The use of Radon Transform in Handwritten Arabic (Indian) Numerals Recognition

SABRI A. MAHMOUD[1], MARWAN H. ABU-AMARA[2]
[1]Information and Computer Science, [2]Computer Engineering Department
King Fahd University of Petroleum and Minerals, Dhahran,
SAUDI ARABIA
smasaad@kfupm.edu.sa http://faculty.kfupm.edu.sa/ICS/smasaad/
marwan@kfupm.edu.sa http://faculty.kfupm.edu.sa/COE/marwan/

Abstract: - This paper describes a technique for the recognition of off-line handwritten Arabic (Indian) numerals using Radon and Fourier Transforms. Radon-Fourier-based features are used to represent Arabic digits. Nearest Mean Classifier (NMC), K-Nearest Neighbor Classifier (K-NNC), and Hidden Markov Models Classifier (HMMC) are used. Analysis using different number of projections, varying the number of Radon-based features, and the number of samples used in the training and testing of this technique is presented using the NMC and K-NNC.

A database of 44 writers with 48 samples per digit each totaling 21120 samples are used for training and testing of this technique. The training and testing of the HMMC is different than that of the NMC and K-NNC in its internal working and in the way data is presented to the classifier. Since the digits have equal probability the randomization of the digits is necessary in the training of the HMMC. 80% of the data was used in training and the remaining 20% in testing of the HMMC.

Radon-based features are extracted from Arabic numerals and used in training and testing of the HMM. In this work we didn't follow the general trend, in HMMC, of using sliding windows in the direction of the writing line to generate features. Instead we generated features based on the digit as a unit. Several experiments were conducted for estimating the suitable number of states for the HMM. In addition, we experimented with different number of observations per digit. The Radon-Fourier-based features proved to be simple and effective.

The classification errors were analyzed. The majority of errors were due to the misclassification of digit 7 with 8 and vice versa. Hence, a second Structural Classifier is used in a cascaded (second) stage for the NMC, K-NNC, and HMMC. This stage, which is based on the structural attributes of the digits, enhanced the average overall recognition rate from 3.1% to 4.05% (Recognition rates of 98.66%, 98.33%, 97.1% for NMC, K-NNC, HMMC, respectively).

Key-Words: - Arabic numeral recognition, OCR, Hidden Markov Models, Handwritten Digit recognition, Nearest neighbor classifier.

1 Introduction

The potential application areas of automatic reading machines are numerous. One of the earliest and most successful application is sorting checks in banks, as the volume of checks that circulates daily has proven to be too enormous for manual entry [1]. Handwritten digit recognition is a vital component in many applications; office automation, check verification, and a large variety of banking, business, postal address reading, sorting and reading handwritten and printed postal codes and data entry applications are few examples.

Fig. 1 shows the Arabic (Indian) numerals. In fact Arabic numbers are used by Latin-based languages and Arabs use Indian numbers. Farsi and Urdu use similar numbers to Arabic (Indian) numbers. Although Arabic text is written right to left, Arabic numbers are written left to right with most significant digit being the left most one and the least significant digit is the right most one.

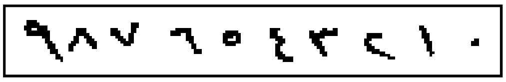


Fig. 1 Arabic (Indian) numerals.

Various methods have been proposed and high recognition rates are reported for the recognition of English handwritten digits [2-6]. In recent years many researchers addressed the recognition of Arabic text including Arabic (Indian) numerals [7-

13]. Surveys on Arabic Optical Text Recognition may be cited in [14-16]. Several researchers reported the recognition of Persian (Arabic) handwritten digits. However, the reported recognition rates need more improvements to be practical [17-21].

Al-Omari presented a recognition system for Indian numeral digits using average template matching approaches [7]. Freehand sketches of online numeric digits placed on an image template were processed to extract a key feature vector representing significant boundary point distances from the digit center of gravity (COG). A model for each numeric digit is formed by processing 30 handwritten digit samples. Classification was made using the Euclidean distance between the feature vector of the test samples and the models. In another work Al-Omari et al. presented a recognition system for online handwritten Indian numerals one to nine. The system skeletonizes the digits then geometrical features of the skeleton of the digits are extracted. Probabilistic neural networks (PNNs) are used for classification. The developed system is translation-, rotation-, and scaling-invariant. The authors' claim that the system may be extended to address Arabic characters [8]. Bouslama presented an algorithm based on structural techniques for extracting local features from the geometric and topological properties of online Arabic characters using fuzzy logic [9]. Salah et al. developed a serial model for visual digit classification based on primitive selective attention mechanism [10]. The technique is based on parallel scanning of a down sampled image to find interesting locations through a saliency map, and by extracting key features at those locations at high resolution.

Shahrezea et al. used the shadow coding method for recognition of Persian handwritten digits. In this method, a segment mask is overlaid on the digit image and the features are calculated by projecting the image pixels into these segments [17]. In [18] the Persian digit images are represented by line segments which are used to model and recognize the digits. Additional features and classifier are needed for discriminating the digit pairs ‘‘0–5’’, ‘‘7–8’’, ‘‘4–6’’. Said et al. fed the pixels of the normalized digit image as is into a neural network for classification, where the number of the hidden units for the neural network classifier is determined dynamically [19]. Sadri et al. used a feature vector of length 16 which is estimated from the derivative of the horizontal and vertical profiles of the image [20]. [21] used the normalized image profile calculated at multiple orientations as the main feature for the recognition of Persian handwritten digits. The crossing counts and projection histogram calculated at multiple orientations are used as complementary features. The authors indicated that most of the system errors occurred in discriminating the digits ‘‘2’’, ‘‘3’’, ‘‘4’’ and ‘‘0’’, ‘‘5’’. Hence, discriminating these digits requires the use of additional features and may require the use of additional classifiers.

In recent years Radon transform has received much attention [22-26]. The Radon transform is used to detect features within an image. This transform is able to transform lines in a two dimensional image into a domain of possible line parameters, where each line in the image gives a peak positioned at the corresponding line parameters. This is similar to the Hough transform [25]. The discrete Radon transform (DRT), which is considered to be the discretization of the continuous Radon transform [22], has been generalized in [26] and adopted in widespread applications, such as computed tomography, magnetic resonance imaging, remote sensing, image and character recognition, detection of curves, motion detection, geophysics, etc. [27-30].

Toft used the generalized Radon transform to detect curves in noisy images [30]. The information of the curves of the images was separated from noise information in the parameter domain. Li used Radon transform for object recognition [29]. [30] used Radon and Fourier transforms for rotation, scale, and translation invariant water marking. In this work we are using Radon and Fourier transforms for Arabic numeral recognition.

Several research papers are published using Hidden Markov Models (HMM) for text and digit recognition [11, 31-34]. In order to use HMMs several researchers computed the feature vectors as a function of an independent variable. This simulates the use of HMM in speech recognition where sliding frames/windows are used. The same technique is used in off-line text recognition where the independent variable is in the direction of the line length [31, 32]. In this paper we are using different technique to extract the features of an Arabic numeral using the numeral as a whole and not a sliding window that calculates the features based on partial parts of the character. However, we are using the same HMM classifier without modification.

In this paper, we present a simple, effective, and scalable technique for the recognition of offline handwritten Indian numerals (0,1,…,9) used in Arabic writing. The presented technique is implemented using the Nearest Mean (NMC), K-

Nearest Neighbor (KNNC), and HMMC classifiers. The results of the three classifiers are analyzed and compared. To the authors knowledge this is the first time Radon and Fourier transforms are used for Arabic character/digit recognition. It is also expected to be the first time to use HMM with Radon-Fourier features. To improve the recognition rate a second structural classifier is cascaded with the NMC, KNNC, and HMM classifiers. The introduced structural classifier for digits 7 and 8 proved very effective in increasing the over all recognition rate by over 3%. Fig. 2 shows the model used for combining the feature extraction and classification using the three classifiers in the first stage and the structural classifier in the second stage.

An Arabic number may consist of an arbitrary number of digits. The recognition system performs classification on each digit independently preserving its relative position with respect to other digits in order to obtain the actual value of the number after recognition. The developed recognition system has enough flexibility to treat variations, line thickness, writing size, and translation of the handwritten string. The left to right position order of each digit is preserved to account for the digit weight after individual digit classification.

This paper is organized as follows. Radon transform is presented in Section 2. Feature extraction is addressed in Section 3. Training, recognition, and experimental results are addressed in Section 4, and finally the conclusions are presented in Section 5.

