Anirban Mukhopadhyay, Pawan Kumar Singh*, Ram Sarkar and Mita Nasipuri

Handwritten *Indic* Script Recognition Based on the Dempster–Shafer Theory of Evidence

https://doi.org/10.1515/jisys-2017-0431
Received July 29, 2017; previously published online February 6, 2018.

Abstract: In a multilingual country like India, script recognition is an important pre-processing footstep necessary for feeding any document to an optical character recognition (OCR) engine, which is, in general, script specific. The present work evaluates the performance of an ensemble of two MLP (multi-layer perceptron) classifiers, each trained on different feature sets. Here, two complementary sets of features, *namely*, gray-level co-occurrence matrix (GLCM) and Gabor wavelets transform coefficients are extracted from each of the handwritten text-line and word images written in 12 official scripts used in Indian subcontinent, which are then fed into an individual classifier. In order to improve the overall recognition rate, a powerful combination approach based on the Dempster–Shafer (DS) theory is finally employed to fuse the decisions of two MLP classifiers. The performance of the combined decision is compared with those of the individual classifiers, and it is noted that a significant improvement in recognition accuracy (about 4% for text-line data and 6% for word level data) has been achieved by the proposed methodology.

Keywords: Handwritten script recognition, Dempster–Shafer theory, *Indic* scripts, gray-level co-occurrence matrix, Gabor wavelet transform.

1 Introduction

Script identification is a method of identifying the scripts, written or handprinted in any multilingual, multi-script environment. Script identification is an essential requirement for any optical character recognition (OCR) engine, which is used to recognize the characters written in a particular script of the underlying document. In general, recognition of different scripts by a single OCR module is next to impossible. As a substitute, a pool of OCR systems corresponding to different scripts [33] could be an answer to solve the said problem. This implies that there is a need to develop a script identification system to identify the script type of the document, so that a specific OCR engine can be selected to convert the text image into a computer-editable format. The script identification from the handwritten samples has applications in automatic archiving and indexing of multilingual documents, searching necessary information from digitized archives of the multilingual document images, and so on.

Handwritten documents allow different representations for character sets of different scripts, which are somewhat restricted in printed documents. Individual differences, cultural differences, and even differences in the way people write at different times, increase the inventory of possible word shapes seen in handwritten documents. Also, problems like ruling lines, word fragmentation due to low contrast, noise, skewness, etc., are common in handwritten documents. The visual appearance of the script varies sufficiently from word to word, and not so much from character to character. Multi-script scenarios are common in a country like India, where 12 different scripts are written using 23 constitutionally recognized languages. This also requires identification of the scripts at both the line level and word level. As the *intra-class* variations among the 12

***Corresponding author: Pawan Kumar Singh,** Department of Computer Science and Engineering, Jadavpur University, 188 Raja S.C. Mullick Road, Kolkata-700032, West Bengal, India, e-mail: pawansingh.ju@gmail.com
Anirban Mukhopadhyay, Ram Sarkar and Mita Nasipuri: Department of Computer Science and Engineering, Jadavpur University, 188 Raja S.C. Mullick Road, Kolkata-700032, West Bengal, India

Indian scripts are mostly found at these two levels, therefore, the identification of the scripts at line level and word level is a challenging task.

Script recognition articles for handwritten documents are relatively limited in comparison to its printed counterpart. Spitz [37] proposed a method for distinguishing between *Asian* and *European* languages by analyzing the connected components. Tan et al. [39] proposed a method based on texture analysis for automatic script identification from document images using multiple channel (Gabor) filters and gray-level co-occurrence matrices for seven languages: *Chinese*, *English*, *Greek*, *Korean*, *Malayalam*, *Persian*, and *Russian*. Hochberg et al. [16, 17] described an algorithm for script and language identification from handwritten document images using statistical features based on connected component analysis. Wood et al. [42] described the projection profile method to determine characters of *Roman*, *Russian*, *Arabic*, *Korean*, and *Chinese* scripts. Chaudhuri et al. [6] discussed an OCR system to read two Indian scripts viz., *Bangla* and *Devanagari* (written in *Hindi* language). Pal et al. [27] proposed an algorithm for word-wise script identification from documents containing *English*, *Devanagari*, and *Telugu* text, based on conventional and water reservoir features. Chaudhury et al. [5] proposed a method for identification of Indian languages by combining Gabor filter-based techniques and direction distance histogram classifier for *Hindi*, *English*, *Malayalam*, *Bengali*, *Telugu*, and Urdu languages. Recently, Singh et al. [33] provided a comprehensive survey considering various feature extraction and classification techniques associated with the offline script identification of printed as well as handwritten *Indic* scripts.

