Handwritten Numeral Databases of Indian Scripts and Multistage Recognition of Mixed Numerals

Ujjwal Bhattacharya, *Member*, *IEEE*, and B.B. Chaudhuri, *Fellow*, *IEEE*

Abstract—This paper primarily concerns the problem of isolated handwritten numeral recognition of major Indian scripts. The principal contributions presented here are 1) pioneering development of two databases for handwritten numerals of the two most popular Indian scripts, 2) a multistage cascaded recognition scheme using wavelet-based multiresolution representations and multilayer perceptron (MLP) classifiers, and 3) application of 2 for the recognition of mixed handwritten numerals of three Indian scripts—Devanagari, Bangla, and English. The present databases include, respectively, 22,556 and 23,392 handwritten isolated numeral samples of Devanagari and Bangla collected from real-life situations, and these can be made available free of cost to researchers of other academic institutions. In the proposed scheme, a numeral is subjected to three MLP classifiers corresponding to three coarse-to-fine resolution levels in a cascaded manner. If rejection occurs even at the highest resolution, another MLP is used as the final attempt to recognize the input numeral by combining the outputs of three classifiers of the previous stages. This scheme has been extended to the situation when the script of a document is not known a priori or the numerals written on a document belong to different scripts. Handwritten numerals in mixed scripts are frequently found in Indian postal mail and tabular form documents.

Index Terms—Handwritten character recognition, Indian script character recognition, multiresolution recognition of characters, multistage recognition of characters.

1 Introduction

With the spread of computers in public/private organizations and individual homes, automatic processing of tabular forms, bank checks, and postal mail is rapidly gaining importance in India. However, such automation needs research and development of handwritten character recognition methodology for Indian scripts. Also, use of mixed scripts is a part of the scope of the present problem. For example, in Fig. 1, two real-life handwritten documents are shown. In each of them, Bangla and English numerals have been mixed to write certain numeric information.

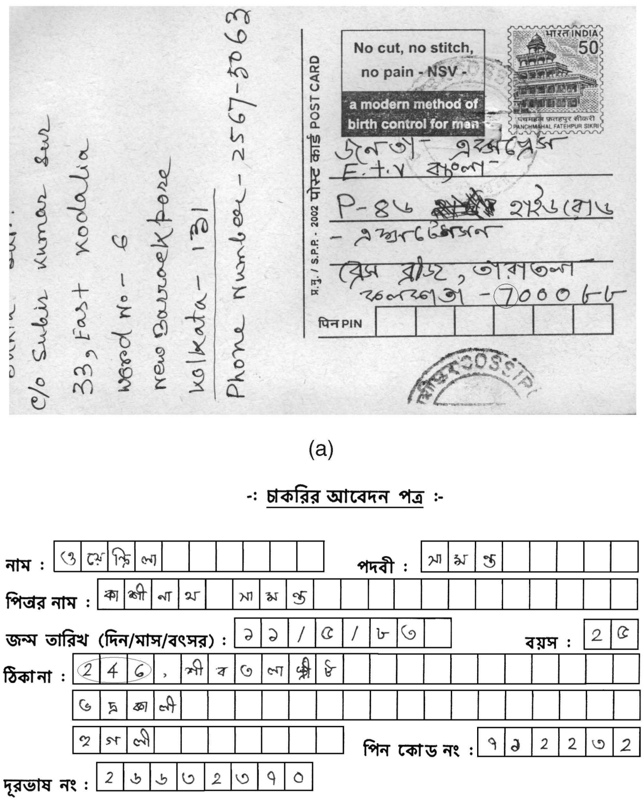
A major obstacle to research on handwritten character recognition of Indian scripts is the nonexistence of standard/benchmark databases. Previous studies were reported on the basis of small databases collected in laboratory environments. However, any fruitful work in this area primarily needs a benchmark database. Several standard databases, such as NIST, MNIST [1], CEDAR [2], and CENPARMI, are available for Latin numerals. Similar databases like [3], [4], and [5] exist for a few other scripts also. On the other hand, to the best of our knowledge, the two databases described in the present paper are the first of its kind for Indian scripts.

• *The authors are with the Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, 203, B. T. Road, Kolkata-700108, India. E-mail: {ujjwal, bbc}@isical.ac.in.*

Manuscript received 20 June 2007; revised 19 Jan. 2008; accepted 25 Feb. 2008; published online 9 Apr. 2008.
Recommended for acceptance by L. O'Gorman.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-06-0374.
Digital Object Identifier no. 10.1109/TPAMI.2008.88.

Extensive experimentations toward recognition of offline handwritten characters, in particular numerals, of different scripts were carried out during the last few decades [6]. Some of the available works include [7] for English, [8] for Chinese, [9] for Arabic, [10] for Korean, and [11] for Kanji script. However, similar experiments were not performed on handwritten numerals of Indian scripts. On the other hand, Hindi and Bangla, the two most popular languages of India, are spoken by, respectively, 500 and 220 million people all over the world. Devanagari is the script of several Indian languages, including Hindi. Also, Bangla is the script of two other Indian languages, viz., Assamese and Manipuri, and it is the official language/script of Bangladesh, a neighboring country of India.

In the literature, a number of approaches are available for preprocessing, feature extraction, classification, and postprocessing. However, two critical issues of developing a handwriting recognition system are selection of a feature set and designing a classifier. Related surveys are found in [12], [13], [14], [15]. The state-of-the-art classifiers include statistical classifier such as modified quadratic discriminant function (MQDF) [16], neural classifiers such as multilayer perceptron (MLP) [17], radial basis function (RBF) classifier [18], polynomial classifier (PC) [19], learning vector quantization (LVQ) classifier [20], and support vector machine (SVM) classifier [21]. Several classifiers such as MLP, RBF, MQDF, PC, and LVQ are efficient with respect to both memory and computational cost. On the other hand, although SVM is capable of delivering high recognition accuracy, it is computationally expensive [22]. Also, LeCun et al. [1] proposed a convolutional neural network architecture which provides high recognition accuracy;



(a)

(b)

Fig. 1. Examples of handwritten documents where mixed numerals (English numerals are circled) have been used: (a) a postcard and (b) a job application form.

however, it is a complicated architecture consisting of seven layers. Liu et al. [23] performed an important comparative study of performance of several classifiers on handwritten digit recognition.

The state-of-the-art features used in handwritten character recognition are the different variants of direction feature [24]. These include chaincode [25], gradient [26], and curvature features [27]. Gabor transform [28] and statistical/structural features [29] have also been successfully used in character recognition research. The directional features are usually extracted from the skeleton or contour or gradient maps of binary image of the input character. The gradient feature can be extracted from both the gray-scale and binary image. Shi et al. [27] considered a composite feature by combining gradient and curvature feature vectors.

Use of multiresolution analysis for character recognition is found in [30] and [31]. Favata et al. [30] implemented a multiresolution approach to feature generation by computing the features at three ranges: local, intermediate, and global. The dynamic character recognizer proposed by Park et al. [31] starts with features extracted at a coarse resolution while finer resolution of a subimage is considered on each successive recursive passes until the classification meets certain acceptance criteria. Cao et al. [32] used a multistage classifier system comprising two different neural networks for recognition of handwritten numerals in two stages.

Obtaining high recognition accuracy is often difficult for a single classifier due to the enormous variability in the handwritten shapes. A few possible solutions to this problem have been discussed in [33]. One solution is to employ a cascade of classifiers with a rejection strategy. Giusti et al. [34] presented a two-stage recognition system in which the patterns rejected by the first classifier were classified by the second classifier. Nunes et al. [35] used a cascade of classifiers along with a recognition scheme having low computational complexity.

One of the main objectives of the present work includes development of a recognition scheme for handwritten numerals of Indian scripts which is capable of providing high recognition accuracy. Toward this goal, we propose a novel scheme which implements wavelet filter-based multiresolution analysis of input numeral images in a cascaded manner. A set of wavelet filters can capture the details of an image at multiple resolutions. Thus, such a scheme is suitable for a coarse-to-fine recognition strategy. Also, real-life images are composed of large areas of uniform gray-value distribution but sharp changes at object edges and our biological vision is sensitive to such edges. With the help of wavelet analysis, we make an attempt to exploit the spatial and temporal aspects of the above fact. Moreover, several wavelets like Daubechies wavelet, as a problem-solving tool, fit efficiently with digital computer with its basis functions defined by multiplication and addition operators —there are no derivatives or integrals involved.

Intermediate results of our multiresolution analysis based on wavelet filter for recognition of Bangla numerals were reported in [36] and [37]. There, we used pixel values of the low-frequency (smooth-smooth) component of wavelet-filtered image as features, while, in the present work, we consider high-level features based on contour representations of all the four frequency components (high-high, high-low, low-high, and low-low) of the wavelet-filtered image. Further, in this work, we do not use wavelet transform coefficients but employ the convolved images in further processing. Also, here we have used an adaptive strategy for combination of outputs at different resolution levels when it is rejected at each of these levels. Finally, this work deals with multiple Indian scripts on a single document.

We use a distinct MLP classifier at each stage of our recognition scheme. Each such classifier either classifies or rejects an input numeral at the corresponding resolution level. If a numeral is rejected by the MLP classifier at a coarser resolution level, the classifier of the following stage attempts to recognize it at the next higher resolution level. Finally, if rejection still occurs at the highest resolution level, the output vector of each of these three MLP classifiers is transformed into a kind of likelihood measurement. We use another MLP classifier to obtain the final decision by combining these three likelihood measurement vectors. Combination of MLP outputs in the above manner is justified because an individual MLP classifier acts as a character classifier as well as a feature extractor [38].

We implemented the above scheme for recognition of handwritten numerals of Devanagari, Bangla, and English. The same has also been simulated for the mixed numeral situation. Experimental results are provided in Section 4.

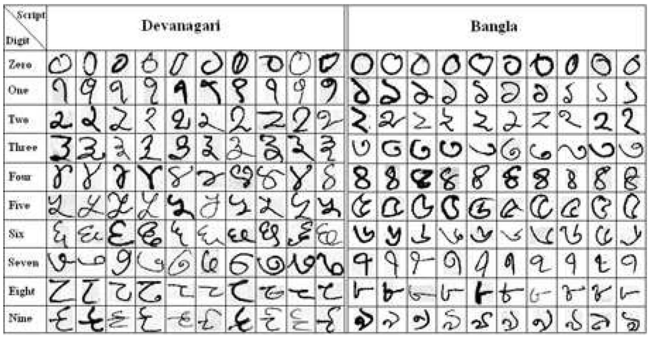


Fig. 2. Handwritten numeral samples from the two databases (10 samples per class are shown).