2 Radon-Fourier-based Features

The Radon transform computes projections of an image along specified directions. A projection of a two-dimensional function f(x,y) is a set of line integrals. The radon transform computes the line integrals from multiple sources along parallel paths in a certain direction. The paths are spaced 1 pixel unit apart. To represent an image, the radon transform takes multiple, parallel- projections of the image from different angles. Fig. 3 shows a single projection at a specified rotation angle. For example, the line integral of f(x,y) in the vertical direction is the projection of f(x,y) onto the x-axis; and the line integral in the horizontal direction is the projection of f(x,y) onto the y-axis.

The integral along a line through the image f(x,y) is given by,

$$r(\theta,\rho) = \int_s f(x,y)\, ds \tag{1}$$

Where $r(\theta, \rho)$ is the line integral (1-dimensional projection) of the image intensity, f(x,y), along a line s that is distance ρ from the origin and at angle θ off the x-axis.

All the points on this line satisfy the equation:

$$\rho = x \sin(\theta) - y \cos(\theta) \tag{2}$$

Therefore, the projection function $r(\theta,\rho)$ can be rewritten as

$$r(\theta,\rho) = \iint f(x,y)\delta(x\sin\theta - y\cos\theta - \rho)\, dx\, dy \tag{3}$$

The collection of these $r(\theta,\rho)$ at all θ is called the Radon Transform of the image f(x, y). Fig. 4 shows the projections of digit 3 at 0, 45, 90, and 135 degrees.

Based on the Fourier Slice Theorem, the 1-dimensional Fourier Transform of the projection function $r(\theta,\rho)$ is equal to the 2-dimensional Fourier Transform of the image evaluated on the line that the projection was taken on (the line that $r(\theta,0)$ was calculated at). So we can simply take the 2-dimensional inverse Fourier Transform and have our original image.

The 1-dimensional Fourier Transform of r is given by:

$$R(\theta,\omega) = \int e^{-j\omega\rho}\, r(\theta,\rho) d\rho \tag{4}$$

Substitute the expression for $r(\theta, \rho)$ of Equation (3) in Equation (4) we have

$$R(\theta,\omega) = \iiint f(x,y)\, e^{-j\omega\rho}\, \delta(x\sin\theta - y\cos\theta - \rho)\, dx\, dy \tag{5}$$

Using the sifting property of the Dirac delta function we have

$$R(\theta,\omega) = \iint f(x,y)\, e^{-j\omega(x\sin\theta - y\cos\theta - \rho)}\, dxdy \tag{6}$$

Transforming Equation (6) to the discrete case we have,

$$R[\theta,k] = \sum_0^{N-1}\sum_0^{M-1} f[n,m]\, e^{-j2\pi k(n\sin\theta - m\cos\theta - \rho)} \tag{7}$$

Where $R[\theta,k]$ is the 1-d discrete Fourier transform of the discrete Radon projections at the discrete angle θ at which the projections are taken, k is the discrete Fourier frequency, N,M is the width and height of the image f, respectively; ρ is the discrete distance of the projection line from the image origin. The amplitude of the Fourier spectrum is computed by:

$$R[\theta,k] = \sqrt{R_r(\theta,k)^2 + R_i(\theta,k)^2}\ , \qquad (8)$$

where Rr, Ri are the real and imaginary parts of the Fourier spectrum for one projection at angle θ.

The Fourier coefficients are divided by |R [θ, 0]| to make them scale invariant. The extracted features of R[θ, k] at the different projection angles are used to represent Arabic digits in the training and testing of our classifiers. Fig. 5 shows the Radon transform of digits 3 and 5 and the respective Fourier transform of the Radon projections.

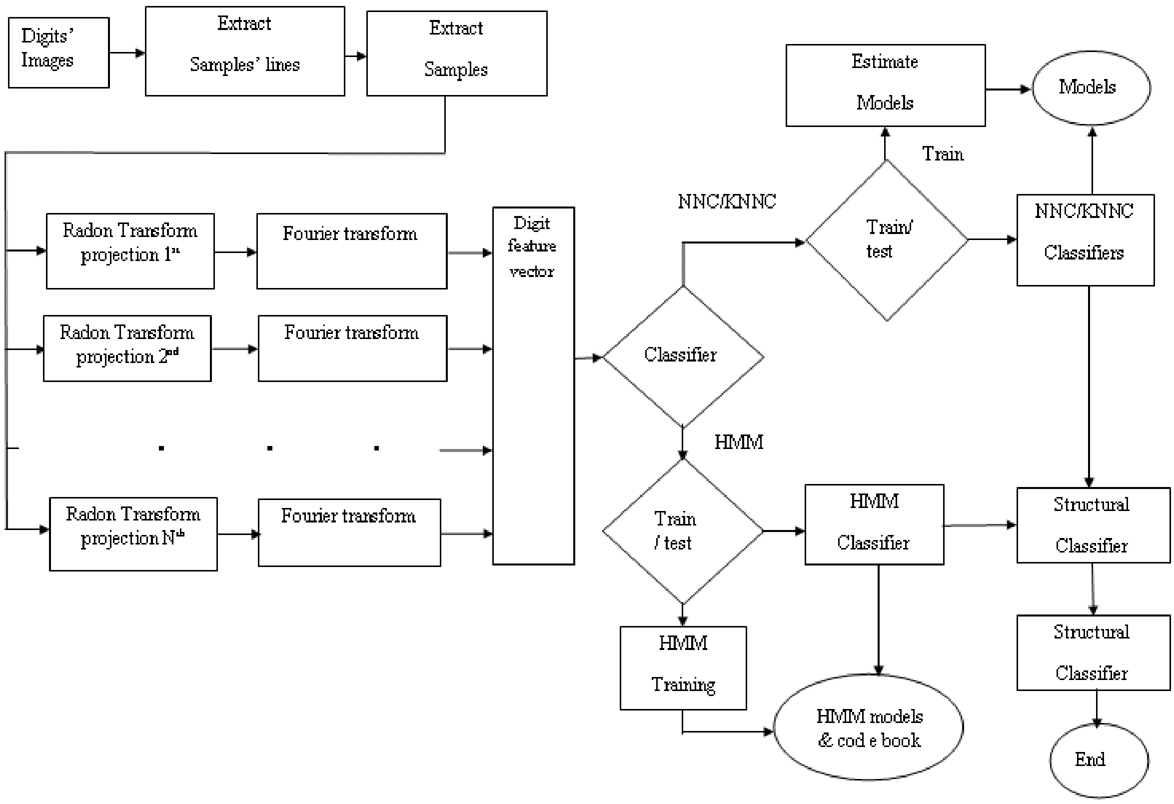


Fig. 2 Model for the implemented Arabic (Indian) Numeral Recognition System

3 Feature Extraction

To extract the features of Arabic (Indian) digits the Radon transform was applied to the Arabic digit image at different projection angles using the discrete version of Equation 3. A number of experiments were carried out to find out the suitable angle increment between the image projections. For each projection of the image the Fourier transform using Equation 7 is applied to the Radon domain. The amplitude of the 1-dimensional Fourier spectrum is computed using Equation 8. The Fourier spectrum coefficients from the different projections are used as the features of Arabic digits.

An offline handwritten numbers are scanned using 300 dpi scanner. The scanned image is horizontally projected and writing lines' boundaries are estimated using black and white regions in the horizontal projection and then lines are extracted. Each writing line is projected vertically and digits are extracted using the same technique for extracting writing lines. Radon transform is applied to each digit followed by the application of the Fourier transform to each Radon projection. Features are extracted from the amplitude of the Fourier spectrum. The features of the different projections are concatenated and used to represent the digit. The system includes training and testing phases. The details of these phases are addressed in the next section. The digits classification is done in two stages. In the first stage the NMC, the K-NNC, or the HMMC is applied. In the second stage, the structural classifier is applied after the first stage classifier.