A variety of individual classifiers has been employed since the last few decades for *Indic* script recognition, including the *k*-nearest neighbors (*k*-NN) [13, 25], linear discriminant analysis (LDA) [29], neural networks (NNs) [9, 29], support vector machine (SVM) [4, 9], tree-based classifier [26, 27], simple logistic [32], and multi-layer perceptron (MLP) [35, 36] among others. Although much evolution has already been carried out, but with a single classifier, it is still difficult to achieve satisfactory performance in almost all practical applications.

Improvement in classification accuracy is considered to be a significant task in solving any pattern recognition problem. In order to achieve this, researchers, depending upon the purpose of interest, have explored several methods since the past few decades. As such, a classification algorithm used with a particular set of features may not be appropriate with another set of features. In addition, classification algorithms are distinct in terms of the hypotheses used and, hence, achieve different degrees of accuracy for different applications. However, a specific feature set used with a specific classifier might achieve better results than those obtained using another feature set and/or classification scheme. Based on these facts, it is difficult to conclude that a particular set of feature vector and/or classification scheme will achieve the best possible classification results [19]. As different classifiers may offer complementary information about the patterns to be classified, combining classifiers, in an efficient way, can achieve better results than any individual classifier (even the best one).

The previous idea has motivated the relatively latest attention in combining classifiers. The idea is not to rely on a single decision-making scheme. Instead, all the designs or their subsets are used for decision making by combining their individual opinions to derive a consensus decision. Various classifier combination schemes have been devised over the years, and it has been experimentally demonstrated that some of them consistently outperform a single best classifier.

From the literature survey, it has been observed that extensive studies have been done for the combination of multiple classifiers, which operates on the outputs of individual classifiers. The main methodologies for combining multiple classifiers include majority voting [23, 38], subset-combining and re-ranking approach [14], the statistical model [15], Bayesian belief integration [43], combination based on Dempster–Shafer (DS) theory of evidence [22, 43], and neural network combinator [20]. The approach followed here is to develop a function or a rule that combines the classifier scores in a predetermined manner. The DS rule is the generalization of the Bayesian theory of probability that introduces the degree of belief on a set of outcomes. Finally, the combination rule gives the confidence value to a class by uniting the multiple sources of information. The class-wise performance-based basic probability assignment (BPA), which outperforms the global performance-based BPA, has been implemented for the multi-classifier combination using the DS theory [45].

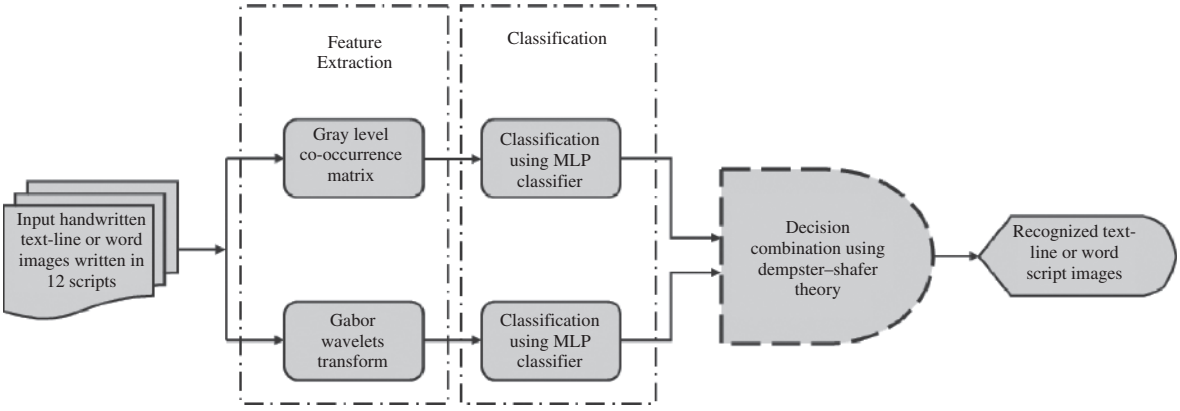

Figure 1: Schematic Diagram of the Present Work.

Earlier, the DS theory-based combination was applied in different fields like handwritten digit recognition [1], skin detection [31], and 3D palm print recognition [24] among other pattern recognition domains.

