Printed and Handwritten Hindi/Arabic Numeral Recognition Using Centralized Moments

Mohamed H. Ghaleb, Loay E. George, and Faisel G. Mohammed

Abstract: Printed and handwritten numerals recognition plays a vital role in postal automation services. The major problem in handwritten recognition is the huge variability and distortions of patterns. The aim of the current research work is to develop fast and efficient method to recognize Hindi printed and free handwritten numerals objects. In this research, the introduced method for extracting features from patterns is based on the relative density distribution of each numeral object; specifically it depends on the centralized moments. This method gives sufficient results to recognize the printed and highly stylized handwritten numeral images. The attained recognition rate is 97.47% for the printed numeral images with total number of samples equal (198) samples and 95.55% for the highly stylized handwritten numeral images with total number of samples equal (90) samples, while, the attained recognition rate is unacceptable when the system is applied for a handwritten numeral samples which have wide differences in their shapes with total number of samples equal (4500) samples. The attained recognition rate is (74.93%). Each tested numeral image is scanned with scanning resolution of 300 dpi.

Keyword: Artificial intelligence, pattern recognition, image segmentation, character recognition, handwritten recognition.

1 Introduction

Handwritten recognition has attracted many researchers across the world **[1, 9, and 10].** The problem of automatic recognition of handwritten text as opposed to machine printed text is a complex one, especially for cursive based languages. Several researchers have introduced algorithms for character recognition for different languages such as English, Chinese, Japanese, and Arabic **[2, 4, and 5].**

Typical Optical Character Recognition (OCR) system consists of the phases: preprocessing, segmentation, feature extraction, classifications and recognition. The output of each stage is used as the input of next stage. Preprocessing stage consists of many adjustment operations for slant correction, normalization and thicking. Many newly proposed methods have been introduced for the purpose of feature extraction **[3, 5].**

Most of the Indian scripts are distinguished by the presence of matras (or, character modifiers) in addition to main characters, while the English script has no matras. Therefore, the algorithms developed specifically for English are not directly applicable to Indian scripts **[6].**

2 Related Works

In recent years some researchers have developed computational intelligence models for accurate recognition of Arabic text. Al-Omari **[7]** used an average template matching approach for recognizing Arabic (Indian) numerals. He suggested the use of feature vectors representing a set of significant boundary points distances from the center of gravity (COG) of the numeral object. He were used these features to derive a model for each numeric digit. An overall hit ratio of 87.22% was achieved in the preliminary results. This ratio reached 100% for some of the digits. But there was misinterpretation between similar digits like (6) and (9). Classification was performed using the Euclidean distance between the feature vector of the test samples and the generated models.

Sadri et al. **[8]** proposed the use of support vector machine for the recognition of isolated handwritten Arabic/Persian numerals. The method views each digit from four different directions, and then extracting the features used which are used to train SVM classifiers to recognize the digit classes. An average recognition rate of 94.14% was obtained.

A new method based on Hidden Markov Model (HMM) for recognition of isolated handwritten Arabic (Indian) numerals was presented by Mahmoud **[9]**. In his method, four sets of features (i.e. angle, circle, horizontal and vertical (ACHV)), were generated based on the segmentation of numeral image, and for each segment the ratio of black pixels to segment size was computed. These features were used for training and evaluating the HMM models. Average recognition rate of 97.99% was achieved.

The use of abductive machine learning techniques for the recognition of handwritten Arabic (Indian) numerals was demonstrated by Lawal **[10].** An average recognition rate of 99.03% was achieved with a set of only 32 features based on FCC codes.

Ghaleb et al. [11] used a heuristic based method for rec-

- *Mohamed H. Ghaleb, Baghdad University, Science College, Computer Science Department, Iraq – Baghdad. E-mail: mohamedghaleb1980@yahoo.com*
- *Dr. Loay E. George, Baghdad University, Science College, Computer Science Department, Iraq – Baghdad. E-mail:* loayedwar57@scbaghdad.edu.iq
- *Dr. Faisel G. Mohammed, Baghdad University, Science College, Computer Science Department, Iraq – Baghdad. E-mail:* faisel@scbaghdad.edu.iq

ognizing Hindi numeral free handwritten objects, this method based on (i) the percentage of strokes in both horizontal and vertical directions and (ii) some morphological operations. The attained recognition rate is 98.15%, the number of tested samples was 4500 samples.

