$, is a right angle. In this case, the dot P_i is a real cusp and will thereafter be considered as a characteristic dot of the digit D.
- Second case: the angle between the direction v_i and all previous directions v_j, $\hat{\imath} \leq j \leq i-1$, is a non-right angle (so it's equal to 135° or 180°). In this case, the dot P_i is a false cusp and should not be considered as a characteristic dot of the digit D.

The following algorithm shows how to detect the characteristic dots from the singular dots $(P_i)_{1\leq i\leq m}$:

Algorithm to detect the characteristic dots from the singular dots

```
The starting dot P1 is considered as a characteristic dot
i = 2
î =1
while i≤m-1
    for j=î:i-1
        if (the angle between the direction vi and the
              direction vj is a right angle)
              Pi is a characteristic dot
              î=î+1
              i=i+1
        else
              i=i+1
        endif
    endfor
endwhile
The last dot Pm is considered as a characteristic dot
```

So, the features of the digit D are:

1. the obtained characteristic dots $(Q_i)_{1\leq i\leq r}$ $(r \leq m)$,
2. the associated derivatives $(u_i, v_j)_{1\leq i\leq r}$, ($(u_i, v_j)$ are the two derivative directions of the dot Q_i),
3. the associated information $(e_i)_{1\leq i\leq r}$ identifying the characteristic points that belong to a loop (e_i=1 if the dot Q_i belongs to a loop and e_i=0 otherwise),
4. the class affiliation c, ($c \in LD, NLD$).

So, the features of the digit D are its class affiliation and the following characteristic matrix:

$$\begin{pmatrix} Q_1 & Q_2 & \dots & Q_r \\ u_1 & u_2 & \dots & u_r \\ v_1 & v_2 & \dots & v_r \\ e_1 & e_2 & \dots & e_r \end{pmatrix}$$

6. Training phase

The training phase consists in choosing some writers and characterizing the different digits written by these writers. Indeed, given n writers S_i, $1 \leq i \leq n$, and a digit D, we denote by M_i the characteristic matrix of the digit D written by the writer S_i and c_i its class affiliation. As the sizes of the matrices M_i aren't necessarily equal (because we don't have the same writer), a standardization of matrix sizes is necessary.

6.1 Standardization of matrix sizes

Put $n^*=max\ n_i$, where n_i, $1 \leq i \leq n$, is the number of columns of the matrix M_i (n_i = the number of characteristic dots of the digit D written by the writer S_i). Let M^* be one of the matrices M_i such that its number of columns is equal to n^*. The standardization approach of matrix sizes that we have adopted is to complete each matrix M_i for which $n_i < n^*$, by additional $(n^*- n_i)$ characteristic dots.

Let N^* (resp. N_i) be the sub matrix of M^* (resp. M_i) formed by the first two lines of M^* (resp. M_i) (it is obtained by keeping in M^* (resp. in M_i) only the coordinates of characteristic dots). First, we match each dot of the matrix N_i the nearest dot of the matrix N^* in the sense of Euclidean norm. So, it will remain (n^*-n_i) dots in N^* for which we have no counterpart in N_i. For each of these (n^*-n_i) dots of N^*, we seek the nearest dot of the digit D written by the writer S_i (see Fig. 9). By respecting the position of corresponding dot in the matrix M^*, we add to the matrix M_i this dot with its associated derivatives and information relating to its belonging or not to a loop. So we get a new matrix M_i^* with the same size as M^*.

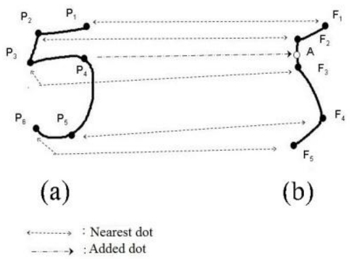

Fig. 9 Standardization of sizes applied to digit 5

6.2 Digit features

The characteristic matrices $(M_i^*)_i$ of the digit D according to the writers $(S_i)_i$ are all of the same size ($5\times n^*$). Thus, the features of the digit D are:

1. The characteristic matrix M_D equal to the mean of the matrices $(M_i^*)_i$:

$$M_D = \frac{1}{n}\sum_{i=1}^{n} M_i^*$$

2. The class affiliation c of the digit D equal to the value c_k which is the most common in the sequence $(c_i)_{1\leq i\leq n}$.

7. **Recognition**

By following the steps in the previous paragraph, we compute the features of each digit of classes LD and NLD. The recognition of an unknown digit D will occur in four steps.

7.1 Features of the digit D

We first compute the class affiliation and the characteristic matrix M_D of the digit D by following the steps developed in Paragraph 5. The next step consists to identify from characteristic matrices M_i of digits D_i belonging to the same class as D, the nearest matrix to M.