The main contribution of the present work is the development of four different procedures using the DS theory of evidence in order to improve the overall recognition accuracy for identifying the script in which a document is written. It is a multi-class classification problem and in the present case, 12 officially used *Indic* scripts are considered, which are *Devanagari*, *Bangla*, *Oriya*, *Gujarati*, *Gurumukhi*, *Tamil*, *Telugu*, *Kannada*, *Malayalam*, *Manipuri*, *Urdu*, and *Roman*. Two different feature vectors based on texture analysis have been estimated from each of the handwritten text-line and word images. The identification of the scripts from the text-line and word images is done with these feature values by feeding the same into different MLP classifiers. The decision of the individual classifiers is then combined using the DS theory of evidence. This kind of work is implemented for the first time assuming the number of *Indic* scripts undertaken. The block diagram of the present work is shown in Figure 1.

2 Feature Extraction

In this paper, two popular methodologies of feature extraction have been used for the classifier combination, *namely*, the gray-level co-occurrence matrix (GLCM) and Gabor wavelets transform. These features have already been applied with satisfactory results for script identification problem [18, 34].

2.1 Gray Level Co-Occurrence Matrix (GLCM)

The GLCM estimates the properties of an image related to the second-order statistics, which considers the relationship among pixels or groups of pixels. Haralick et al. [12] suggested the use of GLCM, which has become one of the most well-known and widely used texture features. This method is based on the joint probability distributions of pairs of pixels. The GLCM shows how often each gray level occurs at a pixel located at a fixed geometric position relative to other pixels, as a function of the gray level [10]. Mathematically, for a given image I of size $M \times N$, and for a displacement vector $d(d_x, d_y)$, the GLCM is represented as a square matrix P of size $L \times L$ where, L is the number of gray level range (0, 1,.....L^{-1}) in the image.

$$P(i, j) = \sum_{i=1}^{M} \sum_{j=1}^{N} 1, \text{ if } I(x, y) = i \text{ and } I(x + d_x, y + d_y) = j \\ 0, \text{ otherwise} \tag{1}$$

Here, i and j are the intensity values of I, x and y are the spatial positions in I, and the offset (d_x, d_y) depends on the distance d at which the matrix is computed. Here, $P(i, j)$ is a count of the number of times $I(x, y) = i$ and

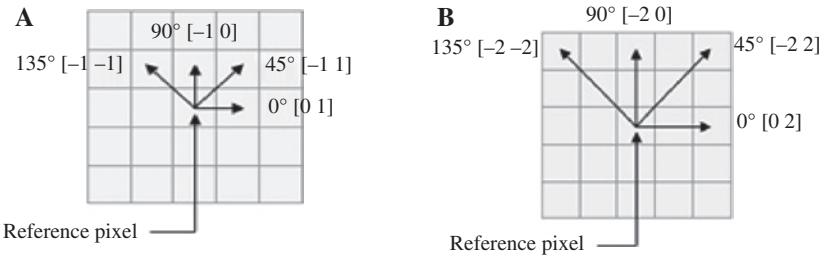

Figure 2: Directions of Co-Occurrence Matrices for Extracting Texture Features Where (A) $d=1$ and (B) $d=2$.

$I(x+d_x, y+d_y)=j$ occurs. Figure 2 illustrates the process to generate eight symmetrical co-occurrence matrices considering a 2×2 image represented with two-tone values 0 and 1. For this purpose, we have considered two neighboring pixels (at $d=1$ and $d=2$) along four possible directions.

Each element of the co-occurrence matrix is the number of times that the two pixels with gray tones i and j are neighboring at distance d and direction θ. Suppose, for 0° co-occurrence matrix where $d=1$, there are 12 occurrences of the pixel intensity value 0 and pixel intensity value 1 adjacent to each other in the input image. This implies that both the occurrence of pixel value 1 adjacent to pixel value 0 and the occurrence of pixel value 0 adjacent to pixel value 1 are 12 times. Hence, these matrices are symmetric in nature and for $\theta=0°$ and $\theta=180°$, the co-occurring pairs would be similar. This concept extends to 45°, 90°, and 135° as well. A set of 10 standard feature descriptors based on the GLCM has been extracted, which are described in Table 1. Here, i and j are the spatial coordinates of the function $P(i, j)$, and N_g is the gray level.

Because of the binary nature of the document images from which the features are estimated, the extraction of such features is unnecessary and indeed counterproductive. As there are only two gray levels, the matrices will be of size 2×2, i.e. it is possible to fully describe each matrix with only three unique parameters due to the diagonal symmetry property [3]. For each feature defined above, the values of $d=\{1, 2\}$ and 0°, 45°, 90° and 135° lead to a total of eight features. So, a total of 80 (i.e. 10∗8) dimensional feature set has been generated using the GLCM.