3 Proposed System Description

Like other languages Indian has 10 basic digits, the scope of this paper is limited to develop a simple and fast Method for detecting the Printed and Handwritten Hindi numerals (from one to nine: 1-9) which are commonly used in Arabic writing.

Figure (1) shows the scheme of the proposed system. In general, the proposed system involved on three stages they are:

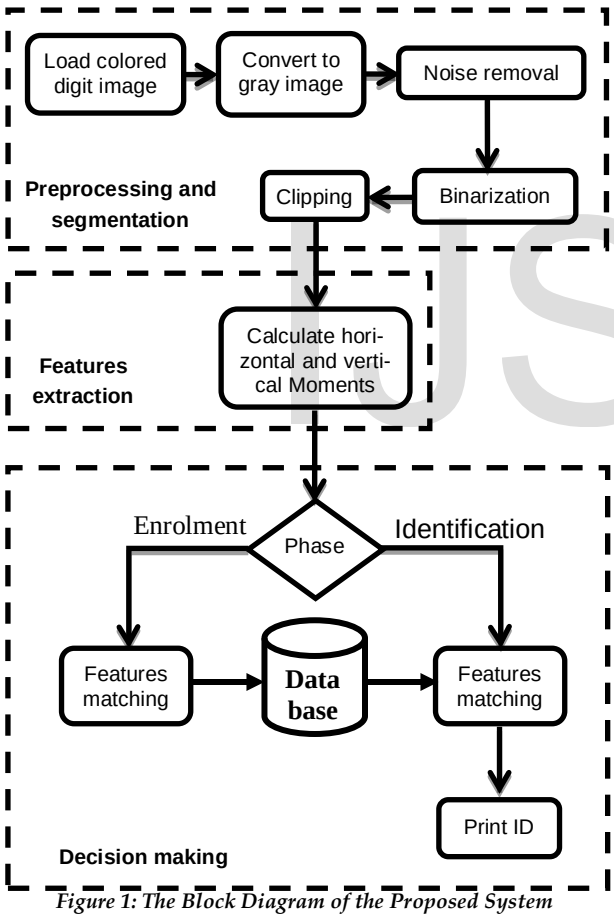


Figure 1: The Block Diagram of the Proposed System

1. Preprocessing and segmentation stage: this stage is involved with loading the numeral color image file; then convert it to gray image. The obtained gray image is enhanced using average filter to remove the unwanted isolated pixels (noise); it is converted to binary image using thresholding method, and then the image is clipped.
2. Feature extraction stage: the centralized image moments in both horizontal and vertical directions are calculated in this stage.
3. Decision making stage: this stage consists of two phases: (i) enrolment phase (save mean and standard deviation of moments in the database, and features analysis) and, (ii) identification phase (moments matching with the database to identify the final ID of the input image). The next sections explain these three stages in details.

3.1 Preprocessing and Segmentation

This stage consists of the following tasks:

A. Load colored digit image: The first step in the proposed system is loading the bitmap image file with format 8 or 24 bit color depth.

B. Convert to gray image: The numeral image is converted to gray scale image by calculating average value of Red, Green, and Blue for each pixel in color image.

C. Noise removal: To remove noise from binary digit image the (3×3) average (mean) filter was used.

D. Binarization: In the normal case the numeral image should consist of two colors (i.e., foreground & background), the largest repeated color refers to the background and the second largest repeated color refers to the foreground color. The steps taken to convert grayscale image to binary image are:

i. *Determine the histogram of the image.*
ii. *Search in the histogram to find the largest two peaks they should be separated at least by certain colors. So, the distance between the locations of these two peaks should be more than a predefined minimum distance value. The midpoint between these two peaks is considered as the threshold value used to convert image from gray to binary:*

Threshold value = (PR+PL)/2

Where, PR is the value of right peak, and PL is the left peak value.

iii. *Scan all images pixels if the gray pixel value is close to the highest peak value then set pixel value (0) otherwise set the value (1).*

Where black pixel (0) refers to the background and white pixel (1) refers to the foreground.