These are based on handwritten numeral databases described in Section 2 for Devanagari and Bangla scripts and the MNIST database [1] for English. Details of the recognition scheme are described in Section 3. Section 5 concludes this paper.

2 Handwritten Database

2.1 Brief Survey of Handwritten Character Recognition Research for Indian Scripts

In one of the earliest available works [39] on handwriting recognition for an Indian script, a small set of handprinted Devanagari character samples were used. Among later studies, Ramakrishnan et al. [40] considered independent component analysis for feature extraction from handwritten Devanagari numeral images. In a recent work, Bajaj et al. [41] combined decisions of multiple classifiers for handwritten Devanagari numerals. However, the above studies are based on small-sized databases containing little shape variations and cannot be considered as representative sample sets. On the other hand, Bhattacharya et al. [42] described a hybrid scheme, while Bhowmick et al. [43] used stroke features for recognition of handwritten Bangla numerals and characters, respectively. Hanmandlu and Murthy [44] proposed a fuzzy model-based scheme for recognition of handwritten Devanagari numerals. In [45], Chinnuswamy and Krishnamoorthy presented a scheme for recognition of handprinted Tamil characters which considered labeled graphs describing the structural composition of a character in terms of primitives and their relational constraints. Sukhaswami et al. [46] used Multiple Neural Network Associative Memory for recognition of both printed and handwritten Telugu characters.

A few notable recognition studies on printed Indian scripts include [47] and [48] for Devanagari text, [49] for skew angle detection of Indian script documents, [50] and [51], respectively, for printed Devanagari and Bangla OCR systems.

2.2 Indian Script Handwritten Numeral Database Generation

A significant contribution of the present work is the pioneering development of two large databases for handwritten numerals of Devanagari and Bangla. An intermediate report on these two databases can be found in [52]. We provide several details of these two databases.

TABLE 1
Distribution of Numeral Samples in the Two Databases

Script	Set	0	1	2	3	4	5	6	7	8	9	Total
Devanagari	Training	1842	1892	1892	1883	1877	1890	1870	1870	1888	1890	18794
	Test	369	378	378	377	375	378	374	378	377	378	3762
	Total	2211	2270	2270	2260	2252	2268	2244	2248	2265	2268	22556
Bangla	Training	1933	1945	1945	1956	1945	1933	1930	1928	1932	1945	19392
	Test	400	400	400	400	400	400	400	400	400	400	4000
	Total	2333	2345	2345	2356	2345	2333	2330	2328	2332	2345	23392

2.2.1 Database Description

The handwritten Devanagari numeral database consists of 22,556 samples written by 1,049 persons, while the Bangla numeral database consists of 23,392 samples written by 1,106 people. A few samples from each of the two databases are shown in Fig. 2. Each database is randomly divided into respective training and test sets approximately in the ratio of 5:1. The details of this division of both the databases are given in Table 1. Here, it may be mentioned that one may divide these databases into respective training and test sets in other ratios to experiment with related effect on recognition performance. In fact, in Section 4, we provided experimental results for two cases where each database is divided approximately in the ratios 4:1 and 5:1. We do not provide any separate validation set of samples. If required, one may partition the training set to get a validation set as is done by us (see Sections 4.1 and 4.2) in the proposed recognition scheme.

2.2.2 Data Collection

Samples of the above databases have been collected using three different types of handwritten documents, viz., postal mail (Fig. 1a), job application forms (Fig. 1b), and a tabular form (Fig. 3) designed for such data collection purpose.

The Devanagari numeral database includes samples collected from 368 mail pieces and 274 job application forms, while Bangla database includes samples from 465 mail pieces and 268 job application forms. We used the form document shown in Fig. 3 for collecting the rest of the samples.

In approximately 75 percent of the cases, the same subject was asked to write on four different occasions, while others wrote on two to three different occasions. We did not disclose the data collection purpose to them and the only restriction imposed was to write one numeral in one rectangular box.

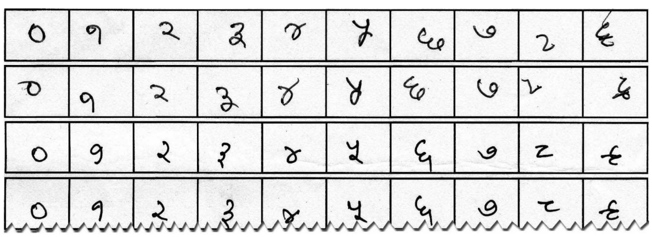


Fig. 3. Specially designed form for data collection.

TABLE 2
Benchmark k Nearest Neighbor Classifier Result on the Two Databases

Script	LL component (16 × 16) of wavelet filtered pixel image						64-dimensional chaincode histogram feature					
	$k=1$	$k=3$	$k=5$	$k=7$	$k=9$	$k=15$	$k=1$	$k=3$	$k=5$	$k=7$	$k=9$	$k=15$
Bangla	93.32	92.87	92.60	92.42	92.15	91.92	95.50	96.17	95.86	95.73	95.61	95.45
Devanagari	95.08	94.87	94.82	94.66	94.50	93.68	97.20	97.26	96.78	96.66	96.63	96.36

2.2.3 Variability in Data Sources

Samples of the present databases had been collected from a wide spectrum of population which includes students (school, college, university, and research), workers (government, private, and self-employed), housewives, unemployed, and retired persons.

Samples collected from real-life postal mail were written with varieties of source pen and paper. Two other types of form documents (shown in Figs. 1b and 3) were printed using three types of papers of 56, 70, and 80 grams per square meter (gsm). While filling these forms, a wide variety of pens was used. These include roller ball pens and gel pens both with 0.7 and 0.5 mm tips, fine liner pens with 0.8 mm polyacetal tip, and ball point pen with medium tip. For the color of the ink, only blue and black were allowed.

2.2.4 Data Format

Source documents were scanned at 300 dpi using an HP flatbed scanner and stored as grayscale (TIF format) images in 1 byte per pixel. Automatic extraction of isolated numeral samples from these scanned documents was done by a software package. Since such software may lead to error and cannot detect possible presence of Latin numerals (due to possible mixed script situations of the Indian context), all the output samples were checked manually and necessary editing/removal was done.

Image samples of the present databases are also maintained in the original form (source scanned documents). The writer information is available for all of these scanned documents except the postal mail.

2.2.5 Detailed Statistics of Sample Data

A number of primary statistics (min, max, average, standard deviation, and three quartiles) of class-wise distributions of various features of handwritten samples (such as variations in size, stroke-widths, contrast, and slants) were computed for both the training and test sets of the two databases. For the size of each sample, we computed four values. These are height, width, volume, and aspect ratio. A directional chain code method has been used before [53] for estimation of slants of handwritten words. However, in the present work, we have used the angle between the vertical and the closest principal axis of the shape represented by the contour of individual samples. These statistics for Devanagari and Bangla databases are provided in Appendices A and B, respectively.

2.2.6 Benchmark Nearest Neighbor Classifier Result

We computed a benchmark k nearest neighbor classifier result on the present databases. These results are obtained for $k = 1$, 3, 5, 7, 9, and 15. These classification results are based on two different feature vectors such as smooth-smooth component (16×16) of wavelet-filtered pixel image and 64-dimensional chain code histogram feature [25] computed from the contour representations of numeral shapes. These results are shown in Table 2.

2.2.7 Comparison with Existing MNIST Database

Both databases of Indian scripts consist of handwritten numeral samples taken from the real life, similar to the well-known MNIST database for handwritten English numerals. However, our databases differ from the MNIST database in several aspects such as 1) samples are available as original gray-level images, 2) images are not size-normalized, and 3) samples had been collected from a wider group of population, which includes students (school, college, university, and research), workers (government, private, and self-employed), housewives, unemployed youths, and retired people.

2.2.8 Problems Unique to Indian Scripts

Devanagari script is used in writing different local Indian languages, and region-specific minor variations in the shape of the numerals written by older people are sometimes observed. However, such variations are less frequent for newer generations. An interesting aspect of handwritten documents in Indian scripts is that they often have one or more entries written in English. Some of its possible reasons are

1. India is a multilingual country with colonial past,
2. English is one of its official languages,
3. at the school level, in existing education systems, either two or three languages are taught, one of which is usually English,
4. at the high school level, text books written in Indian scripts often have entries in English, especially the numerals, and
5. Indian currencies are often written in English.

We observed that while writing a numeric information (such as pin code, phone number, age, etc.) in an Indian script, the writer may casually enter one or more English numerals, causing a mixed-script situation as shown in Fig. 1.

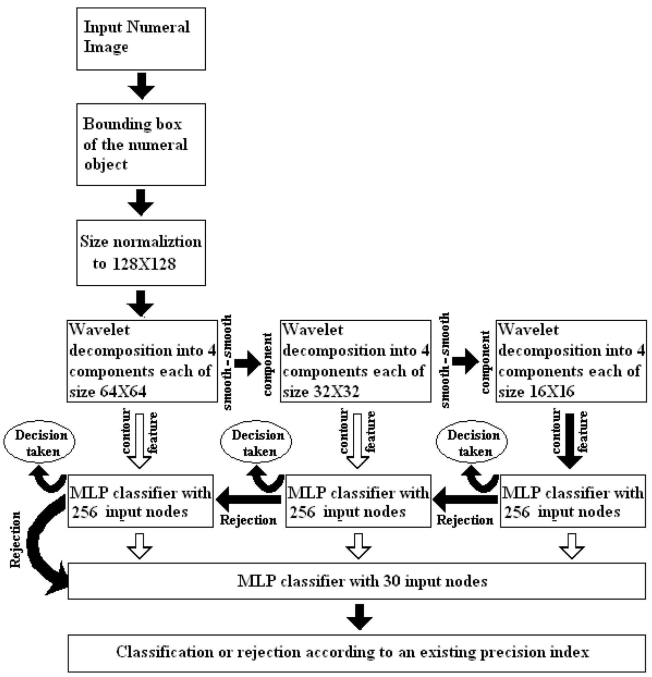


Fig. 4. Block diagram of the proposed multistage recognition scheme for handwritten numerals of Indian scripts.

2.2.9 *Generation of Synthetic Training Data*

Training samples of Devanagari and Bangla numeral databases are subjected to the following two distortion transformations, thereby increasing their number by a factor of 10. They are as follows:

Rotation. From each training sample image, four others are generated by rotation. Two new samples are created by rotating an input sample through two random but different angles in the range $+5^\circ$ to $+10^\circ$ and two others are similarly created through random rotation in the range of -5° to -10°. Thus, at the end of this step, the total volume of the present databases is increased five times.