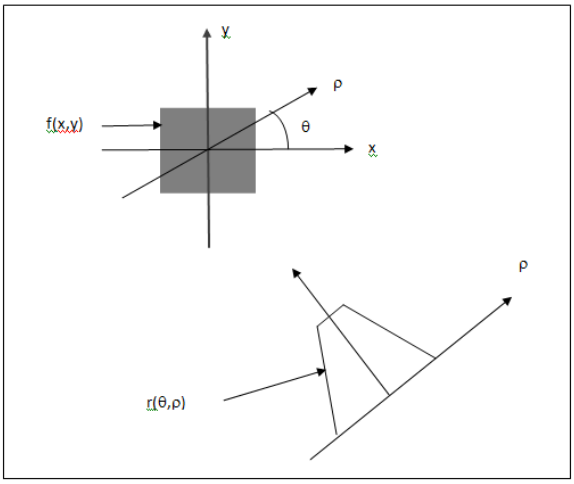


Fig. 3 Radon transform of a two-dimensional function f(x,y)

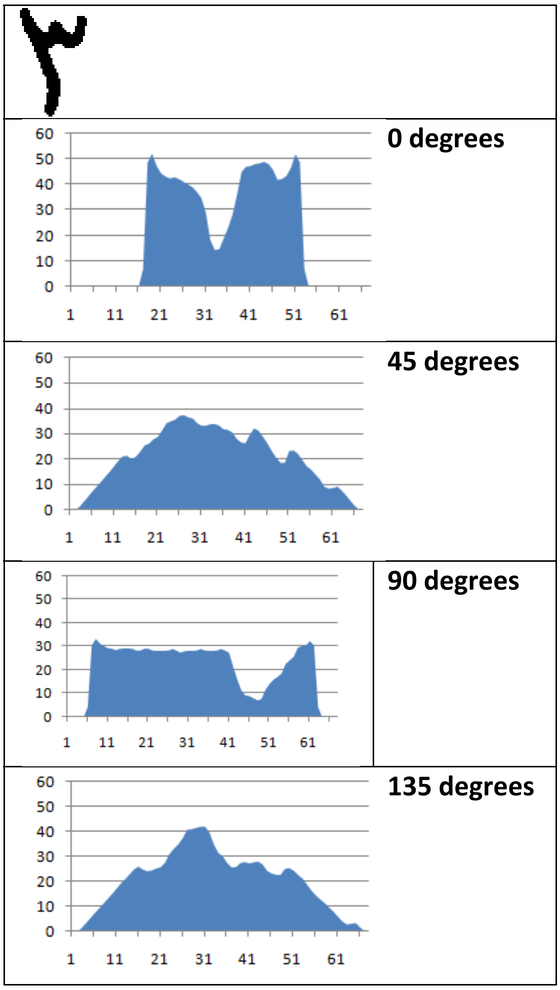


Fig. 4 Digits 3 Radon projections at 0, 45, 90, and 135 degrees.

4 Training and Recognition

In this paper Radon-Fourier transforms based features and the NMC/K-NNC classifiers are used for experimenting with the different number of used features. A number of experiments are carried out to find the most suitable number of projections, the most suitable number of Radon-Fourier-based features, and the different sizes of training and testing data sets. Then the selected features are used in NMC and K-NNC with HMMC classifier for comparison purposes. The results of the three classifiers are then compared.

In general, in the training phase the features of training data are computed and saved as models for the trained classes. In the recognition phase an unknown character features are extracted and compared with the features of the models. The unknown character is labeled to the model label whose features are the closest (or the most probable) to the unknown character.

The used database consists of 21120 samples written by 44 writers, each writer wrote 48 samples per digit. The written pages are then scanned using a scanner with a resolution of 300 pixels per inch. The scanned document images are transformed into binary images (viz. black and white). Reference may be made to [34] for more details on the used database.

The implementation of this work was done using C language and MATLAB. The HTK tools [35] are used in the experimentation of HMMC.

4.1 The NMC/K-NNC Classifiers analysis

In this work, the Radon-Fourier transforms are tested for suitability to perform pattern recognition of hand-written Arabic/Hindi digits. The experimentations are conducted on a total of 44 hand-written sets of digits. We conducted several experiments using different number of samples for training/modeling and testing. In the training phase, using NMC and K-NNC, the feature vectors (V) of the training data are extracted. The features of each digit of each writer are averaged and the averaged features are used as the models of the Arabic numerals.

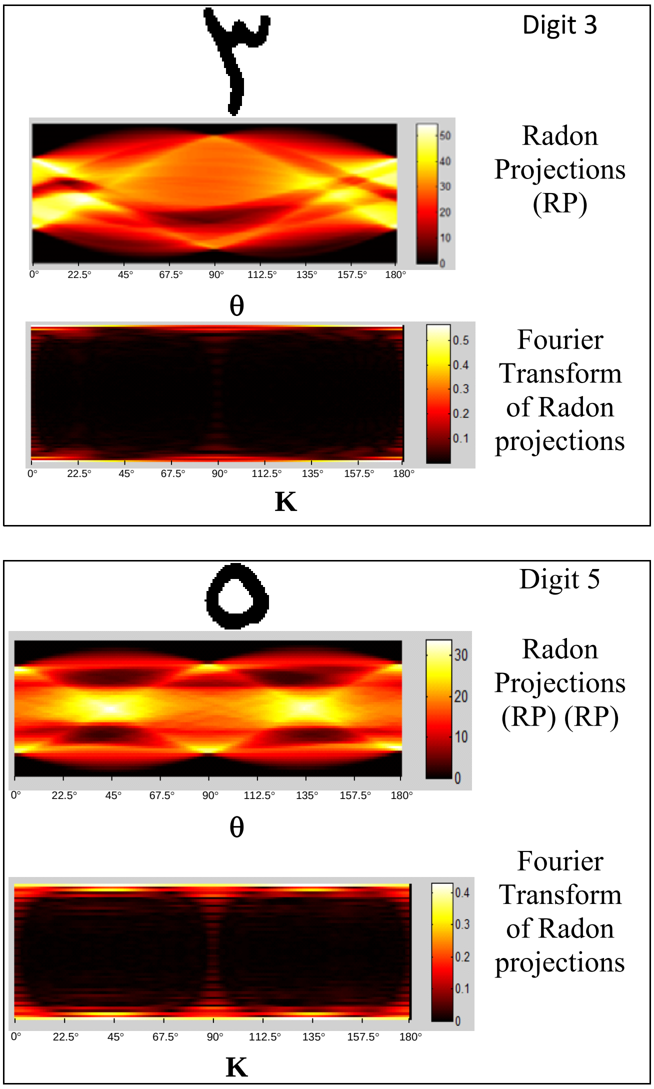


Fig. 5 Radon projections and the corresponding Fourier Transform of the Radon projections of Digits 3 and 5

In the testing phase the feature vector (V) for the unknown character is computed and then compared with the feature vectors of the model classes. The classification decision is based on the nearest mean classification method in the case of NMC and 5-NN in the case of K-NNC. The nearest distance is computed using Euclidean distance given by the following Equation (9):

$$E_{ri} = \sum_{j=1}^{m} [\left| M_{rij} - V_j]^2 \right| \tag{9}$$

where E_{ri} is the distance between the input digit and model r_i (i.e sum of the Euclidean distances between the features of the input digit and those of model r_i), m is the total number of parameters in the feature vector, M_{rij} is the jth feature of model ri, Vj is feature j of the input digit feature vector, i is index for digits 0,1,..9, and r writers from 1 to 44.

.

NMC case: The distance (E_{ri}) between the new digit and all models' feature vectors are found. The argument of the minimum value found, i.e. argmin(E_{ri}), yields the recognized model *i*. This model is considered as the class that matches most closely the obtained features vector of the unknown digit. In the training phase we average all the features of each digit for each writer. Hence, we have 44 models of each digit. The nearest mean classifier finds the model that is nearest the sample.

K-NNC: The distance (E_{ri}) between the new digit and all models' feature vectors are found. We select the 5 models closest to the input digit (digit under test). We take majority voting of the five neighbors. If at least three of the selected models have the same digit label then the input digit is assigned that label otherwise the digit is rejected.

In order to find the best number of Radon-based descriptors (the one that results in the highest average recognition rate) several experiments are conducted using from 1 to 35 descriptors per projection. We used 180 projections with one degree increment. 36 samples of each writer are used for training and the remaining samples are used for testing. The results of these experiments are shown in Fig. 6. It is clear from the figure that the average recognition rate steadily increases with the increase of the number of used descriptors until reaching 10 descriptors. Then it stays nearly the same until 35 descriptors. Hence, we decided to use 10 descriptors in our experimentation as it requires less computation with no loss in recognition rate. In order to find the best number of projections required we run experiments using different angle increment between projections. Comparable results are achieved using 36 projections at 5 degree increment to the results of using 180 projections at 1 degree increment. Hence in the following experiments 36 projections are used.

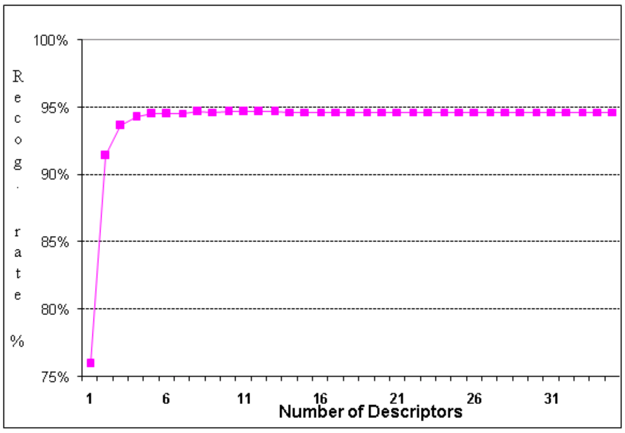


Fig.6 The recognition rate using 1 to 35 descriptors per digit with 180 Radon projections, 36 samples of each writer are used for training and 12 samples of each writer are used for testing of the 1st and 2nd stages.