7.2 Distance between M_D and the characteristic matrix M_i of the digit D_i

Since the matrices M_D and M_i do not necessarily have the same size, we first standardize their sizes. For this, we distinguish three cases on their respective numbers of columns n_D and n_i.

7.2.1 First case: $n_D = n_i$

We ordain the columns of the matrix M_i so that the j^{th} column of the obtained matrix is the closest to the j^{th} column of the matrix M_D.

7.2.2 Second case: $n_D < n_i$

To complete the matrix M_D by $(n_i - n_D)$ additional dots, we proceed as in sub-paragraph 6.1.

7.2.3 Third case: $n_i < n_D$

We first ordain the columns of the matrix M_i so that the j^{th} column, for $j \leq n_i$, of the obtained matrix is the closest to the j^{th} column of the matrix M_D. As M_i is the characteristic matrix of D_i according to several writers (this is the mean of characteristic matrices $(M_{ij})_{1\leq j\leq r}$ of the digit D_i according to the r writers $(w_j)_{1\leq j\leq r}$ used in training phase), we first complete each matrix M_{ij} by $(n_D - n_i)$ additional dots by following the steps of the sub-paragraph 6.1. After, we substitute the matrix M_i by the matrix M_i^* mean of these matrices which is the same size as M_D.
After the standardization phase of matrix sizes, we denote $M_i^*$and M_D^* the obtained matrices, and we put $d_i = \| M_i^* - M_D^* \|$ where $\| \|$ is the Frobenius norm.

7.3 Identification of the digit D

The digit D will be identified with the digit D_i^* satisfying the following minimization equation:

$$d_{i^*} = \min_i d_i$$

The minimum is taken for all digits belonging to the same class as the digit D.

8. **Recognition Results**

The database DB consists of 360 digits. Each digit between 0 and 9 has been written by 36 different writers (see Fig. 10).

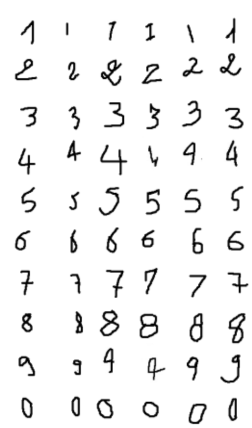

Fig. 10 Sample of handwritten digits.

One part of DB, denoted Tr_DB was used in the training phase, and the rest, denoted Te_DB was reserved to evaluate the system.
We sought to identify the best choice of the set Tr_DB giving the highest recognition accuracy in the test phase.

For this, we denote by S_r the r[th] writer and *RR* the recognition rate.

Given an integer $k \geq 1$, and for any combination of k writers among 36 writers, we first use these k writers as *Tr_DB* in the training phase, and after compute the corresponding recognition rate *RR*. Finally, we identify the combination of *k* writers giving the highest *RR*.

The results obtained for $k < 3$ and $k > 8$ are not interesting. So, we give in Table 1 only the results for $3 \leq k \leq 8$.

Table 1: Set *Tr_DB* of k writers giving the best recognition rate (RR)

K	Set Tr_DB of k writers giving the best RR	RR (%)
3	S_8 ; S_3 ; S_{16}	91.94
4	S_8 ; S_3 ; S_{16} ; S_{22}	94.44
5	S_8 ; S_3 ; S_{16} ; S_{22} ; S_{26}	96.39
6	S_8 ; S_3 ; S_{16} ; S_{22} ; S_{26} ; S_{14}	96.94
7	S_8 ; S_3 ; S_{16} ; S_{22} ; S_{26} ; S_{14}; S_{25}	90.00
8	S_8 ; S_3 ; S_{16} ; S_{22} ; S_{26} ; S_{14}; S_{25} ; S_{18}	86.67

The best performance has been achieved when we use the six writers S_8, S_3, S_{16}, S_{22}, S_{26} and S_{14} in the training phase. The explanation that we can advance on high performance obtained with this list is that the writing styles of these writers cover the different writing styles of all writers. Recognition errors are mainly due to writing styles of some writers. Indeed, the digits 1, 4, 7 and 9 are in some cases very confused, and even humans have difficulties to identify them (see Fig. 10).

For more details, we give in Table 2 the confusion matrix along with the recognition rate of each digit. These results are related to the use of the optimal list (S_8,S_3,S_{16},S_{22},S_{26},S_{14}) in the training phase.