Blurring. All training samples (original and rotated ones) at the end of above step is blurred by applying a Gaussian blurring kernel ($\mu = 0.75$ and $\sigma = 0.33$). Thus, execution of this step multiplies the volume of training samples by two (with and without blurring).

In this way, the available sizes of training sets become 187,940 for Devanagari (the number of original training samples is 18,794) and 193,920 for Bangla (the number of original training samples is 19,392).

3 Multistage Recognition Scheme

A block diagram of the present recognition scheme is shown in Fig. 4. In addition to preprocessing, the present scheme has two recognition stages and the first stage involves a cascade of three MLP classifiers. If a decision about the possible class of an input numeral image cannot be reached by any of the MLPs of the first stage, then certain estimates of its class conditional probabilities obtained from these classifiers are fed to another MLP of the second stage. In this second stage, the numeral image is either classified or rejected according to a precision index [5].

3.1 Preprocessing and Featuring

We first binarize an input numeral image by Otsu's [54] thresholding method. The minimum bounding rectangle of the binarized image is also computed for further processing. It is then linearly normalized to the size 128×128. Reasons for this choice of size are as follows: 1) The Daubechies wavelet filter, which we apply in the next step, requires the size of the input image to be $2^n \times 2^n$, $n \in I^+$. 2) In the proposed scheme, we use images of an input numeral at three successive resolutions produced by repeated application of Daubechies filter. 3) If an input numeral image is initially normalized to the size 64×64, then, after three successive applications of Daubechies filter, the size of each component image is reduced to the size 8×8. However, there are many samples in our databases, when reduced to such a small size, their visual (manual) recognition become difficult.

3.1.1 *Wavelet Filter*

Orthonormal wavelet filters are capable of providing an elegant tool for multiresolution analysis of handwritten numeral images. Such a tool decomposes a function into a hierarchy of several levels of resolution making a coarse-to-fine recognition scheme possible. Use of an input numeral image at different resolutions characterizing different shape structures often helps to get rid of ambiguity and increase correct recognition accuracy. Also, a wavelet filter, consisting of linear operations is computationally fast and suitable for real-life applications.

The basis functions of the wavelet filter are called wavelets. There exist infinitely many possible sets of wavelets, and different sets of wavelets make tradeoffs between how compactly they are localized in space and how smooth they are.

The first and simplest possible orthogonal wavelet system is the Haar wavelet. However, Daubechies [55] constructed a set of orthonormal wavelet basis function that is compactly supported in the time domain and has good frequency-domain decay. In the present work, we have chosen Daubechies wavelet filter due to these reasons. The simplest member of this family is the Daubechies-4 wavelet, which has only four coefficients, given by,

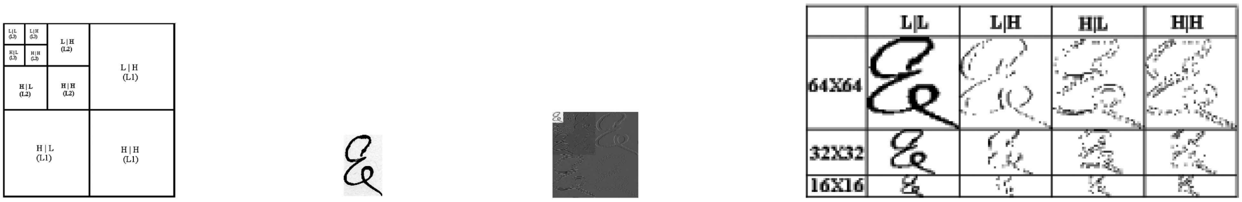
$$l_0 = \frac{1+\sqrt{3}}{4\sqrt{2}}, l_1 = \frac{3+\sqrt{3}}{4\sqrt{2}}, l_2 = \frac{3-\sqrt{3}}{4\sqrt{2}}, l_3 = \frac{1-\sqrt{3}}{4\sqrt{2}}.$$

These coefficients form a set of low-pass or smoothing filter L and another set of four coefficients,

$$h_0 = l_3, h_1 = -l_2, h_2 = l_1, \text{ and } h_3 = -l_0,$$

constitutes the high-pass filter H. (In signal processing contexts, L and H are called quadrature mirror filters.)

An input signal x is split into a low-pass or smooth component x_0 and a high-pass or detail component x_1 by the low-pass filter L and the high-pass filter H, respectively. Both components are downsampled in the ratio 2:1. The low-pass component x_0 is then split further into x_{00} and x_{01} as above for the second time and they are again downsampled in the ratio 2:1. This process (pyramidal algorithm [56]) of splitting and downsampling is continued as far as



(a) (b) (c) (d)

Fig. 5. (a) Layout of wavelet filtering (L $\rightarrow$ low-pass filter, H $\rightarrow$ high-pass filter, $j \rightarrow$ level no.); (b) original image of a handwritten Devanagari numeral ("6") sample; (c) results of successive application of wavelet filter (according to the layout in (a)) on the sample in (b) after its normalization to the size 128 $\times$ 128; (d) four components (binarized) of wavelet-filtered images at three resolution levels (64 $\times$ 64, 32 $\times$ 32, and 16 $\times$ 16).

required or until a trivial size (usually 2) of the smooth-smooth component is reached.

Extension of the above principle to 2D array is simple [56]. Wavelet filtering of an input image array (2D) is done on the row index first and then on the column index. Since each application of the filter consists of multiplication by an orthogonal matrix, due to the associativity, the result is independent of the order in which the rows and columns are considered.

The Daubechies wavelet filter can be applied on an image matrix of size $M \times M$, where M is a power of 2. The size of each component of the filtered image is $\frac{M}{2} \times \frac{M}{2}$. The layout of application of such wavelet filter recursively on an image is shown in Fig. 5a. The $L|L$, $L|H$, $H|L$, and $H|H$ components of the wavelet filter at each resolution level correspond to low frequencies in both horizontal and vertical directions, low frequency in horizontal and high frequency in vertical directions, high frequency in horizontal and low frequency in vertical directions, and high frequencies in both horizontal and vertical directions, respectively.

3.1.2 Multiresolution Representation of Numeral Image

Initially, a wavelet filter is applied on the size normalized image (128 $\times$ 128) of input sample. This produces four components, $D^{(1)}_{L|L}$, $D^{(1)}_{L|H}$, $D^{(1)}_{H|L}$, and $D^{(1)}_{H|H}$, each of size 64 $\times$ 64, corresponding to high and low frequencies in horizontal and vertical directions. The smooth-smooth component $D^{(1)}_{L|L}$ is used for second-time application of wavelet filter. As before, we get another set of four components, $D^{(2)}_{L|L}$, $D^{(2)}_{L|H}$, $D^{(2)}_{H|L}$, and $D^{(2)}_{H|H}$, each of size 32 $\times$ 32. Again, the smooth-smooth component $D^{(2)}_{L|L}$ is used for the third and last time, producing another set of four components, $D^{(3)}_{L|L}$, $D^{(3)}_{L|H}$, $D^{(3)}_{H|L}$, and $D^{(3)}_{H|H}$, each of size 16 $\times$ 16.

Thus, by successive application of wavelet filter three times, we get three sets of four image components corresponding to three fine-to-coarse resolution levels. An example of the resulting image corresponding to an input sample (Fig. 5b) is shown in Fig. 5c. Also, its four components (after binarization) at three resolution levels are shown separately in Fig. 5d.

3.1.3 Feature Extraction

We compute features from the above components of wavelet-filtered image at different resolutions as follows.

Chain code histogram features [25] are obtained corresponding to each of the detailed image components ($D^{(k)}_{L|L}, k = 1, 2, 3$). This is achieved by dividing the bounding box of chain code representation of the contour of each $D^{(k)}_{L|L}$ into seven equal horizontal and vertical strips saving the rightmost vertical or bottommost horizontal strip which have one more row or column, respectively. Thus, the chain code representation of each $D^{(k)}_{L|L}$ is divided into 49 rectangular blocks. In each block, a local histogram of the chain codes is calculated. Since the eight directions $\{0, 1, \ldots, 7\}$ along the contour should be effectively quantized into one of four possible values, viz., 0 or 4, 1 or 5, 2 or 6, and 3 or 7, the histogram of each block has four components and each of them is divided by the product of row and column of the corresponding block. Thus, we obtain a 196 ((7×7) blocks $\times$ 4 directions) component normalized feature vector corresponding to each $D^{(k)}_{L|L}$. The dimension of this feature vector is reduced to 64 ((4×4) blocks $\times$ 4 directions) by downsampling with a 3 $\times$ 3 Gaussian filter.

For computation of features from $D^{(k)}_{L|H}$, $D^{(k)}_{H|L}$, and $D^{(k)}_{H|H}$, $k = 1, 2, 3$, we divide each bounding box of each of them into 8 $\times$ 8 equal blocks, and the number of black pixels in each of them is computed. Each of these numbers is divided by the product of row and column numbers of each block for normalization. Thus, each component provides a feature vector of 64 dimensions.

Finally, at each resolution level $k = 1, 2, 3$, we concatenate the feature vectors corresponding to $D^{(k)}_{L|L}$, $D^{(k)}_{L|H}$, $D^{(k)}_{H|L}$, and $D^{(k)}_{H|H}$, $k = 1, 2, 3$ from left to right and obtain a 256-dimensional feature vector.

3.2 Classification Strategy

In the proposed recognition scheme, classification starts at the resolution level (16 $\times$ 16), and in case of rejection, it is attempted at higher resolutions (32 $\times$ 32) and finally at (64 $\times$ 64). We use a distinct MLP classifier at each resolution level. Successful training of these MLP classifiers is a crucial factor for satisfactory performance of the recognition scheme. The rejection criterion for each level is set by minimizing misclassifications at the individual levels. This is achieved empirically and on the basis of a validation set. On the other hand, for the MLP classifier of the final stage that combines outputs of three classifiers of first stage, the rejection criterion is set according to the IPTP competitions in Japan [5]. According to this criterion, the cost of an error is 10 times that of a rejection, and the value of $index = 10 \times error + rejection$ is minimized.

3.2.1 MLP Architecture Selection

The input layer of each MLP classifier of first stage consists of 256 nodes, while the MLP of the second stage has 30 input nodes. The number of output nodes for each of the four MLPs is 10. On the other hand, the choice of an optimal size of hidden layer for a classification task has been studied in a number of works [57], [58], but a rigorous generalized solution could not be found. However, since the ultimate objective of the present situation is to achieve an acceptable rate of correct recognition [59], the choice is made through extensive simulations with different choices of the number of hidden nodes. The minimum number used is 10 and the maximum number is 50 at an interval of 5. For each choice, we obtained the recognition performance of the concerned MLPs on the validation set and the number of hidden nodes providing the best recognition accuracies were used for reporting experimental results in the present paper. The optimal number of hidden nodes for first and second stages are 30 and 15, respectively.