The confusion matrix using 5-NNC with 10 descriptors, 36 Radon projections with 5 degree increment, 36 samples of each writer for training and 12 samples for testing is shown in Table 1. The table gives also the average recognition (c), the rejection, and error (e) rates. It is interesting to note that only 5 digits out of over five thousand digits, used in testing, are rejected as shown in Table 1. We reject a digit if 5-NNC do not have majority (i.e. less than 3 NN have the same digit label).

Table 2 shows the confusion matrix using NMC classifier, 36 projections with 10 descriptors per projection. It is clear from the confusion matrices of Tables 1 and 2 that NMC classified has an average recognition rate improvement over the 5-NN classifier by 0.59%. Fig. 7 shows the recognition rates of each digit using NMC and 5-NNC classifier. In general the recognition rate of NMC classifier is better or similar to that of 5-NN except for digit 8 where the 5-NN is better by less than 1% as shown in Fig. 7.

Fig. 8 shows the recognition rate for each writer using NMC and K-NNC classifiers. It is clear from the figure that the recognition rate for some writers is high (99.17% for writer 15 in K-NNC and for writers 11 and 16 in NMC) and other writers have low recognition rate (the lowest is 87.5% for writers 21 in NMC and writers 25 and 34 I K-NNC). This is normal as some writers differ in their writing style from the general trend of other writers. Fig. 9 shows samples of writers 11 and 21 that show the writing styles of the two writers resulting in the difference in the recognition rates.

4.2 Structural Classifier (SC) analysis

As evident from the results of the recognition rates based on the digits shown in Tables 1 and 2, the lowest 2 results are for digits 7 and 8. Also, it is observed that, on average, approximately 98% of the classification errors of digits 7 and 8 are due to misclassifying 7 as 8 and vice versa. On analyzing the images of digits 7 and 8 and the projections of these digits it is clear that the projections of these digits are similar. This explains the reason for having the high misclassification errors between digits 7 and 8. Digits 7 and 8 are similar; rotating one by 180 degrees gives the other. In order to address these high misclassification errors of digits 7 and 8 a second stage classification is added. A Structural Classifier (SC) is added to reclassify the 7 or 8 digits of the first classifier output. The algorithm of Fig. 10 is applied to discriminate between digits 7 and 8 in the second stage.

Table 3 shows the recognition rates of digits 0 to 9 using NMC classifier in the first stage and the Structural Classifier (SC) in the second stage. As evident from the results, the overall percentage of correctly identified digits has improved (from 94.87% to 98.66% for NMC classifier) due to improving the classification of digits 7 and 8. Similarly, a recognition rate improvement of 4.05% in the case of 5-NNC (from 94.28% to 98.33%) is observed.

The recognition rates of the writers using NMC and 5-NNC in the first stage and SC in the second stage of classification using 10 descriptors per digit, 36 Radon projections, using 36 samples /writer for training and 12 samples /writer for testing are shown in Fig. 11. The results for the same case before the structural classifier are also included for comparison purpose. The average recognition rate for all writers is 98.66%, an improvement of 3.79% using NMC classifier and average recognition rate for all writers is 98.33% an improvement of 4.05 in the case of K-NNC classifier as indicated earlier.

Table 1 The confusion matrix using 5-NNC classifier with 36 projections with 10 descriptors per projection.

Digit	0	1	2	3	4	5	6	7	8	9	Reject	%c	%reject	%e
0	524	0	0	0	0	0	0	0	0	3	1	99.24%	0.19%	0.57%
1	0	520	0	0	2	0	0	1	0	4	1	98.48%	0.19%	1.33%
2	3	0	521	2	0	1	0	0	0	0	1	98.67%	0.19%	1.14%
3	0	0	1	518	2	0	0	3	0	3	1	98.11%	0.19%	1.70%
4	1	1	7	0	502	1	12	0	0	4	0	95.08%	0.00%	4.92%
5	3	0	0	0	0	525	0	0	0	0	0	99.43%	0.00%	0.57%
6	0	0	0	0	2	0	525	0	0	1	0	99.43%	0.00%	0.57%
7	1	0	0	0	0	0	0	384	142	0	1	72.73%	0.19%	27.08%
8	3	0	0	0	0	0	0	83	442	0	0	83.71%	0.00%	16.29%
9	4	1	0	0	3	1	1	1	0	517	0	97.92%	0.00%	2.08%
												94.28	0.095	5.63%

Table 2 The confusion matrix using NMC classifier with 36 projections with 10 descriptors per projection

Digit	0	1	2	3	4	5	6	7	8	9	%c	%e
0	525	0	0	0	0	0	0	1	0	2	99.43%	0.57%
1	0	521	2	0	0	0	0	0	0	5	98.67%	1.33%
2	3	0	522	2	0	1	0	0	0	0	98.86%	1.14%
3	0	0	1	521	1	0	0	3	0	2	98.67%	1.33%
4	1	1	8	0	506	0	10	0	0	2	95.83%	4.17%
5	2	0	0	0	0	526	0	0	0	0	99.62%	0.38%
6	1	0	0	0	2	0	525	0	0	0	99.43%	0.57%
7	0	0	0	0	0	0	0	405	123	0	76.70%	23.30%
8	3	0	0	0	0	0	0	88	437	0	82.77%	17.23%
9	3	1	0	0	2	1	0	0	0	521	98.67%	1.33%
											94.87%	5.13%

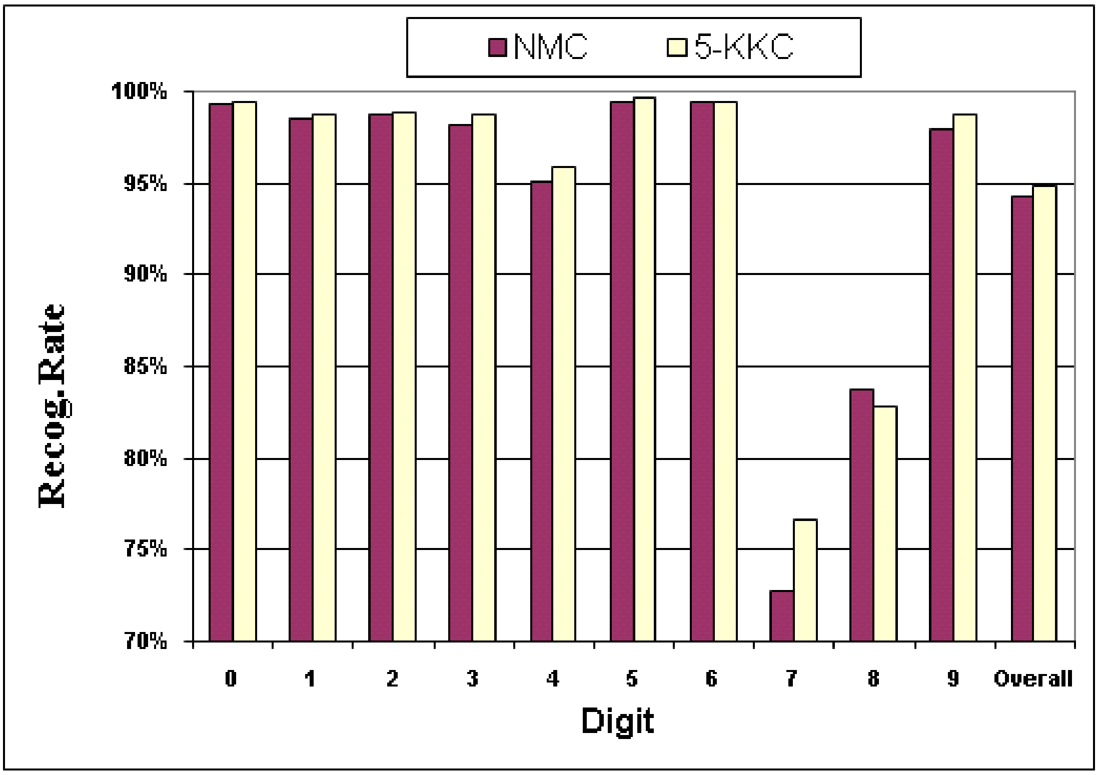


Fig. 7 the average recognition rates of digits 0 to 9 using NMC and 5-NNC classifiers.