2.2 Gabor Wavelets Transform

Different wavelet transform functions filter out different range of frequencies (i.e. sub bands). In other words, the wavelet is a powerful tool, which decomposes the image into low-frequency and high-frequency subband images. Among various wavelet bases, the Gabor function provides the optimal resolution in both the time (spatial) and frequency domains, and the Gabor wavelet transform seems to be the optimal basis to extract local features. Besides, it has been found to show a distortion tolerance property for typical pattern-recognition tasks. The Gabor kernel is a complex sinusoid modulated by a Gaussian envelope. The Gabor wavelets have filter responses similar in shape to the respective fields in the primary visual cortex in mammalian brains [9]. The kernel or mother wavelet [21] in the spatial domain is given by:

$$\psi_j(\vec{x})=\frac{k_j^2}{\sigma^2}\exp\left(\frac{-k_j^2x^2}{2\sigma^2}\right)\left[\exp(j\vec{k}_j\vec{x})-\exp\left(\frac{-\sigma^2}{2}\right)\right] \tag{17}$$

where,

$$\vec{k}_j=\begin{pmatrix}k_v\sin\phi_\mu\\ k_v\cos\phi_\mu\end{pmatrix},\ k_v=2^{\frac{-v+1}{2}}\pi,\ \phi_\mu=\mu\frac{\pi}{8},\ \vec{x}=(x_1, x_2)\forall x_1, x_2\in R^2 \tag{18}$$

σ is the standard deviation of the Gaussian, $\vec{k}$ is the wave vector of the plane wave, ϕ_μ and k_v denote the orientations and frequency scales of Gabor wavelets, respectively, which are obtained from the mother wavelet. The Gabor wavelet representation of a block image is obtained by doing a convolution between the image and

Table 1: Standard GLCM Texture Descriptors [12].

Texture descriptors	Formula	
Energy	$\text{Energy} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} P^2(i,j)$	(2)
Entropy	$\text{Entropy} = -\sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} P(i,j)\log P(i,j)$	(3)
Inertia	$\text{Inertia} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} (i-j)^2 P(i,j)$	(4)
Autocorrelation	$\text{Autocorrelation} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} i.j.P(i,j)$	(5)
Covariance	$\text{Covariance} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} (i-M_x)(j-M_y)P(i,j)$	(6)
	Here,	
	$M_x = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} iP(i,j)$	(7)
	$M_y = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} jP(i,j)$	(8)
Contrast	$\text{Contrast} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} P(i,j)\lvert i-j\rvert^k,\ k \in \mathbb{Z}$	(9)
Local homogeneity	$\text{Local homogeneity} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} \frac{1}{1+(i-j)^2} P(i,j)$	(10)
Cluster shade	$\text{Cluster shade} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} (i-M_x+j-M_y)^3 P(i,j)$	(11)
Cluster prominence	$\text{Cluster prominence} = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} (i-M_x+j-M_y)^4 P(i,j)$	(12)
Information measure of correlation	$\text{Information measure of correlation} = \frac{-\sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} P(i,j)\log P(i,j) - H_{xy}}{\max(H_x, H_y)}$	(13)
	Here,	
	$H_{xy} = -\sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1} P(i,j)\log\left(\sum_{j=0}^{N_g-1} P(i,j).\sum_{i=0}^{N_g-1} P(i,j)\right)$	(14)
	$H_x = -\sum_{i=0}^{N_g-1}\left\{\sum_{j=0}^{N_g-1} P(i,j).\log\sum_{j=0}^{N_g-1} P(i,j)\right\}$	(15)
	$H_y = -\sum_{j=0}^{N_g-1}\left\{\sum_{i=0}^{N_g-1} P(i,j).\log\sum_{i=0}^{N_g-1} P(i,j)\right\}$	(16)

a family of Gabor filters as described by equation (19). The convolution of image $I(x)$ and a Gabor filter $\psi_j(\vec{x})$ can be defined as follows:

$$J_j(\vec{x}) = I(\vec{x}) * \psi_j(\vec{x}) \tag{19}$$

Here, $*$ denotes the convolution operator, and $J_j(\vec{x})$ is the Gabor filter response of the image block with orientation ϕ_μ and scale k_v. This is referred to as a wavelet transform because the family of kernels are self-similar