3.2.2 MLP Training

A well-known backpropagation (BP) algorithm [17] is used for training the MLP classifiers. It is well known that performance of an MLP classifier largely depends on the amount of training—both inadequate and overtraining are detrimental to the performance.

We address the above problem by keeping aside a small subset of training samples for validation and use the remaining samples for BP training. Here, the BP training is continued until its recognition performance on the validation set improved.

During BP training of an MLP, its connection weights are iteratively obtained by minimizing certain system error [17]. However, there is no guarantee that the global minimum can be reached. However, a good local minimum may be obtained through proper selection of various parameters (learning rate, momentum factor). Small values to these parameters usually lead to a good local minimum, although the speed of training becomes very slow. During our simulations, we used $\text{learning rate} = 0.1$ and $\text{momentum factor} = 0.05$.

3.3 Combination of Outputs of Three MLPs

When an input sample is rejected at each of three resolutions, the outputs from three MLPs are combined. Each of these three sets of outputs $\{o_{ij}, j = 1, \ldots, 10\}$ for $i = 1, 2, 3$ is converted into certain likelihood measures $\{L_{ij} = o_{ij} / \sum_j o_{ij}, j = 1, \ldots, 10\}$ for $i = 1, 2, 3$. These likelihood measures are concatenated in a left to right fashion to form the input pattern for the MLP of the final stage. For training of this second-stage MLP, we consider only those training samples which have been rejected by all three MLPs of the first stage. The rejection criteria for this MLP are set empirically on the basis of validation samples (rejected during the first stage) and the criterion in [5]. In fact, for each sample fed to the MLP classifier of the final stage, we compute the difference between the first two maximum responses at the output nodes of the MLP. If this difference is less than a threshold T, then we reject the sample. Otherwise, the input sample is classified according to the output node with maximum response. To obtain an optimal value of T, we considered several choices of T in the range 0.05-0.5 at an interval of 0.025, and for each T, we computed the $Index = 10 \times (no.\ of\ misclassifications) + (no.\ of\ rejections)$ on the set of validation samples fed to this MLP. The value of T which minimizes the $Index$ is selected.

TABLE 3
Performance Statistics of Different Stages of the Proposed Scheme on the Devanagari Numeral Database

Dataset	Stage	No. of input samples	Recognition accuracy		
			Correct	Rejection	Substitution
Train	16X16 resolution	16794	12726	4029	39
	32X32 resolution	4029	1735	2272	22
	64X64 resolution	2272	760	1508	4
	Final combination	1508	1451	54	3
	Overall		16672	54	68
	Overall (%)		99.27	0.32	0.41
Test	16X16 resolution	3762	2842	909	11
	32X32 resolution	909	315	587	7
	64X64 resolution	587	189	395	3
	Final combination	395	380	9	6
	Overall		3726	9	27
	Overall (%)		99.04	0.24	0.72

4 Experimental Results

We simulated the proposed recognition scheme on handwritten numeral databases of three major Indian scripts, viz., Devanagri, Bangla, and English. For Devanagari and Bangla, we used databases described in Section 2.2 while we have taken the MNIST [1] database for English numerals. Also, we simulated this recognition scheme in two mixed script situations, viz., Devangari-English and Bangla-English. As stated before, such multiscript and mixed-script situations may appear in various forms. Mail pieces may be written in Devanagari/Bangla or some other native Indian script depending upon the place of origin of the document and also English, officially recognized as an Indian script. In several situations, a part of a handwritten document may be written in some native Indian script while English is used in another part of the same document. In a third situation, while writing a PIN code or telephone number, an Indian writer may use mixture of English and some other Indian script numerals at random. Examples of two such situations are already shown in Fig. 1.

4.1 Results on Devanagari Numeral Database

The original training and test sets of Devanagari numeral database consist of 18,794 and 3,762 samples, respectively. We randomly selected 2,000 training samples (200 from each class) to form the validation set and the remaining 16,794 samples are used for training purpose. On using the distortion transformations described in Section 2.2, the available volumes of training and validation sets of Devanagari database are increased to 167,940 and 20,000, respectively.

The proposed multistage recognition scheme provided 99.27 percent and 99.04 percent recognition accuracies, respectively, on the original (excluding synthetically distorted ones) training and test sets of Devanagari numeral database. The detailed recognition results are provided in Table 3. To the best of our knowledge, the above recognition

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. 6. Devanagari (a) "0" misclassified as "7"; (b) "1" misclassified as "7"; (c) "2" misclassified as "5"; (d) "3" misclassified as "2"; (e) "4" misclassified as "5"; (f) "5" misclassified as "2"; (g) "6" misclassified as "9"; (h) "7" misclassified as "0"; (i) "8" misclassified as "2"; and (j) "9" misclassified as "8".

performance is better than other available recognition experiments on handwritten Devangari numerals. For example, a fuzzy model-based approach [44] claims 95 percent overall recognition rate, a moment-based approach [60] provides 87 percent accuracy, a multiple classifiers-based approach [41] reports 89.6 percent accuracy, and another approach [61], which combined ANN and HMM classifiers, provided 92.83 percent accuracy.

A few misclassified test samples of the Devanagari database are shown in Fig. 6.

The results shown in Table 3 correspond to the division of the whole Devanagari database into training and test sets in the ratio 5:1. We also obtained simulation results on this Devanagari database when it had been randomly divided into training and test sets, approximately in the ratio 4:1. In this situation, the training, validation, and test sets of Devanagari numeral database consist of 16,044, 2,000, and 4,512 original sample images, respectively. Using distortion transformations as before, volumes of training and validation sets are increased to 160,440 and 20,000, respectively. In this situation, the proposed recognition scheme provided 99.29 percent and 98.23 percent accuracies with respect to the corresponding (excluding synthetically distorted ones) training and test samples.

4.2 Results on Bangla Numeral Database

There are 19,392 and 4,000 training and test samples in the Bangla numeral database. We randomly selected 2,000 training samples (200 from each class) to form the validation set and the remaining 17,392 training samples are actually used for training purpose. On applying the distortion transformations of Section 2.2, the available sizes of training and validation sets of Bangla numeral database become 173,920 and 20,000, respectively.

The present recognition scheme correctly recognizes 99.14 percent and 98.20 percent of the original (excluding synthetically distorted ones) training and test samples of Bangla numeral database. The detailed recognition results at various stages are provided in Table 4. To the best of our knowledge, this recognition performance is also better than other studies that include vector skeleton-based approach [42] obtaining 93.26 percent accuracy on test samples and a direction code histogram feature-based approach [62] providing a maximum of 96.93 percent accuracy on the test samples.

A few misclassified test samples from the Bangla database are shown in Fig. 7.

The results of Table 4 correspond to the division of the whole Bangla numeral database into training and test sets in the ratio 5:1. However, we also obtained simulation results on the same Bangla numeral database when it is randomly divided into training and test sets approximately in the ratio 4:1. In this case, the training, validation, and test sets of Bangla numeral database consist of 16,714, 2,000, and 4,678 original sample images, respectively. Using distortion transformations as before, volumes of training and validation sets are increased to 167,140 and 20,000, respectively. The proposed recognition scheme provided 99.26 percent and 98.01 percent recognition accuracies, respectively, on the corresponding training (excluding synthetically distorted samples) and test sets.

TABLE 4
Performance Statistics of Different Stages of the Proposed Scheme on the Bangla Numeral Database

Dataset	Stage	No. of input samples	Recognition accuracy		
			Correct	Rejection	Substitution
Train	16X16 resolution	17392	13377	3908	107
	32X32 resolution	3908	1919	1979	10
	64X64 resolution	1979	731	1242	6
	Final combination	1242	1216	13	13
	Overall		17243	13	136
	Overall (%)		99.14	0.08	0.78
Test	16X16 resolution	4000	2897	1060	43
	32X32 resolution	1060	373	682	5
	64X64 resolution	682	268	411	3
	Final combination	411	390	7	14
	Overall		3928	7	65
	Overall (%)		98.20	0.18	1.62

4.3 Results on MNIST English Numeral Database

There are 60,000 and 10,000 training and test samples in the MNIST database for handwritten English numerals. Six thousand training samples (600 from each class) are randomly taken to generate the validation set and the remaining 54,000 training samples are actually used for training purposes.

The present recognition scheme misclassifies 0.81 percent and 1.01 percent of the training and test samples of handwritten English numerals. The detailed recognition results at different stages are provided in Table 5.

4.4 Results on Devanagari-English Mixed Numeral Data Set

As discussed in Section 2.2, mixed recognizers (such as Devanagari-English or Bangla-English) have some relevance in the Indian context. In the literature, we did not find any other recognition study of such mixed numeral problem. We developed two recognizers for Devanagari-English mixed numerals. On one hand, we consider it a 17-class problem, while, on the other hand, we look at it as a 20-class (10 + 10) problem.

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. 7. Bangla (a) "0" misclassified as "3"; (b) "1" misclassified as "4"; (c) "2" misclassified as "4"; (d) "3" misclassified as "0"; (e) "4" misclassified as "1"; (f) "5" misclassified as "0"; (g) "6" misclassified as "3"; (h) "7" misclassified as "2"; (i) "8" misclassified as "6"; and (j) "9" misclassified as "1".