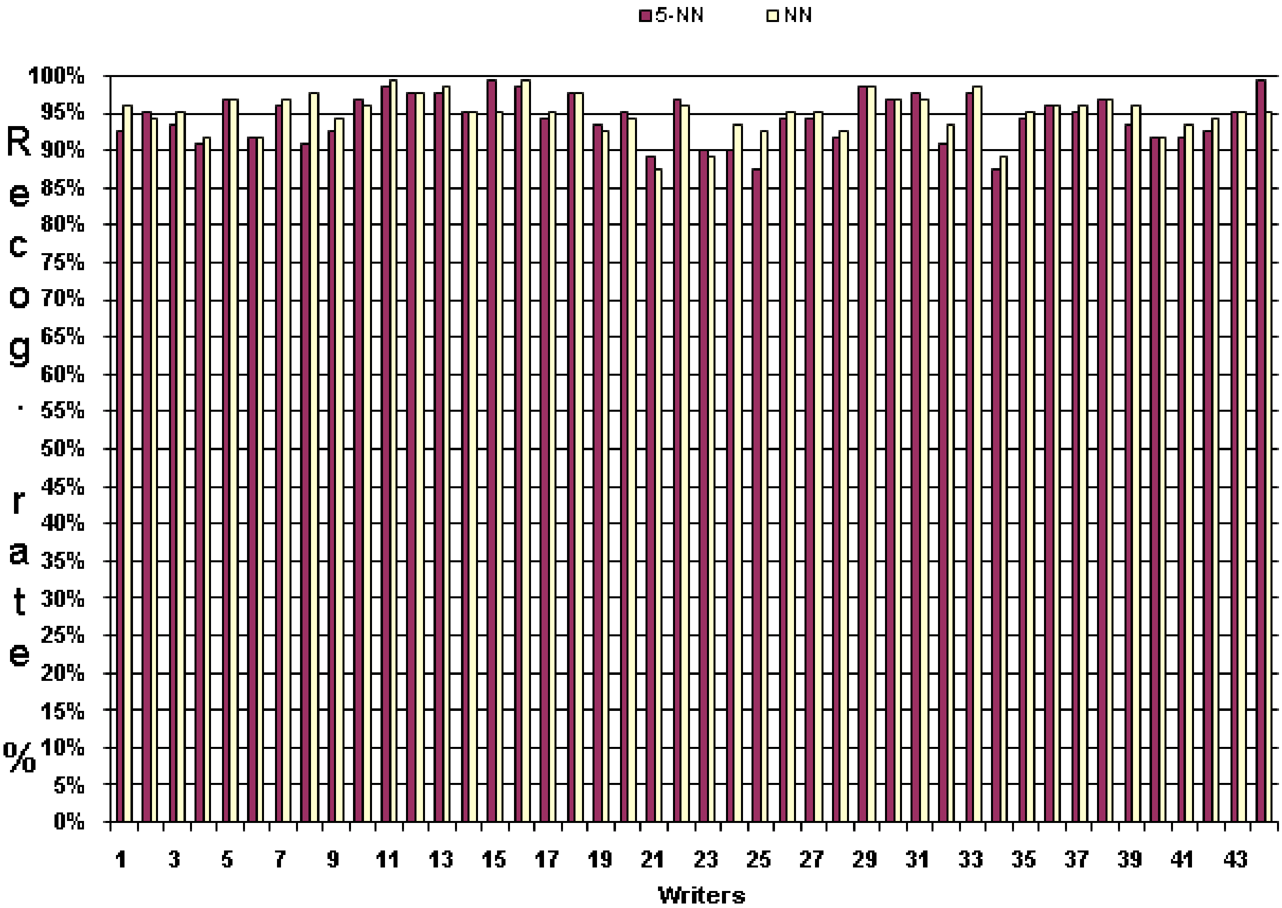


Fig. 8 The recognition rate for the writers using 5-NNC and NMC classifiers using 36 Radon projections with 10 descriptors per projection, 36 samples of each writer are used for training and 12 samples of each writer are used for testing

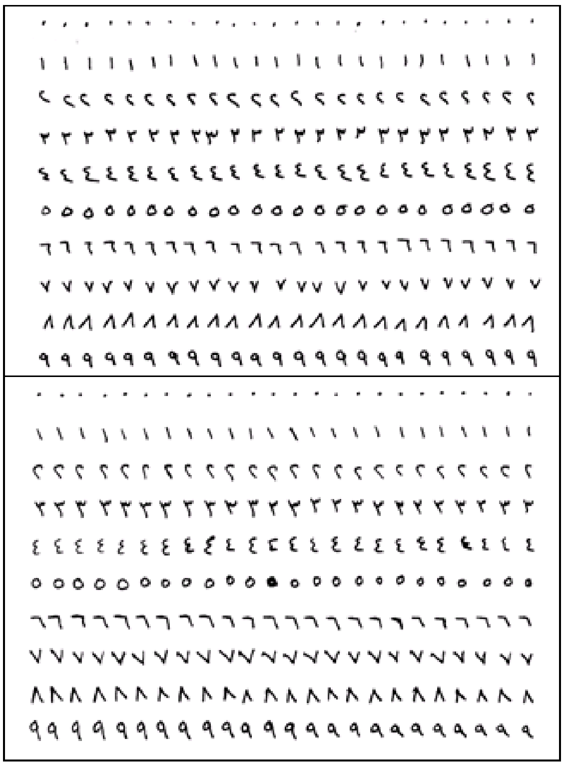


Fig. 9 Samplers of Writer 11 with highest recognition rate of 99.17% (top) and Writer 34 with lowest recognition rate of 87.5% (bottom).

Structural Classifier (SC) Algorithm

The digit image is scanned horizontally at half the height and vertically at half the width. The number and length of horizontal and vertical black segments are estimated.

The average length of the horizontal segments SHavg and the average length of vertical segments SVavg are estimated. The digit average length of the black segments (Savg) is taken as min (SHavg, SVavg).

The digit is then scanned horizontally at intervals Savg/2 from right to left.

The vertical location of the first

two horizontal segments encountered is found and the locations of the center of the two horizontal segments are estimated (SF1c, SF2c).
Scanning continues until the last two segments are encountered. The vertical location of the last two horizontal segments encountered is found and the locations of the center of the last two horizontal segments are estimated (SL1c, SL2c).
a. If the distance between the first two segments (i.e. SF2c - SF1c) is greater than the distance between the last two segments (i.e. SL2c - SL1c) then scan the digit vertically at (SF1c + SF2c)/2. The number and location of the segments are estimated.
If only one segment is encountered and the vertical location of the segment is below the vertical location of the first encountered segments then the digit is 7.
b. Else if the distance between the first two segments (i.e. SF2c - SF1c) is less than the distance between the last two segments (i.e. SL2c - SL1c) then scan the digit vertically at (SL1c + SL2c)/2. The number and location of segments is estimated.
If only one segment is encountered and the vertical location of the segment is above the vertical location of the second encountered segments then the digit is 8.

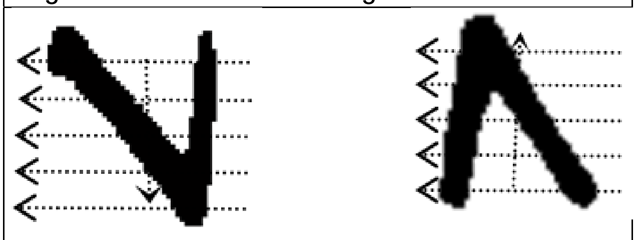


Fig. 10 Structural classifier Algorithm

4.3 Hidden Markov Model Classifier (HMMC)

In this paper a left to right HMM are used for our offline Arabic handwritten numeral recognition Fig. 12 shows the case of a 5-state HMM. This is in line of several research works using HMM [31, 32]. This model allows relatively large variations in horizontal position of the Arabic numeral. The sequence of state transition in the training and testing of the model is related to each digit feature observations. In this work we experimented with using different number of states and number of observations per digit and selected the best performing ones. Although each digit model could have different number of states we decided to use the same number of states for all digits as was done in [31, 32].

Each Arabic numeral is represented by 360-dimensional feature vector (viz. 36 projections at angle increment of 5 degrees, each with 10 Fourier coefficient features per projection). Each numeral requires a number of observations to train and test the HMMC. So, in one experiment our feature vector is divided into 36 separate sub vectors of 10 features each. In order to find the best number of states to use in the recognition and classification stages several runs are conducted on the data with different number of states from 4 to 38. Note that 38 is the maximum number of states that can be reached using 36 observations per digit. The HMMC is trained with the observations of 80% of the data (of 21120 samples) and tested the HMMC with the observations of the remaining 20% of the data. In these experiments code books of 128 and 256 are used. A codebook of 256 entries has better correction and accuracy results than the book of 128 entries. Fig. 13 shows summary of the results of the tested characters (the correctness and accuracy) using 4 to 38 states. It is clear from the figure that the recognition rate increases with the increase of the number of states. The results at 4 states show that the model did not have enough states to give acceptable results. The results continued to improve with adding more states until a maximum of 18 states is reached. The accuracy and recognition rates are fluctuating after that, at some states it increase and then reduced at other states.