Table 2: Confusion Matrix and the recognition rate (RR) of each digit

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**RR (%)**
0	36	0	0	0	0	0	0	0	0	0	100
1	0	34	0	0	0	0	0	**2**	0	0	94,44
2	0	0	36	0	0	0	0	0	0	0	100
3	0	0	0	36	0	0	0	0	0	0	100
4	0	**4**	0	0	30	0	0	0	0	**2**	83,33
5	0	0	0	0	0	36	0	0	0	0	100
6	0	0	0	0	0	0	36	0	0	0	100
7	0	0	0	0	0	0	0	36	0	0	100
8	0	0	0	0	0	0	0	0	36	0	100
9	0	0	0	0	**3**	0	0	0	0	33	91,66

9. **Conclusion**

We presented in this work a new approach of digit recognition. It is based on the extraction the Hermite data from the digit shape (dots with their derivatives). The choice of this approach was dictated by the possibility of recovering a close shape to that of the digit using the Bézier curve theory on these data.

The obtained results are very interesting, and we plan to improve them using other classifiers (the artificial neural networks, the hidden Markov models and the support vector machine) during both training and testing phases. Similarly, we will enrich our database in order to perform tests on a more consistent data base.

References

[1] C. L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, "Handwritten digit recognition: benchmarking of state-of-the-art techniques", Pattern Recognition, Vol. 36, 2003, pp. 2271–2285.

[2] S. Impedovo, F.M. Mangini and D. Barbuzzi, "A novel prototype generation technique for handwriting digit recognition", Pattern Recognition, Available online 3 May 2013 http://dx.doi.org/10.1016/j.patcog.2013.04.016

[3] O. Rashnoodi, A. Rashnoodi and A. Rashnoodi", Off-line Recognition of Persian Handwritten Digits using Statistical Concepts", International Journal of Computer Applications, Vol. 53, No. 8, 2012, pp. 20–28.

[4] M. Shi, Y. Fujisawa, T. Wakabayashi and F. Kimura, "Handwritten numeral recognition using gradient and curvature of gray scale image", Pattern Recognition, Vol. 35, No. 10, 2002, pp. 2051–2059.

[5] A. Mazroui, Aissa Kerkour El Miad, "*Handwritten Arabic characters modeling by Bézier curves for recognition*", in proceeding of 4[th] International Conference on Approximation Methods and Numerical Modelling in Environment and Natural Resources, 2011, pp. 535-538.

[6] G. Farin, and P. Massart, Curves and Surfaces for Computer Aided Geometric Design, San Diego, CA, Academic Press, 1983.

[7] S. Arora, D. Bhattacharjee, M. Nasipuri, L. Malik, M. Kundu and D. K. Basu, "Performance Comparison of SVM and ANN for handwritten Devnagari Character Recognition", International Journal of Computer Science Issues, Vol. 7, No. 6, 2010, pp. 18-26.

[8] X. X. Niu, and C. Y. Suen "A novel hybrid CNN–SVM classifier for recognizing handwritten digits", Pattern Recognition, Vol. 45, No. 4, 2012, pp. 1318–1325.

[9] S. Mahmoud, "Recognition of writer-independent off-line handwritten Arabic (Indian) numerals using hidden Markov models", Signal Processing, Vol. 88, No. 4, 2008, pp. 844–857.

[10] D. Sharma, and D. Gupta, "Isolated Handwritten Digit Recognition using Adaptive Unsupervised Incremental Learning Technique", International Journal of Computer Applications, Vol. 7, No. 4, 2010, pp. 27-33.

[11] K. Mori, M. Matsugu, and T. Suzuki, "Face recognition using SVM fed with intermediate output of CNN for face detection", in Proc. of the IAPR Conf. on Machine Vision Applications (MVA), 2005, pp. 410–413.

[12] M. Szarvas et al., "Pedestrian detection with convolutional neural networks", in Proc. of the IEEE Symp. On Intell. Vehicles (IV), 2005, pp. 224–229.

[13] L.M. Lorigo, and V. Govindaraju, "Offline Arabic handwriting Recognition: A survey", IEEE Trans. Pattern Anal. Machine Intell., Vol. 28, 2006, pp.712-724.

[14] A.M. AL-Shatnawi, S. AL-Salaimeh, F. H. AL-Zawaideh, and O. Khairuddin, "Offline Arabic Text Recognition-An

Overview", World of Computer Science and Information Technology Journal (WCSIT), Vol. 1, No. 5, 2011, pp. 184-192.
[15] A. Mesleh, A. Sharadqh, A. Al-Azzeh, J. Abu-Zaher, M. Al-Zabin, N. Jaber,and T. Odeh, Hasn, "An optical character recognition", Contemporary Engineering Sciences, Vol. 5, No. 11, 2012, pp.521-529.
[16] Y. Y. Zhang, P. S. P. Wang, "A New Parallel Thinning Methodology", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 8, 1994, pp 999-1011.