TABLE 5
Performance Statistics of Different Stages of the Proposed Scheme on the MNIST English Numeral Database

Dataset	Stage	No. of input samples	Recognition accuracy Correct	Rejection	Substitution
Train	16X16 resolution	54000	42594	11105	301
	32X32 resolution	11105	5442	5630	33
	64X64 resolution	5630	1780	3831	19
	Final combination	3831	3693	53	85
	Overall		53509	53	438
	Overall (%)		99.09	0.10	0.81
Test	16X16 resolution	10000	7840	2092	68
	32X32 resolution	2092	757	1316	19
	64X64 resolution	1316	530	778	8
	Final combination	778	752	20	6
	Overall		9879	20	101
	Overall (%)		98.79	0.20	1.01

TABLE 6
Performance Statistics of Different Stages of the 17-Class Mixed (Devanagari-English) Numeral Recognizer

Dataset	Stage	No. of input samples	Recognition Accuracy Correct	Rejection	Substitution
Devanagari Test	16X16 resolution	3762	2766	982	14
	32X32 resolution	982	370	601	11
	64X64 resolution	601	208	386	7
	Final Combination	386	364	15	7
	Overall		3708	15	39
	Overall (%)		98.56	0.40	1.04
English Test	16X16 resolution	10000	7653	2269	78
	32X32 resolution	2269	746	1501	22
	64X64 resolution	1501	622	868	11
	Final Combination	868	826	34	8
	Overall		9847	34	119
	Overall (%)		98.47	0.34	1.19

4.4.1 Seventeen-Class Devanagari-English Mixed Numeral Recognizer

The shapes of three English numerals, 0, 2, and 9, have significant similarity with Devanagari numerals 0, 2 and 1, respectively. So, here we consider a $17\ (= 10 + 10 - 3)$ class recognizer. Now, training samples in each class are selected as follows: 1) There is a mismatch in number of training samples between Devanagari and English. The Devanagari database has a little more than 1,800 training samples per class (see Table 1) while, for English, there are 6,000 training samples per class. To make equal representation of each class in the training set for this mixed script recognizer, additional samples are synthetically generated by randomly rotating the original training samples of seven Devanagari numerals (which do not have shape similarity with some English numeral) three times in the range $+5^\circ$ to $+10^\circ$ or -5° to -10° followed by blurring as described in Section 2. The required numbers of training samples are randomly selected from the transformed samples and added to the original training samples, making the total number of training samples 6,000 per class. 2) A total of 6,000 training samples for each of the three classes corresponding to English 0, 2, and 9 are composed of 3,000 random samples from the corresponding training set of the MNIST English numeral database and another 3,000 random samples from the training sets of Devanagari numeral classes 0, 2, and 1. The above 3,000 training samples of each of the said three Devanagari numeral classes are obtained by the approach as described in 1 above.

The training samples (6,000 per class) developed as above are randomly divided, obtaining separate training, and validation sets consisting of 5,400 and 600 samples per class. The recognition accuracies of this mixed recognizer are 98.56 percent and 98.47 percent on the test sets of Devanagari and English numerals, respectively. See Table 6 for detail.

4.4.2 Twenty-Class Devanagari-English Mixed Numeral Recognizer

In the 20-class Devanagari-English mixed numeral recognizer in which we do not consider fusion of shape-similar classes as above, the training and validation samples for this mixed recognizer have been chosen in a similar way as above. This recognizer provided 77.63 percent and 82.55 percent recognition accuracies on the test sets of Devanagari and English numerals, respectively. A major source of this poor recognition performance is the confusion between shape-similar classes (English numerals 0, 2, and 9 confused with Devanagari numerals 0, 2, and 1, respectively). The modified recognition accuracies obtained after ignoring the misclassifications within the shape-similar classes are 98.22 percent and 98.08 percent on the test sets of Devanagari and English databases, respectively.

4.5 Results on Bangla-English Mixed Numeral Data Set

The handwritten shapes of four English numerals, namely 0, 2, 8, and 9, are almost similar to the handwritten shapes of four Bangla numerals 0, 2, 4, and 7, respectively. So, here we simulate a $16\ (= 10 + 10 - 4)$ class recognizer (by fusing shape similar classes) as well as a 20-class recognizer (no fusion of shape similar classes) for a comparative study.

4.5.1 Sixteen-Class Bangla-English Mixed Numeral Recognizer

The mismatch between the training set sizes of Bangla (approximately 1,900 training samples per class) and English (6,000 training samples per class) is tackled in a similar manner as in Section 4.4. The resulting set consisting of 6,000 training samples per class is further divided, forming a training set consisting of 5,400 samples per class and a validation set consisting of 600 samples per class, as before.

TABLE 7
Performance Statistics of Different Stages of the 16-Class Mixed (Bangla-English) Numeral Recognizer

Dataset	Stage	No. of input samples	Recognition Accuracy Correct	Rejection	Substitution
Bangla Test	16X16 resolution	4000	2819	1135	46
	32X32 resolution	1135	432	692	11
	64X64 resolution	692	261	425	6
	Final Combination	425	410	8	7
	Overall		3922	8	70
	Overall (%)		98.05	0.20	1.75
English Test	16X16 resolution	10000	7656	2271	73
	32X32 resolution	2271	746	1506	19
	64X64 resolution	1506	624	874	8
	Final Combination	874	838	30	6
	Overall		9864	30	106
	Overall (%)		98.64	0.30	1.06

TABLE 8
Various Details of the Samples in the Devangari Numeral Database

		Training Set							Test Set						
		Height	Width	Vol	Aspect ratio	Penwidth	Contrast	Major Axis	Height	Width	Vol	Aspect ratio	Penwidth	Contrast	Major Axis
Class 0	Max	106	79	8374	2.28	10.00	114.17	44.96	91	77	6734	2.20	7	120.89	44.92
	Min	24	20	540	0.40	2.00	38.44	0.02	25	24	648	0.52	2	41.25	0.16
	Avg	48.75	43.37	2153.76	1.15	4.03	62.54	14.33	49.04	44.29	2219.71	1.13	4.20	62.84	15.99
	S.D.	10.62	9.22	785.98	0.26	0.92	11.79	10.99	10.85	9.54	860.58	0.25	0.97	11.66	11.12
	1st quart	41	37	1584.00	0.97	3.00	53.85	5.58	41	37	1620	0.96	4	54.00	6.60
	2nd quart	48	43	2058.00	1.14	4.00	60.67	11.72	48	43	2035	1.13	4	60.69	14.23
	3rd quart	56	49	2622.00	1.32	5.00	69.17	20.83	55	50	2695	1.29	5	70.23	23.32
Class 1	Max	117	68	6468	4.81	7	100.22	44.78	105	72	5429.00	4.67	6	87.11	44.69
	Min	29	15	527	0.90	2	32.85	0.06	36	15	697.00	0.97	1	33.92	0.18
	Avg	67.21	33.26	2289.41	2.09	4.29	58.99	14.14	67.88	33.30	2303.14	2.14	4.33	59.00	14.29
	S.D.	13.84	8.11	883.58	0.52	0.77	10.45	9.11	13.53	8.90	901.16	0.59	0.82	10.76	9.55
	1st quart	57	27	1600.00	1.71	4	51.58	7.15	58	27	1643.50	1.69	4	51.42	6.44
	2nd quart	67	32	2112.00	2.03	4	57.39	13.09	68	32	2110.00	2.07	4	58.49	13.42
	3rd quart	77	38	2829.00	2.41	5	64.92	19.67	78	38	2875.00	2.48	5	66.08	19.90
Class 2	Max	114	117	9918	2.56	11	88.27	44.99	92	111	7568	2.32	9	84.98	44.93
	Min	30	23	874	0.38	2	30.60	0.01	32	28	1258	0.46	2	31.49	0.09
	Avg	60.82	57.58	3524.33	1.14	4.07	48.23	21.81	61.36	56.43	3496.03	1.16	4.09	48.07	22.50
	S.D.	11.29	17.09	1276.79	0.35	1.04	9.77	12.61	10.74	16.47	1275.86	0.34	1.01	9.31	12.54
	1st quart	52	45	2558	0.90	3	40.93	10.95	53	44	2465.5	0.93	3	41.24	12.34
	2nd quart	60	55	3392	1.11	4	45.73	21.64	61	53	3294	1.16	4	45.85	22.53
	3rd quart	68	67	4324	1.37	5	54.47	32.45	69	64.75	4267.5	1.38	5	54.04	32.46
Class 3	Max	113	139	13335	3.05	7	92.58	44.93	110	154	9856.00	3.00	8	79.50	44.83
	Min	40	22	1350	0.50	2	26.97	0.01	38	25	1300.00	0.42	1	31.26	0.05
	Avg	70.71	57.82	4153.15	1.34	3.98	50.84	16.33	70.44	58.68	4171.19	1.32	3.82	49.22	17.07
	S.D.	11.77	20.28	1779.17	0.41	0.91	10.47	11.62	11.38	21.18	1688.67	0.41	0.97	10.96	11.97
	1st quart	63	43	2820	1.04	3	43.26	6.93	63	43	3000.00	1.04	3	40.74	7.60
	2nd quart	70	52	3744	1.33	4	49.94	14.27	69	53	3834.00	1.30	4	47.86	13.93
	3rd quart	78	68	5111.5	1.59	5	57.41	23.15	77	68	5110.00	1.60	4	56.17	24.83
Class 4	Max	147	120	11880	2.45	10	85.07	44.99	108	117	9900.00	2.35	7	83.68	44.92
	Min	23	22	897	0.44	2	28.13	0.04	42	26	1586.00	0.54	3	34.04	0.35
	Avg	61.02	56.17	3526.77	1.16	4.30	51.14	21.34	74.05	60.20	4489.80	1.31	4.47	52.11	22.35
	S.D.	14.49	17.40	1611.92	0.36	0.98	10.60	12.50	11.43	16.69	1497.78	0.37	0.87	9.63	12.83
	1st quart	51	43	2207.25	0.89	4	42.38	10.79	66	48	3432.00	1.02	4	45.44	11.47
	2nd quart	60	54	3245.00	1.09	4	49.98	20.58	73	57	4256.00	1.30	4	50.45	22.35
	3rd quart	70	67	4600.75	1.38	5	58.58	31.50	81	70	5457.50	1.55	5	57.32	33.50
Class 5	Max	114	135	12198	2.65	9	88.42	44.97	108	117	9900.00	2.35	7	83.68	44.83
	Min	33	26	1295	0.49	2	28.68	0.00	42	26	1586.00	0.54	3	34.04	0.77
	Avg	71.93	61.55	4481.67	1.25	4.14	49.90	23.40	74.05	60.20	4489.80	1.31	4.47	52.11	22.68
	S.D.	12.19	17.69	1636.12	0.36	0.94	9.47	12.41	11.43	16.69	1497.78	0.37	0.87	9.63	12.41
	1st quart	64	48	3224.00	0.98	3	43.25	12.92	66	48	3432.00	1.02	4	45.44	12.05
	2nd quart	71	59	4266.00	1.21	4	48.25	23.88	73	57	4256.00	1.30	4	50.45	23.11
	3rd quart	80	73	5544.00	1.48	5	55.13	33.87	81	70	5457.50	1.55	5	57.32	32.40
Class 6	Max	125	123	11000	2.88	8	87.31	45.00	110	110	10400.00	2.72	8	81.85	44.89
	Min	34	25	1508	0.43	1	29.70	0.05	37	25	1624.00	0.47	1	32.99	0.08
	Avg	72.42	58.45	4259.16	1.31	4.09	50.01	25.54	71.48	57.74	4157.32	1.32	4.18	50.19	25.79
	S.D.	13.10	14.91	1440.71	0.38	0.94	9.00	12.60	12.39	16.00	1498.22	0.39	0.98	9.90	12.88
	1st quart	64	48	3198.00	1.03	3	43.51	15.44	63	46	3175.50	1.05	4	42.66	15.65
	2nd quart	71	56	4047.00	1.27	4	48.64	27.22	70	55	3844.00	1.29	4	48.54	27.99
	3rd quart	80	67	5022.00	1.55	5	55.99	36.18	79	66	4816.75	1.57	5	56.38	36.83
Class 7	Max	96	140	9585	1.87	7	94.42	44.92	93	116	9579	1.63	8	86.53	44.82
	Min	22	28	1080	0.27	1	30.65	0.04	20	34	1050	0.33	2	35.05	0.06
	Avg	46.65	69.75	3305.74	0.70	4.19	54.49	17.63	49.12	67.55	3368.51	0.76	4.31	55.02	18.93
	S.D.	12.56	17.00	1350.61	0.23	0.88	11.30	12.28	13.59	16.77	1404.45	0.26	0.89	10.91	12.66
	1st quart	36	57	2240.00	0.53	4	46.09	7.31	38	55	2296.5	0.56	4	45.77	8.16
	2nd quart	45	68	3081.00	0.65	4	52.29	15.32	48	63	3193.5	0.71	4	53.14	16.53
	3rd quart	55	81	4080.00	0.82	5	61.07	26.58	60	77	4138.5	0.90	5	62.71	28.99
Class 8	Max	86	130	8686	1.65	8	92.50	44.99	69	121	6649	1.36	7	85.01	44.99
	Min	17	30	805	0.23	1	29.33	0.16	22	33	936	0.26	1	30.15	2.07
	Avg	40.99	64.73	2685.11	0.66	3.97	52.71	30.19	41.00	64.56	2687.23	0.67	3.96	52.31	30.16
	S.D.	9.79	15.67	1021.25	0.20	0.99	9.09	10.20	9.70	17.01	1061.55	0.20	0.97	9.42	10.21
	1st quart	34	54	1927.00	0.52	3	46.46	23.42	34	52	1804	0.52	3	46.04	23.19
	2nd quart	39	63	2520.00	0.63	4	51.38	31.53	40	62	2464	0.64	4	51.42	31.52
	3rd quart	47	74	3235.50	0.77	5	57.34	38.62	47	74	3354	0.79	5	57.33	38.98
Class 9	Max	127	129	12348	2.00	7	87.99	44.89	127	129	12032	1.77	9	81.77	44.98
	Min	28	30	1344	0.40	1	29.00	0.01	38	30	1500	0.51	2	32.00	0.05
	Avg	66.45	69.40	4725.01	0.99	3.95	48.48	20.39	66.17	70.09	4779.43	0.98	3.80	47.92	20.74
	S.D.	14.59	16.38	1865.02	0.23	0.91	9.83	12.40	15.47	17.75	2011.65	0.24	0.95	10.34	13.39
	1st quart	55	58	3328.00	0.82	3	41.17	9.95	53	56	3218	0.78	3	39.78	9.31
	2nd quart	66	68	4446.50	0.95	4	46.75	19.57	65	68	4548	0.93	4	46.01	19.19
	3rd quart	77	79	5778.75	1.13	5	53.95	30.44	78	81.75	6041	1.14	4	54.65	32.74