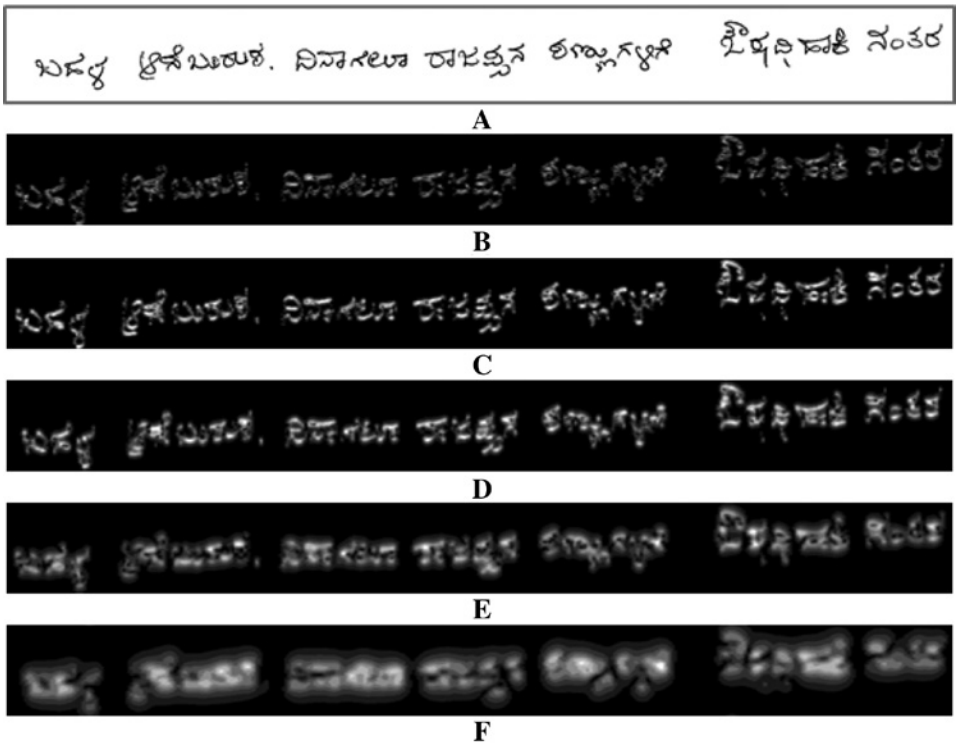

Figure 3: Output Images After Applying Gabor Wavelets on (A) a Sample Handwritten *Kannada* Text-Line Image for (B–F) Five Scales and Orientation = 45°.

and are generated from one mother wavelet by scaling and rotation of frequency. This transform extracts features oriented along ϕ_μ and for the frequency k_υ. Each combination of μ and υ results in a sub-band of same dimension as the input image I. For the present work, $\mu \in \{0, 1, \ldots, 5\}$ and $\upsilon = \{1, 2, \ldots, 5\}$. Five frequency scales and six orientations would yield 30 sub-bands. The Gabor wavelet output images for the sample handwritten text-line and word images are shown in Figures 3 and 4, respectively. For the feature extraction purpose, we computed the energy and entropy values [10] from each of the sub-bands of the Gabor wavelet transform, which create 60 dimensional feature vectors.

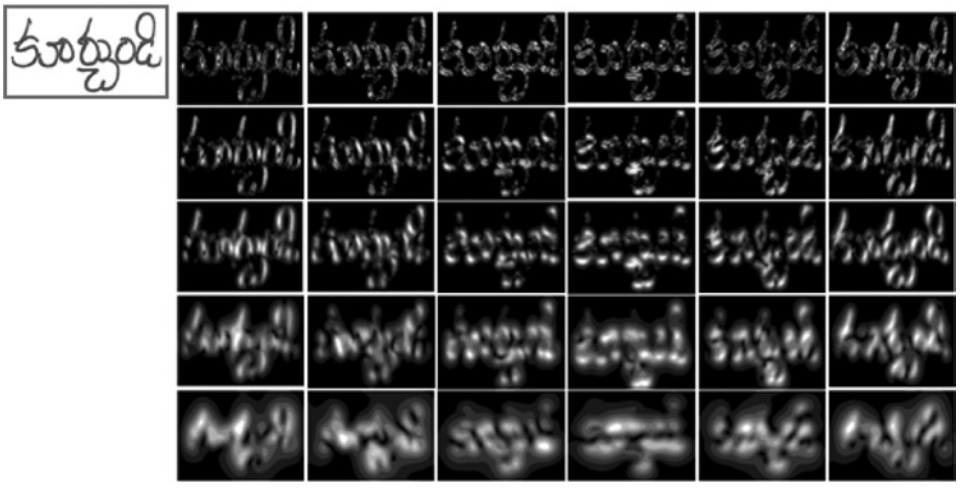

Figure 4: Output Images of Gabor Wavelet Transform on a Sample Handwritten *Telugu* Word Image (Left) for Five Different Scales and Six Different Orientations.