E. Clipping: This process is used to trim the numeral image by removing the left, right, top, and bottom space such that the new image boundaries confine the numeral object area. The scanning process consists of the following steps:

i. *Find the left and right edge (the most left and right columns contain white pixels (1)).*
ii. *Find the top row and bottom row (the first and last row contain white pixels (1)).*

The new width and height of the clipped image are:

Width = right edge – left edge + 1
Height =bottom edge – top edge + 1
So, the size of the clipped image is (width, height)

3.2 Features Extraction

The extracted features are mainly based on the relative density distribution of each numeral object, by calculating its centralized moments in both horizontal and vertical direction to higher order. The calculation steps for the centralized moments for both horizontal and vertical direction are:

i. *Let HP as array [0 to width-1] and VP as array [0 to height - 1] which represents the horizontal and vertical percentage of the number of black pixel in the clipped image for each columns and rows.*
ii. *Let Xcenter as the center coordinate of width and Ycenter as the center coordinate of height of the clipped image.*
iii. *Calculate moments for horizontal and vertical from the following equations.*

$$Momx(n)=\sum_{i=0}^{Width-1} HP(i)\ ((i-Xcenter)/Xcerter)^{n}$$

$$Momy(n)=\sum_{i=0}^{Height-1} VP(i)\ ((i-Ycenter)/Ycerter)^{n}$$

Where, Momx is the horizontal moments, Momy is the vertical moments, and n is the moment order.

3.3 Decision Making

The proposed system has two phases enrolment phase and identification phase. The next subsections explain each phase in details.

3.3.1 Enrolment phas: The enrolment phase implies the following steps:

A. Database*:* Each numeral type (i.e., 1 to 9) should have a set of template values for the horizontal and vertical moments, which represent its mean of features. So, the mean and standard deviation of these features and for each numeral (i.e., from 1 to 9) should be calculated from training samples and saved in the database file. The following stepes was used to save the mean and standard deviation in the database file.

i. *Let Amomx and Amomy as two arrays [0 to moment order] which represent the features of horizontal and vertical training sample.*
ii. *Let Meanx, Meany, Standx, and Standy as four arrays [1 to 9, 1 to moment order] which represent the horizontal and vertical means and standard deviations and set all elements of these arrays equal to 0.*
iii. *for all training samples calculate Amomx and Amomy by applying features extraction steps and then for each moment order (i.e., 1 to 11) calculate the following equations: Meanx[i,j]= Meanx[i,j] + Amomx[j]*
Meany[i,j]= Meany[i,j] + Amomy[j]
Where, i is the training sample name and j is the moment order
iv. *Divide all elements belong Meanx or Meany by number of training samples.*
v. *for all training samples calculate the following equations:*
Standx[i,j]= Standx[i,j] + (Amomx[j]-Meanx[I,j])^2
Standy[i,j]= Standy[i,j] + (Amomy[j]-Meany[I,j])^2
vi. *Divide all elements belong Standx or Standy by number of training samples and then determine its square root.*
vii. *Open database file as binary and then put all elements of Meanx, Meany, Standx, and Standy in this file.*

B. Features Analysis: Features extraction stage could generate a large number of features by calculating the horizontal and vertical moments up to high orders, but not all of these features are suitable. Features analysis task is responsible for the selection of only suitable features and drop the poorly discriminating features. This process is done by trying all possible combinations of features to find the proper combination which can lead to best recognition rate.