We tested this mixed recognizer separately on the test sets of both the Bangla and English databases consisting of 4,000 and 10,000 samples, respectively. The recognition accuracies are, respectively, 98.05 percent and 98.64 percent. Detailed results are provided in Table 7.

TABLE 9
Various Details of the Samples in the Bangla Numeral Database

		Training Set							Test Set						
		Height	Width	Vol	Aspect ratio	Penwidth	Contrast	Major Axis	Height	Width	Vol	Aspect ratio	Penwidth	Contrast	Major Axis
Class 0	Max	109	95	9310	2.27	16	152.61	44.96	136	96	11016.00	1.97	10.00	130.00	44.31
	Min	20	15	300	0.63	2	41.69	0.00	22	17	425.00	0.63	3.00	41.94	0.11
	Avg	45.15	37.85	1804.20	1.22	4.63	73.74	13.73	48.60	41.28	2167.52	1.20	5.38	76.87	12.47
	S.D.	12.17	10.61	1000.66	0.25	0.95	13.70	10.26	15.91	13.09	1473.58	0.24	1.11	12.92	9.86
	1st quart	36	30	1120	1.04	4	63.80	5.66	38	32	1228.50	1.03	5.00	68.75	4.77
	2nd quart	44	36	1575	1.19	5	72.61	11.65	45	38	1721.00	1.17	5.00	76.28	10.21
	3rd quart	51	44	2193	1.36	5	82.10	19.27	56	49	2673.00	1.33	6.00	84.45	17.62
Class 1	Max	161	125	19250	3.02	24	126.33	44.97	87	98	8526.00	3.09	15.00	97.05	44.67
	Min	22	16	450	0.54	1	32.28	0.02	30	21	828.00	0.63	2.00	21.78	0.11
	Avg	53.78	42.52	2413.43	1.31	5.05	64.57	17.09	51.78	37.79	1978.40	1.42	4.31	60.16	13.33
	S.D.	16.20	12.85	1478.56	0.34	1.33	12.12	11.22	10.37	8.65	711.03	0.38	1.15	10.00	10.08
	1st quart	43	34	1520	1.08	4	56.62	8.14	44	32	1518.00	1.15	4.00	53.27	5.71
	2nd quart	51	41	2014	1.27	5	64.46	15.38	50	37	1887.00	1.36	4.00	58.93	11.75
	3rd quart	61	48	2808	1.49	6	71.84	24.23	58	42	2295.25	1.63	5.00	65.58	18.91
Class 2	Max	158	203	22939	2.88	18	106.86	44.99	140	179	22820.00	2.10	11.00	83.77	44.88
	Min	24	20	552	0.33	1	25.71	0.01	29	23	896.00	0.41	3.00	29.14	0.17
	Avg	54.21	51.39	2904.07	1.11	4.63	53.11	23.74	56.75	55.08	3304.58	1.09	5.30	55.63	24.60
	S.D.	15.38	16.01	1688.72	0.32	1.28	10.57	12.47	16.59	19.67	2141.60	0.31	1.28	9.12	12.22
	1st quart	44	40	1890	0.88	4	45.88	14.02	46	41	1980.00	0.88	4.00	48.88	15.39
	2nd quart	52	49	2520	1.08	4	51.95	24.38	53	52	2754.00	1.07	5.00	55.68	24.89
	3rd quart	61	59	3392	1.29	5	59.26	34.09	65.25	64	3940.25	1.26	6.00	62.00	34.45
Class 3	Max	116	162	17334	2.00	13	102.70	44.94	132	130	17160.00	1.63	9.00	111.68	44.94
	Min	18	26	630	0.27	1	30.21	0.06	20	30	630.00	0.40	3.00	28.67	0.07
	Avg	46.30	58.47	2816.23	0.82	4.53	59.50	21.25	47.60	60.74	3112.19	0.80	4.21	57.74	22.76
	S.D.	13.04	16.16	1526.00	0.23	1.09	11.25	12.79	17.47	18.80	2119.17	0.22	0.88	12.73	13.16
	1st quart	37	47	1848.75	0.67	4	51.38	10.08	36	46	1689.75	0.64	4.00	47.87	11.22
	2nd quart	44	56	2451	0.78	4	58.49	20.99	44	56.5	2410.00	0.77	4.00	57.58	22.48
	3rd quart	53	66	3360	0.93	5	66.85	31.65	55	74	3970.50	0.93	5.00	65.47	34.54
Class 4	Max	131	109	13696	2.90	20	153.00	44.82	125	84	8148.00	2.12	10.00	123.20	44.65
	Min	27	17	561	0.61	1	36.55	0.01	33	21	819.00	0.84	3.00	48.77	0.00
	Avg	56.23	39.48	2337.12	1.46	4.62	74.51	10.87	58.49	41.65	2551.53	1.43	5.43	79.42	10.97
	S.D.	15.05	11.10	1339.52	0.29	1.15	14.81	8.22	14.70	10.72	1273.31	0.26	1.20	13.38	8.28
	1st quart	46	32	1530	1.27	4	64.36	4.28	48	34	1626.00	1.23	5.00	69.17	4.21
	2nd quart	53	37	1980	1.44	4	73.47	9.02	57	40	2258.50	1.40	5.00	78.33	9.57
	3rd quart	63	44	2680	1.63	5	83.28	15.56	66	48	3061.50	1.59	6.00	87.77	16.10
Class 5	Max	137	146	16029	2.03	11	115.11	44.99	114	98	9135.00	1.75	12.00	118.32	44.72
	Min	26	23	754	0.40	1	35.77	0.01	24	24	840.00	0.55	2.00	41.08	0.18
	Avg	52.67	51.75	2841.89	1.04	4.52	65.82	20.16	53.57	52.06	2935.10	1.04	4.37	64.36	21.26
	S.D.	13.35	13.05	1471.72	0.20	1.08	11.81	12.81	15.20	13.05	1550.55	0.20	1.05	12.89	12.48
	1st quart	44	43	1927	0.89	4	57.51	9.35	43	42	1849.50	0.90	4.00	55.05	10.41
	2nd quart	51	49	2499	1.02	4	64.64	19.06	51	49	2475.00	1.02	4.00	62.31	21.08
	3rd quart	59	58	3276	1.16	5	73.50	30.59	60	59	3485.25	1.17	5.00	72.49	31.53
Class 6	Max	137	183	18666	1.91	21	94.70	44.96	116	135	14036.00	1.31	8.00	94.85	44.48
	Min	24	24	768	0.34	2	30.32	0.05	20	27	700.00	0.47	2.00	31.97	0.20
	Avg	47.74	56.32	2802.64	0.87	4.57	56.32	21.42	50.96	61.70	3380.06	0.84	4.40	53.48	19.79
	S.D.	12.15	15.15	1495.31	0.20	1.16	9.92	12.80	16.24	18.80	2198.18	0.17	1.00	10.51	12.37
	1st quart	39	46	1862	0.74	4	48.75	10.39	40	49	2049.00	0.72	4.00	45.33	9.22
	2nd quart	46	54	2475	0.85	4	55.34	20.91	47	57	2700.00	0.82	4.00	52.21	18.62
	3rd quart	54	64	3300	0.98	5	62.92	32.11	58	72	3901.25	0.95	5.00	60.15	29.75
Class 7	Max	184	97	12696	3.71	10	98.68	44.96	184	106	19504.00	4.14	15.00	103.22	44.63
	Min	30	18	720	0.71	2	29.70	0.01	30	19	735.00	0.81	2.00	29.77	0.07
	Avg	68.02	41.66	2955.60	1.67	4.72	58.15	20.28	63.44	37.78	2487.88	1.73	4.63	60.68	18.33
	S.D.	19.16	10.90	1546.76	0.40	0.99	10.47	11.39	16.71	9.73	1405.02	0.42	1.27	10.83	11.12
	1st quart	55	34	1924	1.40	4	50.54	11.55	52.75	31	1710.75	1.44	4.00	52.34	9.16
	2nd quart	65	40	2565	1.62	5	57.41	19.62	61	37	2212.50	1.65	4.00	59.61	17.95
	3rd quart	76	47	3480	1.89	5	64.62	28.67	71	43	2960.00	1.93	5.00	67.19	26.09
Class 8	Max	136	185	19448	1.97	16	88.64	44.99	136	150	19448.00	1.66	11.00	83.28	44.82
	Min	22	25	702	0.33	1	25.75	0.04	25	27	945.00	0.38	2.00	31.29	0.09
	Avg	54.31	61.58	3471.86	0.93	4.38	50.87	17.00	55.00	62.53	3677.77	0.92	5.19	55.24	16.45
	S.D.	12.41	19.04	1798.79	0.24	1.04	9.28	12.41	15.88	22.04	2474.13	0.23	1.16	8.96	12.43
	1st quart	46	49	2310	0.76	4	44.09	6.54	44	48	2242.00	0.75	4.00	49.07	5.97
	2nd quart	53	59	3078	0.90	4	49.83	14.43	52	57	2864.50	0.89	5.00	54.70	13.47
	3rd quart	61	70	4095	1.07	5	56.62	25.77	63.25	71	4237.50	1.08	6.00	60.84	25.46
Class 9	Max	152	205	19608	2.50	18	100.07	44.99	136	156	20280.00	2.24	9.00	91.80	44.95
	Min	30	27	900	0.30	1	29.30	0.00	26	27	754.00	0.58	3.00	31.01	0.12
	Avg	63.24	55.12	3614.56	1.19	4.72	56.03	21.88	72.28	60.72	4657.28	1.23	4.38	51.87	22.77
	S.D.	14.87	15.94	1846.01	0.27	1.22	9.96	12.99	20.92	19.36	2726.65	0.30	0.95	10.50	13.37
	1st quart	53	45	2496.75	1.00	4	49.05	10.58	57.75	47	2852.50	1.00	4.00	44.13	11.25
	2nd quart	61	53	3237	1.17	5	55.28	21.11	69.5	57	3858.00	1.19	4.00	50.79	22.01
	3rd quart	71	61	4157	1.35	5	62.16	32.86	83	71	5810.00	1.42	5.00	58.93	35.19