3 Classifier Combination Using DS Theory of Evidence

Classifier combination can be grouped into different categories based on the stage at which the process is applied, type of information (classifier output) being fused, and the number and type of classifiers being combined as mentioned in Ref. [40]. In this case, the classifier combination is applied on the measurement level information provided by the classifier through the confidence scores for every output class. A set of heterogeneous classifiers are generated by training the same classifier with different feature sets and tuning them to optimal values of their parameters. This procedure eliminates the need for normalization of the confidence scores provided by different classifiers. In the MLP classifier used here, the last layer has nodes containing a final score for each of the 12 output classes, which are used for the rule-based combination. So, it is a decision level combination that is carried out without having to consider much about the ideas behind the feature extraction and classification processes.

The DS framework [30] is based on the view whereby propositions are represented as subsets of a given set W, referred to as a frame of discernment. The propositions of interest are in a one-to-one correspondence with the subsets of W. Evidence can be associated to each proposition (subset) to express the uncertainty (belief) that has been observed. Evidence is usually computed based on a density function m called BPA, and $m(p)$ represents the belief exactly committed to the proposition p. If $m(p) > 0$, then p is said to be discerned by the BPA m and is called a focal element [8].

The DS theory has an operation called *Dempster's rule of combination* that aggregates two (or more) bodies of evidence defined within the same frame of discernment into one body of evidence. Let m_1 and m_2 be two BPAs defined in W. The new body of evidence is defined by the BPA $m_{1,2}$ as:

$$m_{1,2}(A) = \begin{cases} 0 & \text{if } A = \varnothing \\ \dfrac{1}{1-K} \displaystyle\sum_{B \cap C = A} m_1(B) m_2(C) & \text{if } A \neq \varnothing \end{cases} \tag{20}$$

where, $K = \displaystyle\sum_{B \cap C = \varnothing} m_1(B) m_2(C)$, and A is the intersection of subsets B and C.

In other words, the Dempster's combination rule computes a measure of agreement between two bodies of evidence concerning various propositions determined from a common frame of discernment. The rule focuses only on those propositions that both bodies of evidence support. Here, m takes into account the BPA's associated to the propositions discerned by m_1 and m_2. The denominator is a normalization factor that ensures that m is a BPA, called the conflict.

To address the issue of conflict, the Yagar's modification [44] provides a different normalization factor using the ground probability mass assignment (designated by q). The combined ground probability assignment is defined as:

$$q(A) = \sum_{B \cap C = A} m_1(B) m_2(C) \tag{21}$$

where, A is the intersection of subsets B and C [both in the power set $P(X)$], and $q(A)$ denotes the ground probability assignment associated with A. The Yagar's modification of the DS theory has been implemented in the paper with the normalizing factor as 1.

The BPA scheme, applied here, first considers the top three classes of every data input from the confidence values given by the classifier's output. This selection is then used to form three subsets for each of the two classifiers given by:

{Highest – ranked class confidence},
{Highest – ranked class confidence, second highest-ranked class confidence},
{Highest – ranked class confidence, second highest-ranked class confidence,
third highest – ranked class confidence}

In order to get the required mass assignments for the subsets formed from the two classifier outputs, two different procedures, *namely*, BPA1 and BPA2, are followed. These subsets with their masses act as two information sources that have to be combined using the rule.

BPA1: The probability assignment for a subset containing a single element is the maximum class confidence output of the MLP classifier divided by the sum of its all outputs. For the other sets, they are considered as the union of single elements, and then, the probability assignments for all these singletons are computed. Finally, the probability assignment of the entire subset is determined to be the minimum of the probability assignments of the constituent singletons of the subset.

BPA2: In the second procedure, which has been proposed in this paper, the BPA1s for the sets from a given classifier gets multiplied with the overall accuracy of that classifier on the dataset. Hence, the results of the classifier's output have been assigned a weightage based on the performance of the classifier on the particular feature set given by the following relation:

$$BPA2(I_{i,j}) = \text{Accuracy}_i * BPA1(I_{i,j}) \tag{22}$$

where $I_{i,j}$ represents the subset (forming the information source) from the ith classifier and the jth proposition subset.

After this, two closely related schemes, *namely*, *S*1 and *S*2 have been considered for the initialization of the output class confidences before the combination processes, as defined below:

S1: There is no initial confidence assigned to any of the classes; hence, there is no bias before the combination.