3.3.2 Identification Phase: The first proposed method utilizes the minimum distance classifier to identify the numeral images. The horizontal and vertical moments of the tested image matches the mean and standard deviation for all samples in the database. The following stepes was used to implement the identification phase:

i. *Calculate Dif (which represent the difference between the tested image and one digit in the database) from the following equation:*

$$Dif=\sum_{i}^{N} |(Momx(i)-Meanx(i))/(Standx(i))| + \sum_{j=0}^{N} |(Momy(i)-Meany(i))/(Standy(i))|$$

Where, N is the moment order, x and y is the horizontal and vertical direction, Mom is the moments for input image, Mean is the mean in database, and Stand is the standard deviation in database.

ii. *Calculate difference for each digit in the database by repeat step (i).*
iii. *Find the minimum difference, which represent the ID of the tested image.*

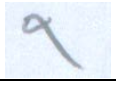
4. The Tested Image Samples

There are two types of samples used to determine the performance of the proposed system; the first type is a set of

printed Hindi numeral images, while the second type is a set of handwritten Hindi numeral images.

The number of printed samples was 22 samples for each numeral type (from one to nine); the total number of printed samples was 198 samples with Arial font of size extending from 8 to 28 with and without bolding effect, while the handwritten numeral type was written by different peoples in different style, as shown in Table (1). The number of handwritten numeral samples was 500 samples for each numeral type (from one to nine); the total number of samples was 4500 samples extracted from (42) scanned documents prepared by (42) persons. The proposed system has ability to recognize numeral objects in different background and foreground colors as shown in Table (1).

Table 1: Different Styles of Handwritten Hindi Numerals Samples

Numeral	Style1	Style2	Style3
1			
2			
3			
4			
5			
6			
7			
8			
9			

5. Results and Conclusions

The proposed system has been tested for both printed and handwritten Hindi numeral images, which mentioned in the previous section, the highest taken moment order was 11 in all conducted test.

The following equation has been used to calculate each recognition ratio:

Recognition Ratio= (No.of succesful hits)/(No.of samples) x100

where No. of successful hits represents the correct recognition results, and No. of samples represents the number of all tested samples.

5.1 Experimental Results for Printed Numerals

The results of tests on printed numerals indicated that the recognition ratio is (97.47%), when the database contains six means and standard deviations for each numeral type (from 1 to 9), each of the first three means and standard deviations correspond to certain range of font size (small, medium, and large) and for the bold font samples, while the second three means and standard deviations is the same as the first three font size but refers to the samples without bolding effect. The horizontal moments of orders ={1,2,10,11} with the vertical moments of orders ={1,5,9,10}, and the images scanned using 300 dpi resolution. Table (2) shows the success and misclassified ratios for each digit (from 1 to 9) with and without bolding effect, where the cells in first row represents the tested digit, the cells in first column represents the results of tested digit, and the master diagonal (cells with gray background) represents success ratios.

Table2. The final recognition ratio of (Success and Misclassified Rate) for printed samples

	1	2	3	4	5	6	7	8	9
1	100	0	0	0	0	0	0	0	0
2	0	100	0	0	0	0	0	0	0
3	0	0	100	0	0	0	0	0	0
4	0	0	0	100	0	0	0	0	0
5	0	0	0	0	100	0	0	0	0
6	0	0	0	0	0	86.363	0	0	4.545
7	0	0	0	0	0	4.545	100	0	4.545
8	0	0	0	0	0	0	0	100	0
9	0	0	0	0	0	9.09	0	0	90.909

5.2 Experimental Results of Handwritten Numerals

Numeral 3 in handwritten samples has two different styles, some people write it as "3" while other people write it as "2", to solve this problem the 3ed class is considered as two subclasses (class3.1 which represents the numeral images as "3", and class 3.2 which represents the numeral images "2").

Handwritten Hindi numeral images have been tested with the database contain one mean and standard deviation for each numeral type (from 1 to 9). The results of test indicate that the recognition ratio is (74.93%), when the horizontal moments of orders ={1,2,6,11} with the vertical moments of orders ={1,2,9,10}, and images are scanned using 300 dpi resolution.