4.5.2 Twenty-Class Bangla-English Mixed Numeral Recognizer

Similarly to the 20-class Devanagari-English mixed recognizer, we simulated another 20-class Bangla-English mixed numeral recognizer trained in a similar fashion. This recognizer provided 65.20 percent and 70.85 percent recognition accuracies on the test sets of Devanagari and English numerals, respectively. If we ignore misclassification within the shape-similar classes (i.e., misclassifications of English numerals 0, 2, 8, and 9, respectively, as classes representing Bangla numerals 0, 2, 4, and 7), the above figures are improved to 97.93 percent and 98.52 percent, respectively.

5 Conclusion

In this paper, we have presented a pioneering effort for the development of handwritten numeral database of Indian scripts. Also, a multistage method for high accuracy recognition of these handwritten numerals has been described. Moreover, the proposed method has been implemented for recognition of handwritten numerals in mixed script situations. Similar mixed script situations in handwritten Indian documents are a common phenomenon and recognition studies for such situations have enough practical importance in automatic reading of postal code or tabular form documents.

Appendix A

Details of the samples in the Devangari numeral database are shown in Table 8.

Appendix B

Details of the samples in the Bangla numeral database are shown in Table 9.

Acknowledgments

The authors would like to thank everybody who extended their help in developing the databases described in the present paper. Also, the authors are grateful to the reviewers for many valuable comments and suggestions.

References

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.

[2] J.J. Hull, "A Database for Handwritten Text Recognition Research," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, pp. 550-554, 1994.

[3] T. Saito, H. Yamada, and K. Yamamoto, "On the Database ELT9 of Handprinted Characters in JIS Chinese Characters and Its Analysis (in Japanese)," *Trans. IECEJ*, vol. J.68-D, no. 4, pp. 757-764, 1985.

[4] Y. Al-Ohali, M. Cheriet, and C. Suen, "Databases for Recognition of Handwritten Arabic Cheques," *Pattern Recognition*, vol. 36, pp. 111-121, 2003.

[5] T. Noumi, T. Matsui, I. Yamashita, T. Wakahara, and T. Tsutsumida, "Tegaki Suji Database 'IPTP CD-ROM1' no ichi bunseki (in Japanese)," *Autumn Meeting of IEICE*, vol. D-309, Sept. 1994.

[6] C.Y. Suen, M. Berthod, and S. Mori, "Automatic Recognition of Handprinted Characters—The State of the Art," *Proc. IEEE*, vol. 68, no. 4, pp. 469-487, 1980.

[7] S.N. Srihari, E. Cohen, J.J. Hull, and L. Kuan, "A System to Locate and Recognize ZIP Codes in Handwritten Addresses," *Int'l J. Research and Eng.-Postal Applications*, vol. 1, pp. 37-45, 1989.

[8] J. Tsukumo and H. Tanaka, "Classification of Handprinted Chinese Characters Using Nonlinear Normalization Methods," *Proc. Ninth Int'l Conf. Pattern Recognition*, pp. 168-171, 1988.

[9] A. Amin and H.B. Al-Sadoun, "Hand Printed Arabic Character Recognition System," *Proc. 12th Int'l Conf. Pattern Recognition*, pp. 536-539, 1994.

[10] S.W. Lee and J.S. Park, "Nonlinear Shape Normalization Methods for the Recognition of Large-Set Handwritten Characters," *Pattern Recognition*, vol. 27, pp. 895-902, 1994.

[11] H. Yamada, K. Yamamoto, and T. Saito, "A Non-Linear Normalization Method for Handprinted Kanji Character Recognition—Line Density Equalization," *Pattern Recognition*, vol. 23, pp. 1023-1029, 1990.

[12] O.D. Trier, A.K. Jain, and T. Taxt, "Feature Extraction Methods for Character Recognition—A Survey," *Pattern Recognition*, vol. 29, pp. 641-662, 1996.

[13] I.-S. Oh and C.Y. Suen, "Distance Features for Neural Network-Based Recognition of Handwritten Characters," *Int'l J. Document Analysis and Recognition*, vol. 1, no. 2, pp. 73-88, 1998.

[14] R. Plamondon and S.N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 63-84, 2000.

[15] N. Arica and F. Yarman-Vural, "An Overview of Character Recognition Focused on Off-Line Handwriting," *IEEE Trans. Systems, Man, and Cybernetics—Part C: Applications and Rev.*, vol. 31, pp. 216-233, 2001.

[16] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake, "Modified Quadratic Discriminant Functions and the Application to Chinese Character Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 1, pp. 149-153, 1987.

[17] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning Internal Representations by Error Propagation," Inst. for Cognitive Science Report 8506, Univ. of California, 1985.

[18] C.M. Bishop, *Neural Networks for Pattern Recognition.* Clarendon, 1995.

[19] J.S.U. Kreßel, *Pattern Classification Techniques Based on Function Approximation*, pp. 49-78. World Scientific Publishing, 1997.

[20] T. Kohonen, "Improved Versions of Learning Vector Quantization," *Proc. Int'l Joint Conf. Neural Networks*, vol. 1, pp. 545-550, 1990.

[21] C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Knowledge Discovery and Data Mining*, vol. 2, no. 2, p. 143, 1998.

[22] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Müller, P.S.E. Säckinger, and V. Vapnik, "Comparison of Learning Algorithms for Handwritten Digit Recognition," *Proc. Int'l Conf. Artificial Neural Networks*, p. 5360, 1995.

[23] C.-L. Liu, H. Sako, and H. Fujisawa, "Performance Evaluation of Pattern Classifiers for Handwritten Character Recognition," *Int'l J. Document Analysis and Recognition*, vol. 4, no. 3, pp. 191-204, 2002.

[24] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, "Handwritten Digit Recognition: Benchmarking of State-of-the-Art Techniques," *Pattern Recognition*, vol. 36, pp. 2271-2285, 2003.

[25] F. Kimura, Y. Miyake, and M. Sridhar, "Handwritten ZIP Code Recognition Using Lexicon Free Word Recognition Algorithm," *Proc. Third Int'l Conf. Document Analysis and Recognition*, vol. 2, pp. 906-910, 1995.

[26] H. Liu and X. Ding, "Handwritten Character Recognition Using Gradient Feature and Quadratic Classifier with Multiple Discrimination Schemes," *Proc. Eighth Int'l Conf. Document Analysis and Recognition*, pp. 19-25, 2005.

[27] 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, pp. 2051-2059, 2002.

[28] A. Shustorovich, "A Subspace Projection Approach to Feature Extraction: The Two-Dimensional Gabor Transform for Character Recognition," *Neural Networks*, vol. 7, no. 8, pp. 1295-1301, 1994.

[29] L. Heutte, T. Paquet, J.V. Moreau, Y. Lecourtier, and C. Olivier, "A Structural/Statistical Feature Based Vector for Handwritten Character Recognition," *Pattern Recognition Letters,* vol. 19, no. 7, pp. 629-641, 1998.

[30] J. Favata, G. Srikantan, and S. Srihari, "Handprinted Character/Digit Recognition Using a Multiple Feature/Resolution Philosophy," *Proc. Fourth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 57-66, 1994.

[31] J. Park, V. Govindaraju, and S.N. Srihari, "OCR in a Hierarchical Feature Space," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 4, pp. 400-407, Apr. 2000.

[32] J. Cao, M. Ahmadi, and M. Shridhar, "Recognition of Handwritten Numerals with Multiple Features and Multi-Stage Classifiers," *Pattern Recognition,* vol. 28, no. 2, pp. 153-160, 1995.

[33] C.Y. Suen and J. Tan, "Analysis of Errors of Handwritten Digits Made by a Multitude of Classifiers," *Pattern Recognition Letters,* vol. 26, no. 1, pp. 369-379, 2005.