S2: This *novel* procedure provides an initial bias to every class that appears in the top three positions of the classification result. These classes are initialized to the confidence values that had been assigned to the classes by the individual classifiers. Through this process, the original classifier's outcome has a direct influence over the confidence values that the classes receive after going through the combination process. So, the initialization of the top three classes gets done in the following way:

$$\text{Class confidence}\,(C_i) = \text{Class confidence}\,(C_i) + \text{Classifier output confidence}\,(S_{j,i}) \tag{23}$$

where, C_i is the class corresponding to the ith best confidence score, $S_{j,i}$ gives the confidence score for the ith best confidence score from the jth classifier, and $i = 1, 2, 3$ for the top three classifier output classes.

After the initialization and BPA steps, the confidence values for each of the output classes is generated using the DS rule of combination given in equations (20) and (21). Only the classes of the considered sets from the two sources have their products accumulated for a single intersection, while the products of the other set and assigned values accumulated to the output class corresponding to the intersection. If there is no intersection, the remaining portion of the unit spectrum of confidence values in the set consisting of the remaining classes gets multiplied with the BPA assigned to the current set and is added to the confidence generated for the classes in the set. Other cases do not contribute to the output class confidence in our application.

Four different procedures, introduced in this paper to implement the ideas that have been defined earlier, can be summed up as follows along with the schematic diagram illustrated in Figure 5:

P1: It implements the DS theory of evidence with Yagar's modification. The BPA is derived using procedure BPA1, and there is no class confidence initialization as in *S*1.

P2: A bias is introduced for the classes that appear in the top three positions of the classifier output, as described in *S*2. These candidate classes get initialized with the value provided for that class by the respective classifiers. The BPA1 procedure is applied in generating the information sources to be combined. Now, the combination process is carried out to include the contributions from the common classes among the subset of propositions.

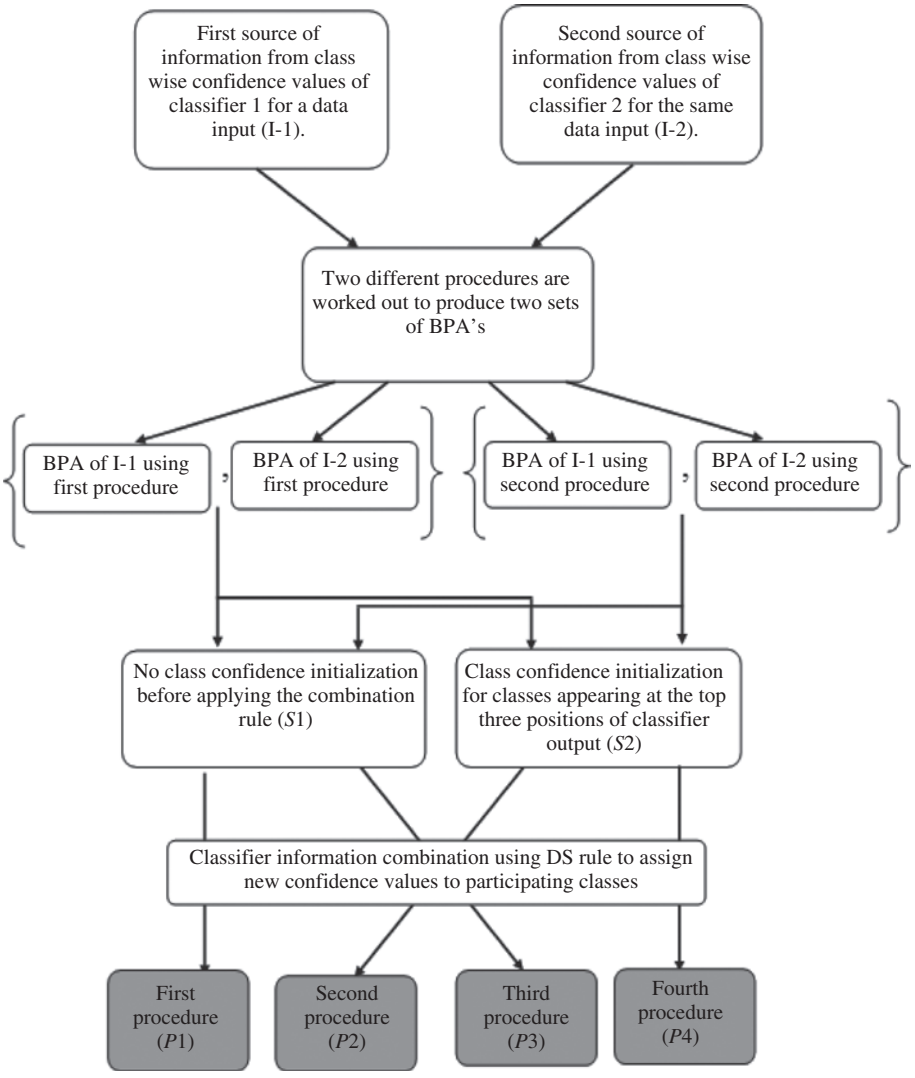

Figure 5: Schematic Diagram of the Four Proposed Procedures Based on DS Theory.