Table 4 shows the correctness and accuracy at states 4,10,18,38 with codebooks of 128 and 256. It shows also the mean and variance of the results of states 10 to 38. As expected, the results show that, in general, a 256 book size is better than 128. Although there are a number of states that give local maximum results, the results of the other nearby states have an accuracy and correctness within 0.6% of the local maxima.

Other experiments are run using different number of observations. In these experiments the use of 36 observations, 10 features each are compared with 18 observations with 20 features each. Using 36 observations with 10 features each per digit produced slightly better recognition rates (1%) than using 18 observations with 20 features each per digit. Hence, 36 observations of 10 features each are used in our experimentations.

Table 3 The recognition rates of digits 0 to 9 using NMC classifier in the first stage and the Structural Classifier (SC) in the second stage.

Digit	0	1	2	3	4	5	6	7	8	9	%c	%e
0	**525**	0	0	0	0	0	0	1	0	2	***99.43%***	***0.57%***
1	0	**521**	2	0	0	0	0	0	0	5	***98.67%***	***1.33%***
2	3	0	**522**	2	0	1	0	0	0	0	***98.86%***	***1.14%***
3	0	0	1	**521**	1	0	0	3	0	2	***98.67%***	***1.33%***
4	1	1	8	0	**506**	0	10	0	0	2	***95.83%***	***4.17%***
5	2	0	0	0	0	**526**	0	0	0	0	***99.62%***	***0.38%***
6	1	0	0	0	2	0	**525**	0	0	0	***99.43%***	***0.57%***
7	0	0	0	0	0	0	0	**520**	8	0	***98.48%***	***1.52%***
8	3	0	0	0	0	0	0	3	**522**	0	***98.86%***	***1.14%***
9	3	1	0	0	2	1	0	0	0	**521**	***98.67%***	***1.33%***
											98.66%	**1.34%**

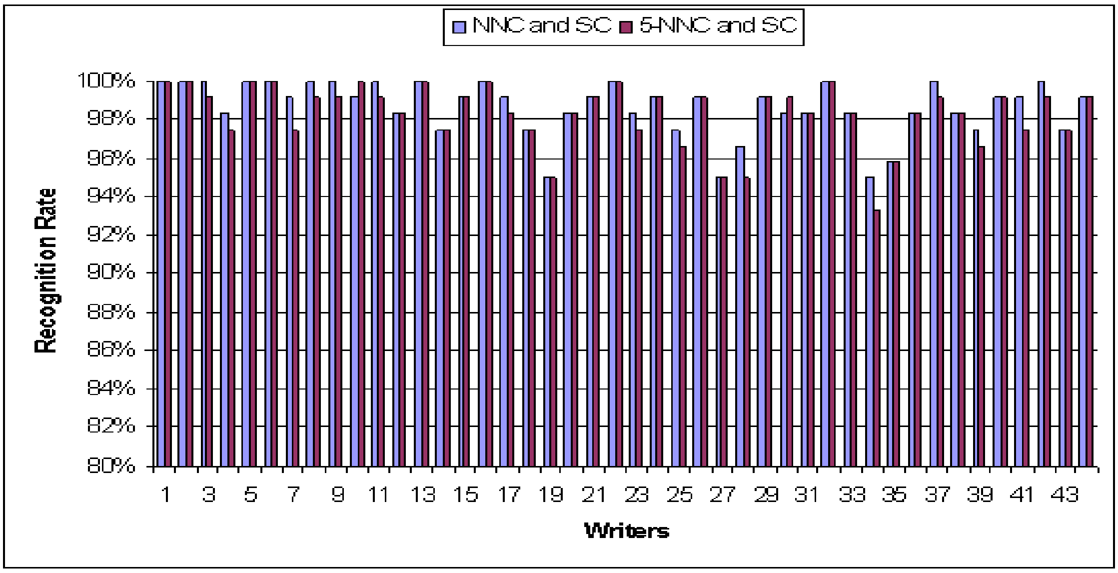


Fig. 11 The recognition rate for all the writers before and after the Structural Classifier.

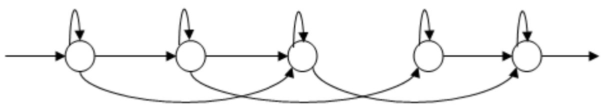


Fig. 12 A 5-state Hidden Markov Model (HMM)

The confusion matrix for the 36 observation, 10 feature/observation case is shown in Table 5. %c is percentage of recognition rate and %e is the error percentage. The average recognition rate is 94.01%. It is clear from the confusion matrix that the majority of errors (67%) are due to confusing 7 with 8 and vice versa. The average recognition rate excluding digits 7 and 8 is 97.6%.

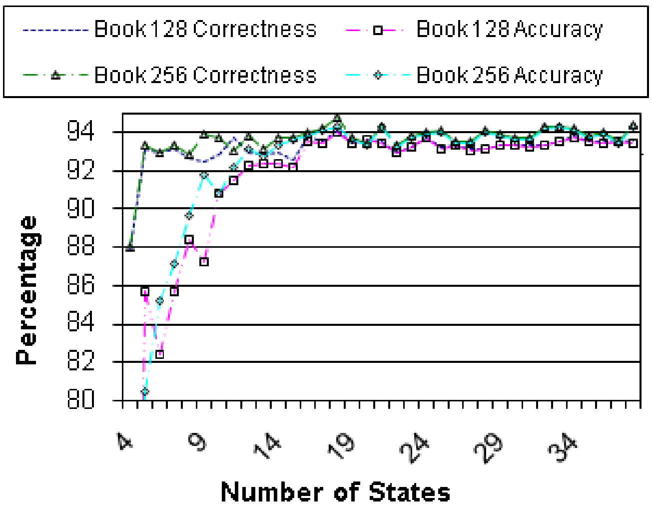


Fig. 13 The correct recognition rate using 4 to 38 states, and with 128 and 256 code books.

Table 4 Correctness and accuracy at states 4,10,18,38 and the mean and variance of the correctness and accuracy for states 10 to 38 for codebooks of 128 and 256.

Code Book Size	128	128	256	256
State	Correctness	Accuracy	Correctness	Accuracy
4	87.9	41	87.97	43.72
10	92.87	90.82	93.72	90.78
18	94.11	93.98	94.81	94.35
38	93.53	93.48	94.44	94.33
Mean of results using 10 to 38 states	93.36	93.10	93.86	93.59
Variance of results using 10 to 38 states	0.35	0.70	0.40	0.73

On using the SC stage of the previous section for the 7 and 8 digits the above recognition rates are improved. The recognition rate of digit 7 improved from 77.3% to 92.9% and the recognition rate of digit 8 improved from 82.02% to 97.3%. This improvement resulted in boosting the average recognition rate of all digits from 94.01% to 97.1% (an improvement of 3.1%).

In order to compare the three classifiers experiments with similar parameters are conducted. The NNC is trained using 36 samples per digit per writer, with angle increment of 5 degrees resulting in 36 projections, using 10 features per projection. The remaining samples are used for testing. The features of the training samples are averaged and used as the models for the digits in the testing phase. The HMMC is tested using the same number of features, a HMM with 18 states, the features of each digit are split into 36 observations with 10 features each. 1700 numbers (totaling 16657 digits) are used in training and digits are randomized. The remaining samples are used in testing (a total of 4463 digits). The results of these experiments of the HMMC are shown in Fig. 14. Applying SC stage improved the recognition rate by 3.1%.

It is worth mentioning that there is no generally accepted data base for Arabic text/numeral recognition that is freely available for researchers. Different researchers of Arabic text/numeral recognition use different data and hence the recognition rates of the different techniques may not be comparable. So, our results are based on our data and we have no access to an Arabic/Indian numerals database other than our database that we used in this work.

The samples of the database that are misclassified are inspected manually. The reasons for the misclassification errors may be attributed to errors due to bad (corrupted) data, errors due to deformed samples or samples with un-proportional segments in relation to other segments in length and orientation, errors due to samples written with different style than the training style, errors related to digit pairs, or genuine errors that are misclassified with no visible reason and that can be attributed to insufficient classification capability of the used features and classifiers.