[34] N. Giusti, F. Masuli, and A. Sperduti, "Theoretical and Experimental Analysis of a Two-Stage System for Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 7, pp. 893-904, July 2002.

[35] C.M. Nunes, A.de.S. Britto Jr., C.A.A. Kaestner, and R. Sabourin, "An Optimized Hill Climbing Algorithm for Feature Subset Selection Evaluation on Handwritten Character Recognition," *Proc. Ninth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 365-370, 2004.

[36] U. Bhattacharya, T.K. Das, and B.B. Chaudhuri, "A Cascaded Scheme for Recognition of Handprinted Numerals," *Proc. Third Indian Conf. Computer Vision, Graphics and Image Processing,* pp. 137-142, 2002.

[37] U. Bhattacharya and B.B. Chaudhuri, "Fusion of Combination Rules of an Ensemble of MLP Classifiers for Improved Recognition Accuracy of Handprinted Bangla Numerals," *Proc. Eighth Int'l Conf. Document Analysis and Recognition,* vol. 1, pp. 322-326, 2005.

[38] F.F. Soulie, E. Vinnet, and B. Lamy, "Multi-Modular Neural Network Architectures: Application in Optical Character Recognition and Human Face Recognition," *Int'l J. Pattern Recognition and Artificial Intelligence,* vol. 5, pp. 721-725, 1993.

[39] I.K. Sethi and B. Chatterjee, "Machine Recognition of Constrained Handprinted Devanagari," *Pattern Recognition,* vol. 9, pp. 69-75, 1977.

[40] K.R. Ramakrishnan, S.H. Srinivasan, and S. Bhagavathy, "The Independent Components of Characters Are "Strokes"," *Proc. Fifth Int'l Conf. Document Analysis and Recognition,* pp. 414-417, Sept. 1999.

[41] R. Bajaj, L. Dey, and S. Chaudhuri, "Devnagari Numeral Recognition by Combining Decision of Multiple Connectionist Classifiers," *Sadhana,* vol. 27, no. part 1, pp. 59-72, 2002.

[42] U. Bhattacharya, T.K. Das, A. Datta, S.K. Parui, and B.B. Chaudhuri, "A Hybrid Scheme for Handprinted Numeral Recognition Based on a Self-Organizing Network and MLP Classifiers," *Int'l J. Pattern Recognition and Artificial Intelligence,* vol. 16, no. 7, pp. 845-864, 2002.

[43] T.K. Bhowmick, U. Bhattacharya, and S.K. Parui, "Recognition of Bangla Handwritten Characters Using an MLP Classifier Based on Stroke Features," *Proc. 11th Int'l Conf. Neural Information Processing,* pp. 814-819, 2004.

[44] M. Hanmandlu and O.V.R. Murthy, "Fuzzy Model Based Recognition of Handwritten Numerals," *Pattern Recognition,* vol. 40, no. 6, pp. 1840-1854, 2007.

[45] P. Chinnuswamy and S. Krishnamoorthy, "Recognition of Handprinted Tamil Characters," *Pattern Recognition,* vol. 12, no. 3, pp. 141-152, 1980.

[46] M.B. Sukhaswami, P. Seetharamulu, and A.K. Pujari, "Recognition of Telugu Characters Using Neural Networks," *Int'l J. Neural Systems,* vol. 6, no. 3, pp. 317-357, 1995.

[47] R.M.K. Sinha, "Rule Based Contextual Post-Processing for Devanagari Text Recognition," *Pattern Recognition,* vol. 20, pp. 475-485, 1987.

[48] V. Bansal and R.M.K. Sinha, "Integrating Knowledge Sources in Devnagari Text Recognition System," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 30, no. 4, pp. 500-505, 2000.

[49] B.B. Chaudhuri and U. Pal, "Skew Angle Detection of Digitized Indian Script Documents," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 2, pp. 182-186, Feb. 1997.

[50] R.M.K. Sinha and H.N. Mahabala, "Machine Recognition of Devanagari Script," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 9, no. 8, pp. 435-441, 1979.

[51] B.B. Chaudhuri and U. Pal, "A Complete Printed Bangla OCR System," *Pattern Recognition,* vol. 31, no. 5, pp. 531-549, 1998.

[52] U. Bhattacharya and B.B. Chaudhuri, "Databases for Research on Recognition of Handwritten Characters of Indian Scripts," *Proc. Eighth Int'l Conf. Document Analysis and Recognition (ICDAR '05),* vol. 2, pp. 789-793, 2005.

[53] F. Kimura, *OCR Technologies for Machine Printed and Hand Printed Japanese Text.* Springer, pp. 49-71, 2007.

[54] N. Otsu, "A Threshold Selection Method from Grey-Level Histograms," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 9, pp. 377-393, 1979.

[55] I. Daubechies, "The Wavelet Transform, Time-Frequency Localization and Signal Analysis," *IEEE Trans. Information Theory,* vol. 36, no. 5, pp. 961-1005, 1990.

[56] S.G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, no. 7, pp. 674-693, July 1989.

[57] R.P. Lippmann, "An Introduction to Computing with Neural Nets," *IEEE ASSP Magazine,* vol. 4, pp. 4-22, 1987.

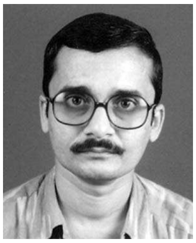
[58] N. Wanas, G. Auda, M.S. Kamel, and F. Karray, "On the Optimal Number of Hidden Nodes in a Neural Network," *Proc. IEEE Canadian Conf. Electrical and Computer Eng.,* vol. 2, pp. 918-921, 1998.

[59] S. Haykin, *Neural Networks: A Comprehensive Foundation,* second ed. Prentice Hall, 1998.

[60] R.J. Ramteke, P.D. Borkar, and S.C. Mehrotra, "Recognition of Isolated Marathi Handwritten Numerals: An Invariant Moments Approach," *Proc. Int'l Conf. Cognition and Recognition,* pp. 482-489, 2005.

[61] U. Bhattacharya, S.K. Parui, B. Shaw, and K. Bhattacharya, "Neural Combination of ANN and HMM for Handwritten Devnagari Numeral Recognition," *Proc. 10th Int'l Workshop Frontiers in Handwriting Recognition,* pp. 613-618, 2006.

[62] K. Roy, C. Chaudhuri, U. Pal, and M. Kundu, "A Study on the Effect of Varying Training Set Sizes on Recognition Performance with Handwritten Bangla Numerals," *Proc. IEEE-India Ann. Conf.,* pp. 570-574, 2005.



Ujjwal Bhattacharya received the MSc and MPhil degrees in pure mathematics from Calcutta University, Kolkata, India, in 1989 and 1991, respectively. He joined the Indian Statistical Institute (ISI), Kolkata, in 1991 as a junior research fellow, where he is now a member of the faculty of the Computer Vision and Pattern Recognition Unit. He received the Young Scientist Award from the Indian Science Congress Association in 1995, the Certificate of Special Mention from the Computer Society of India, and the Amiya K. Pujari IT Award from the Orissa Information Technology Society, India, both in 2006. He visited the Inter-University Center for Astronomy and Astrophysics, Pune, India, in 1996, the Institute of Biomathematics and Biometry, GSF, Munich, in 1998-1999, the Laboratory of DNA Information Analysis of Human Genome Center, Institute of Medical Science, University of Tokyo, the Department of Computational Science and Engineering, Nagoya University, Japan, in 2004, the Computer Science Department, Bristol University, the Department of Computer Science, University of Essex, and the Department of Electronics, University of Kent, Canterbury, United Kingdom, in 2006. His research interests include image processing, pattern recognition, neural networks, machine learning, and especially the applications to character recognition and document processing. He is a member of the IEEE.

B.B. Chaudhuri received the BSc (Hons), BTech, and MTech degrees from Calcutta University, Kolkata, India, in 1969, 1972, and 1974, respectively, and the PhD degree from the Indian Institute of Technology, Kanpur, in 1980. He worked as a postdoctoral fellow at Queen's University, United Kingdom, from 1981 to 1982. He joined the Indian Statistical Institute in 1978, where he served as the project coordinator and the head of the National Nodal Center for Knowledge Based Computing funded by DoE and UNDP. He is currently the head of the Computer Vision and Pattern Recognition Unit of the institute. His research interests include pattern recognition, image processing, computer vision, natural language processing (NLP), information retrieval, speech processing, digital document processing, and OCR. He pioneered the first Indian language Bharati Braille system for the blind, a successful Bangla speech synthesis system, as well as the first workable OCR for Bangla, Devnagari, Assamese, and Oriya scripts. In the NLP area, a robust Indian language spellchecker, morphological processor, multiword expression detector, and statistical analyzer were pioneered by him. Some of his technologies have been transferred to industry for commercialization. He has published about 325 research papers in reputed international journals, conference proceedings, and edited books. He has authored five books entitled *Two Tone Image Processing and Recognition* (Wiley Eastern, 1993), *Object Oriented Programming: Fundamentals and Applications* (Prentice Hall, 1998), *Computer and Software Technology Dictionary* (Ananda Publishers, 2002), *Matlab Programming* (Prentice Hall, India, 2007), and *Digital Document Processing* (B.B. Chaudhuri, ed., Springer-Verlag, 2007). He was awarded a Leverhulme fellowship in 1981-1982, the Sir J.C. Bose Memorial Award in 1986, the M.N. Saha Memorial Award (twice) in 1989 and 1991, the Homi Bhabha Fellowship award in 1992, the Dr. Vikram Sarabhai Research Award in 1995, the C. Achuta Menon Prize in 1996, the Homi Bhabha Award: Applied Sciences in 2003, and the Ram Lal Wadhwa Gold Medal in 2005. He received the prestigious Jawaharlal Nehru Fellowship from 2004 to 2006. He was a visiting scientist at Queen's University from 1981 to 1982. He conducted collaborative research at GSF, Munich, from 1986 to 2002. He worked as a guest faculty member at the Technical University of Hannover from 1986 to 1988. He is a member secretary (Indian Section) of the International Academy of Sciences, a fellow of the Indian National Science Academy, a fellow of the National Academy of Sciences, a fellow of the Institution of Electronics and Telecommunication Engineering, and a fellow of the Indian National Academy of Engineering. In 1998 and 2000, respectively, he was elected a fellow of the International Association of Pattern Recognition (IAPR) and a fellow of the IEEE. He is currently serving as an associate editor of *VIVEK*, the *International Journal of Pattern Recognition and Artificial Intelligence*, the *International Journal of Computer Vision*, and the *International Journal of Document Analysis and Recognition*. He has served as a guest editor of special issues of several journals.