Table (3) shows the recognition ratios for each digit

Table3. The final recognition ratio of (Success and Misclassified Rate) for printed samples

	1	2	3	4	5	6	7	8	9
1	90	3	3.2	4.6	0	5.6	0.8	0.4	0.2
2	2.8	60.6	3.2	5.8	1	1.2	1.2	3.4	10.6
3	1.6	3	61	0	0	1.8	0.4	0.8	8.8
4	1.4	7	0	87.8	0.6	0	0.6	2	0.2
5	0	1.2	0.8	0	93.8	0	4.8	3	0
6	2.8	2.6	5.8	0	0.4	81.8	0.2	0.8	22.6
7	0	0.4	1.6	0	1.4	0	80	7	0.4
8	0.4	16.6	5.6	1.8	1.6	0.2	10.6	76.8	14.6
9	1	5.6	18.8	0	1.2	9.4	1.4	5.8	42.6

Another test has been done on the highly stylized handwritten numeral images, the number of samples was 10 for each numeral type; so, the total number of samples was 90 samples. These samples had been tested with the database contains one mean and standard deviation for each numeral (from 1 to 9). The results of tests indicated that the recognition ratio is (95.55%), when the horizontal moments of orders ={1,2,4,11} with the vertical moments of orders ={1,3,4,10}, and the images are scanned using 300 dpi resolution. Table (4) shows the recognition ratio of this case.

Table4. The final recognition ratio of (Success Rate, Failure Rate, and Misclassified Rate)

Samples	1	2	3	4	5	6	7	8	9
Success Rate	100	100	90	100	100	100	80	100	90
Misclassified Rate	0	0	10	0	0	0	20	0	10

REFERENCES

[1] Wadhwa, D. and Verma, K., "Online Handwriting Recognition of Hindi Numerals using Svm", *International Journal of Computer Applications*, Volume 48 – No. 11, pp (13-17), June 2012.

[2] Elnagar, A., Al-Kharousi, F., and Harous, S., "Recognition of Handwritten Hindi Numerals using Structural Descriptors", *IEEE International Conference*, Volume 2, pp (983-988), 12-15 Oct 1997.

[3] Sinha, G., Rani, R., and Dhir, R., "Handwritten Gurmukhi Numeral Recognition using Zone-based Hybrid Feature Extraction Techniques", *International Journal of Computer Applications*, Volume 47– No.21, pp (24-29), June 2012.

[4] Rani, A., Rani, R., and Dhir, R., "Combination of Different Feature Sets and SVM Classifier for Handwritten Gurumukhi Numeral Recognition", *International Journal of Computer Applications*, Volume 47– No.18, pp (28-33), June 2012.

[5] Dhandra, B.V., Benne R.G., and Hangarge, M., "Printed and Handwritten Kannada Numerals Recognition Using Directional Stroke and Directional Density with KNN", *International Journal of Machine Intelligence*, Volume 3, Issue 3, pp (121-125), November 2011.

[6] Hanmandlu, M., Nath, A.V., Mishra, A.C., and Madasu, V.K., "Fuzzy Model Based Recognition of Handwritten Hindi Numerals using Bacterial Foraging", *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)*, 11-13 July 2007, Melbourne, Australia.

[7] Al-Omari, F., "Hand-Written Indian Numerals Recognition System Using Template Matching Approaches", *ACS/IEEE International Conference on 2001*, pp (83-88), 25-29 Jun 2001.

[8] Sadri, J., Suen, C.Y. and Bui, T.D., "Application of Support Vector Machines for Recognition of Handwritten Arabic/Persian Digits", *Toosi Univ. of Tech*., 2nd MVIP, Volume 1 pp (300-307), Feb 2003, Tehran, Iran.

[9] Mahmoud, S., "Recognition of Writer-Independent off-line Handwritten Arabic (Indian) Numerals using Hidden Markov models", *Signal Processing*, Volume 88, Issue 4, pp (844-857), April, 2008.

[10] Lawal, I.A., Abdel-Aal, R.E., and Mahmoud, S.A., "Recognition of Handwritten Arabic (Indian) Numerals Using Freeman's Chain Codes and Abductive Network Classifiers", International Conference on Pattern Recognition, *IEEE Computer Society*, pp (1884-1887), 23-26 Aug. 2010.

[11] Ghaleb, M.H., George, L.E., and Mohammed, F.G., " Numeral Handwritten Hindi/Arabic Numeric Recognition Method", International Journal of Scientific & Engineering Research, Volume 4, Issue 1, January-2013.