1. Errors due to bad or corrupted data. Fig. 15 shows samples of bad or corrupted data.
2. Errors due to deformed samples or samples with un-proportional segments in relation to other segments in length and orientation. Fig. 16 shows samples with this type of errors.
3. Errors due to samples written with different style than the training style. For example, digit three may be written with three upward segments or two upward segments (٣ , ٣). Majority of the writers used the three stroke style and few used the two stroke style. This type of error can be addressed by allowing a digit to have more than one model.
4. Errors related to digit pairs. Some digits are close in shape to other digits. Hence, if not written specifically they look similar to other digits. For example, if digit 5 is written in small size and the inside hole is small it is normally confused with zero. Another example is the digit zero which is similar to a dot. Sometimes it is written like a small line or spreading pixels which looks as a five. When the digit is normalized it looks very similar to digits one or five. In such cases it is confused with digit 1/five. However, when digits are included with Arabic text then this problem is expected to disappear as the normalization will be based on the line height. Hence, digit zero will be very small compared with a one and hence will not be confused with it. There are other digit pairs that are sometimes confused (viz. digit 7 with digit 8, rotating of 7 by 180 degree gives an 8). The structural classifier eliminated most of these errors.
5. Genuine errors that are misclassified with no visible reason and that can be attributed to insufficient classification capability of the used features and classifiers.

Table 5 The confusion matrix using HMMC with 36 observations, 10 features per observation using a code book of 256 and 18 states.

Digit	0	1	2	3	4	5	6	7	8	9	%c	%e
0	440	4	0	0	0	0	0	1	1	5	97.56%	0.57%
1	0	443	0	0	0	0	0	0	0	0	100.00%	1.51%
2	0	0	435	0	0	0	6	0	0	0	98.64%	1.51%
3	0	0	10	427	2	0	0	4	1	1	95.96%	1.88%
4	0	0	4	0	437	0	9	0	0	1	96.90%	4.89%
5	4	0	15	0	1	425	0	0	1	0	95.29%	1.31%
6	0	1	0	0	3	0	438	0	0	0	99.10%	1.69%
7	0	0	0	0	1	0	0	344	99	1	77.30%	2.80%
8	0	0	0	0	0	1	0	79	365	0	82.02%	2.61%
9	0	0	0	0	11	0	0	1	0	435	97.32%	2.98%
											94.01%	2.18%

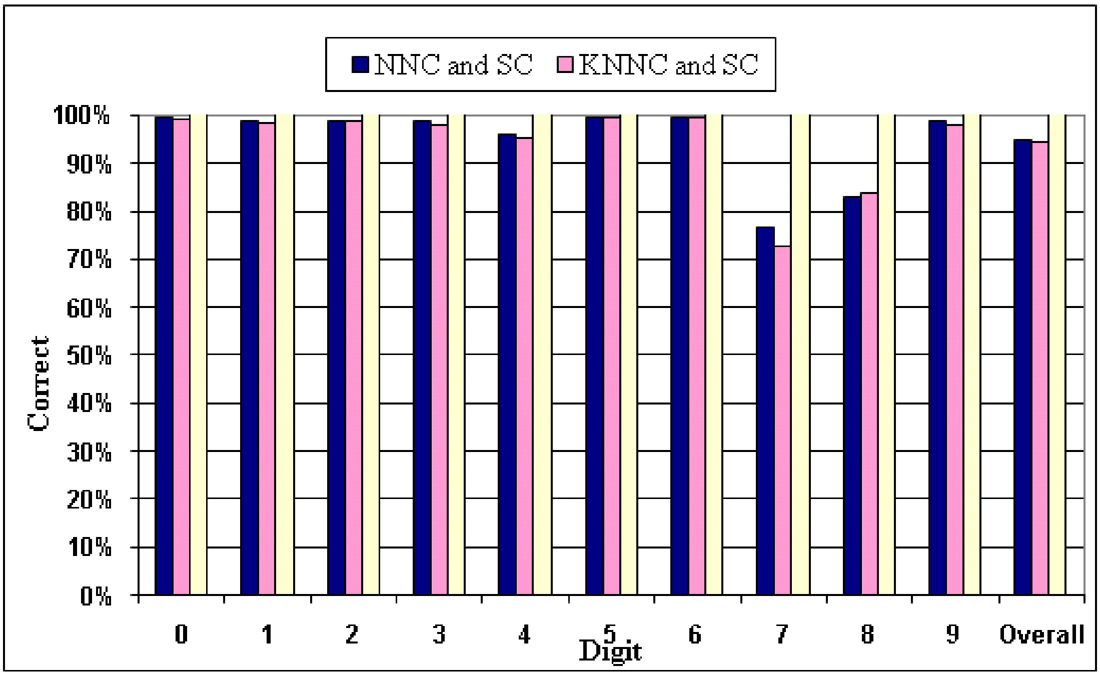
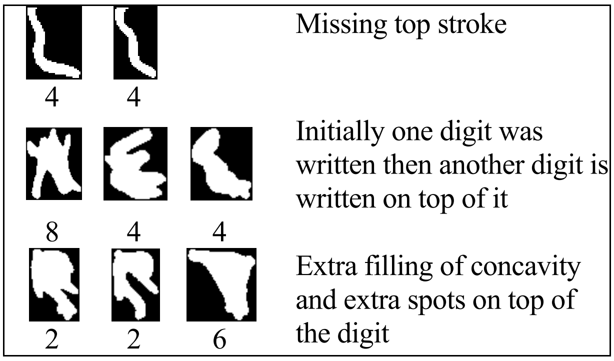


Figure 14 The recognition rates of HMMC, NMCC, and K-NNC using 36 Radon projections with 10 per projection.

Our analysis of the database indicates that a human is liable to make 1% misclassification errors on classifying the data in the database if the context is not present. Hence, previously discussed recognition rates should be compared to a maximum of 99% instead of 100%. This indicates that applying Radon-Fourier-based features with the structural classifier achieved high recognition rates for the offline Arabic (Indian) numerals recognition.



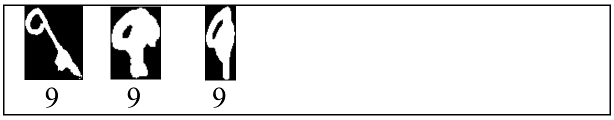


Fig. 15 Samples of badly written or corrupted data

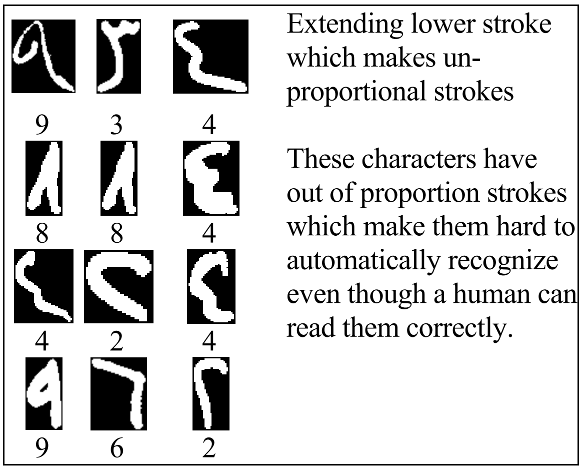


Fig. 16 Samples of error data due deformed or un-proportional segments

5 Conclusions

This paper presented a system for off-line handwritten Arabic (Indian) numeral recognition using Radon and Fourier transforms. In this work the NNC, K-NNC, and the HMMC classifiers are used. Using the NNC, K-NNC we analyzed the technique by running many experiments using variations of the number of used projections and the number of Radon-Fourier-based features. In addition, the number of samples used in the training and testing of this technique are varied. In the training phase a number of samples' features of each digit are averaged and used as the models of Arabic numerals. We varied the number of samples used for training from 24 to 47 samples per writer per digit. The remaining samples are used for testing. A database of 44 writers with 48 samples per digit each, totaling 21120 samples are used for training and testing of this technique. In the testing phase the features of the test sample is compared with the features of all the models. The argument of the minimum distance between the test sample and the models is taken as the label of the test sample.

The performance of this technique using HMMC is analyzed. 1700 Arabic numbers of varying lengths (totaling 16657 digits) are used in training of the HMMC and 471 numbers of varying length (totaling 4463 digits) are used in testing the HMM. Several experiments are conducted to achieve the best recognition rate by using different number of states in the model and by modifying the number of used observations per digit. An average recognition rate of 94.01% is achieved using 360 features presented as 36 observations of 10 features each per digit. An HMMC of 18 states is used as it gives best accuracy. Randomization of presenting the observations was necessary in the training of the HMM. As the digits have equal probability the randomization is applied in the length of Arabic numbers and in the used digits in each number.

On the average a classification rate of 94.87%, 94.28%, and 94.01% are achieved using NNC, K-NNC, and HMMC, respectively. On analyzing the classification errors it is clear that the majority of errors are due to misclassification of digit 7 with 8 and vice versa. Hence, a second classification stage is added using the structural features of digits 7 and 8. This stage improved the overall average recognition rate by 3.1%, 3.79%, and 4.05% for HMMC (from 94.01% to 97.1%), K-NNC (from 94.28% to 98.33%), NMC (from 94.87% to 98.66%), respectively. The classification errors, after the second stage, are analyzed and it is clear that some errors may be attributed to bad data, some to deformation and unbalanced proportion of digit segments, and others are genuine errors. On manually analyzing the erroneous data, after the second stage, it is clear that a human is liable to make 1% misclassification if the data is presented to him without context. This proves the effectiveness of the presented technique to off-line Arabic (Indian) handwritten digit recognition. These results clearly indicate the applicability of this technique to off-line handwritten Arabic (Indian) digit recognition.

This work will be extended to address Arabic text recognition.

Acknowledgment

The authors would like to thank King Fahd University of Petroleum and Minerals for supporting this research and providing the computing facilities.

References:

[1] V. K. Govindan, A. P. Shivaprasad, "Character Recognition - A Review," Pattern Recognition, Vol. 23, No. 7, pp. 671-683 (1990).

[2] Liu, C.L., Nakashima, K., Sako, H., Fujisawa, H., 2003. Handwritten digit recognition: Benchmarking of state-of the- art techniques. Pattern Recognition, 36, 2271–2285.

[3] Shi, M., Fujisawa, Y., Wakabayashi, T.,

Kimura, F., Handwritten numeral recognition using gradient and curvature of gray scale image. Pattern Recognition, 35(2002), 2051–2059.
[4] Cheung, K., Yeung, D., Chin, R.T., A Bayesian framework for deformable pattern recognition with application to handwritten character recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29 (12), (1998), 1382–1388.
[5] Chevalier S., Geoffrois E., Preteux F., 2D Markovian modeling for character recognition and segmentation, WSEAS Transactions and Communications, Vol. 4, No. 9, September, 2005, pp. 915-920.
[6] Kunghee Lee, "Hand-Written Digits Recognition by Graph matching and Annealing Neural Networks", WSEAS Transactions on Information Science & Applications, Issue 5, Volume 5, 2008, pp. pp. 716-725.
[7] Al-Omari F. Hand-written Indian numeral recognition systems using template matching approaches. Proc. ACS/IEEE Int. Conf. Computer Sys Appl. 2001; 83–8.
[8] F.A. Al-Omari, O. Al-Jarrah, Handwritten Indian numerals recognition system using probabilistic neural networks, Advanced Engineering Informatics 18 (2004) 9–16.
[9] Bousalma F. Structural and fuzzy techniques in the recognition of online Arabic characters. Int J Pattern Recogn Artif Intell 1999;13(7): 1027–40.
[10] Salah A, Albaydin E, Akarun L. A selective attention-based method for visual pattern recognition with application to handwritten digit recognition and face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002;24(3): 420–5.
[11] S. Almaadeed, C. Higgens, and D. Elliman, “Recognition of Off-line Handwritten Arabic Words using Hidden Markov Model Approach,” ICPR 2002, Quebec City, August 2002, pp. 481-484.
[12] S. Almaadeed, Higgins, Colin; Elliman, David G., Off-line recognition of handwritten Arabic words using multiple hidden Markov models, Knowledge-Based Systems, Vol. 17, pp. 75-79, 2004.
[13] Touj, Sameh; Amara, Najoua Ben; Amiri, Ham,Arabic Handwritten Words Recognition Based on a Planar Hidden Markov Model, Int. Arab J. Inf. Technolgy, vol. 2, no. 4, pp. 318-325, 2005.
[14] Badr Al-Badr, Sabri A. Mahmoud, “Survey and Bibliography of Arabic Optical Text Recognition,” J. of Signal Processing, Vol. 41, No.1, pp.49-77 (Jan. 1995).
[15] H. El Abed and V. Margner, “Arabic Text Recognition Systems - State of the Art and Future Trends,” 5th International Conference on Innovations in Information Technology (Innovations'08), Al Ain, UAE: 2008.
[16] Liana M. Lorigo, Venu Govindaraju, “Offline Arabic Handwriting Recognition: A Survey”, EEE Trans. Pattern Analysis and Machine Intelligence, vol. 28, no. 5, pp. 712-724,May 2006.
[17] Shahrezea, M.H.S., Faez, K., Khotanzad, A., Recognition of handwritten Persian/Arabic numerals by shadow coding and an edited probabilistic neural network. In: Proceedings of the International Conference on Image Processing, vol. 3,(1995), pp. 436–439.
[18] Hosseini, H.M.M., Bouzerdoum, A., A combined method for Persian and Arabic handwritten digit recognition. Proceedings of the Australian New Zealand Conference on Intelligent Information Systems, 1996, pp. 80–83.
[19] Said, F.N., Yacoub, R.A., Suen, C.Y., Recognition of English and Arabic numerals using a dynamic number of hidden neurons. In: Proceedings of the Fifth International Conference on Document Analysis and Recognition, 1999, pp. 237–240.
[20] Sadri, J., Suen, C.Y., Bui, T.D., Application of support vector machines for recognition of handwritten Arabic/Persian digits. In: Proceedings of Second Iranian Conference on Machine Vision and Image Processing, vol. 1, 2003, pp. 300–307.
[21] Hasan Soltanzadeh, Mohammad Rahmati, Recognition of Persian handwritten digits using image profiles of multiple orientations, Pattern Recognition Letters 25 (2004) 1569–1576.
[22] S. R. Deans, The Radon Transform and Some of Its Applications. New York: Wiley, 1983.
[23]Nurulhuda Ismail, Mas Idayu Md. Sabri, Review of Existing Algorithms for Face Detection and Recognition, Proceedings of the 8th WSEAS International Conference on Recent Advances in Computational Intelligence, Man-Machine Systems and Cybernetics, Puerto De La Cruz, Tenerife, Canary Islands, Spain, December 14-16, 2009, pp. 30-39.
[24] J. Janta, P. Kumsawat, K. Attakitmongkol and A. Srikaew, A Pedestrian Detection System Using Applied Log-Gabor Filters, Proceedings of the 7th WSEAS International Conference on Signal, Speech and Image Processing, Beijing, China, September 15-17, 2007, pp. 55-60.
[25] Illingworth, J. - Kittler, J.: “A survey to the Hough Transform.” Comput. Vision, Graphics, Image Precess. 44(1), 1988, 87-116.
[26] G. Beylkin, “Discrete Radon transform,” IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-35, no. 2, pp. 162-172, Feb. 1987.

[27] Copeland, A.C.; Ravichandran, G.; Trivedi, M.M., “Localized Radon transform-based detection of ship wakes in SAR images”, IEEE Transactions on Geoscience and Remote Sensing, Volume 33, Issue 1, Jan. 1995 Page(s):35 – 45.
[28] Toft, P.A., “Using the generalized Radon transform for detection of curves in noisy images”, 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings. 7-10 May 1996 Volume 4, Page(s):2219 – 2222.
[29] Junhong Li, Quan Pan; Hongcai Zhang; Peiling Cui, “Image recognition using Radon transform”, Proceedings. 2003 IEEE Intelligent Transportation Systems, 2003, Volume 1, 2003 Page(s):741 - 744 vol.1
[30] Lian Cai, Sidan Du, “Rotation, scale and translation invariant image watermarking using Radon transform and Fourier transform”, Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication, 2004, Volume 1, 2004 Page(s):281 - 284.
[31] Bazzi, C. LaPre, J. Makhoul, and R. Schwartz, “Omnifont and Unlimited Vocabulary OCR for English and Arabic,” Proc. Int’l Conf. Document Analysis and Recognition, vol. 2, pp. 842-846, Ulm, Germany, 1997.
[32] I. Bazzi, R. Schwartz, and J. Makhoul,”An Omifont Open-Vacabulary OCR system for English and Arabic”, IEEE Transaction on PAMI, Vol. 21, No. 6, pp. 495-504, 1999.
[33] Hassin, A. H., Tang, X., Liu, J., and Zhao, W. Printed Arabic character recognition using HMM, Journal of Computer Science and Technology, Volume 19 Issue 4, July 2004, pp. 538-543.
[34] Sabri A. Mahmoud, “Recognition of writer-independent Off-line Handwritten Arabic (Indian) Numerals Using Hidden Markov Models”, Signal Processing, Vol. 38, issue 4, April 2008, pp.844-857.
[35] HTK Speech Recognition Toolkit, http://htk.eng.cam.ac.uk/