P3: A change in the BPA scheme to include the performance measure of the individual classifier is proposed in this combination (BPA2). The overall accuracy of the classifier is an indicator of the belief, which can be put into the result of the classifier, and hence, getting that fraction multiplied with the confidence values serves to allow a better scope for combination. Here, the classes have no initial bias before the combination (*S*1).

P4: This procedure incorporates both the new variations that have been proposed in this paper at the two different stages of combination. So, there is an initialization of the classes with the confidence values from the classifier along (*S*2) with a basic probability factored with the overall accuracy of the classifier (BPA2).

4 Results and Discussion

4.1 Preparation of Database

At present, no standard database of handwritten *Indic* scripts are available in the public domain. Hence, we have created our own database of handwritten documents in the laboratory. The document pages for the database

Figure 6: Sample Handwritten Text-Line Images Written in 12 Different *Indic* Scripts.

are collected from different sources on request. Participants of this data collection drive are asked to write a few text lines inside A-4-size pages. No other restrictions are imposed regarding the content of the textual materials. Handwritten text lines were written in 12 official scripts of India. The document pages are digitized at 300 dpi resolution and stored as gray tone images. The scanned images may contain noisy pixels, which are removed by applying a Gaussian filter [10]. Sample snapshots of text-line images written in 12 different scripts are shown in Figure 6. Finally, 3600 handwritten text-line images are created, with exactly 300 text lines per script.

For the application of the DS theory at word level script identification, the words are cropped automatically from the text-line images. After that, the words having a length of less than three candidate characters are excluded. This is because a less number of characters present in the text words may not be useful for identifying the scripts. Hence, a set of 500 text words per script is selected in performing the experiment at word level. Figure 7 shows samples of the word images written in 12 different scripts.

4.2 Performance Analysis at Text-Line Level

The DS procedure, described above, is applied on a dataset of 3600 text lines divided into 12 classes with equal number of instances in each of them. Twelve classes refer to the 12 *Indic* scripts that have been studied

Figure 7: Sample Handwritten Word Images Written in 12 *Indic* Scripts.

before and for which the MLP classifier results can be obtained with high accuracy. The classes numbered from **A** to **L** are *Bangla*, *Devanagari*, *Gujarati*, *Gurumukhi*, *Kannada*, *Malayalam*, *Oriya*, *Tamil*, *Telugu*, *Urdu*, *Roman*, and *Manipuri* in that particular order.

First, the confusion matrix that is obtained from the MLP-based classifier on text-line level dataset using the GLCM feature along with the overall accuracy is presented. Then, the result generated by the same classifier on the Gabor wavelet transform feature applied on the same dataset is also presented. The results have undergone a threefold cross-validation for the classifier parameter values to obtain the optimal results for the dataset, and the parameter values are given before reporting each result. The GLCM feature set consisting of 80 feature values for every input image is fed into the MLP classifier with 50 hidden layer neurons and a learning rate of 0.8. Here, 500 iterations are allowed with an error tolerance of 0.1. The overall accuracy obtained is 90.58%, and the confusion matrix generated in this case is given in Figure 8A. The **R** column in the table refers to the rejection of the input by the recognition module, but the class confidences that are associated with them get accounted for during the combination process.

The Gabor wavelet transform feature set, consisting of 60 feature values for every input data, is fed into the MLP classifier with 40 hidden layer neurons and a learning rate of 0.8. The same error tolerance and the number of iterations, as applied in the case of the GLCM features, are allowed here. A maximum recognition accuracy of 92.06% has been noted. The confusion matrix is shown in Figure 8B.

Now, the confidence values provided to the classes for every input data by the classifiers on the two sets of features form the input for the DS combination procedure. These are the two complementary sources of information about the data, which are considered along with their BPAs using the class confidences. The DS combination is then applied as described in Section 3, and the results are tabulated.

Figure 9A shows the result for the combination for procedure $P1$ where an overall accuracy of 95.91% is reported. The result for procedure $P2$ is shown in Figure 9B where the overall accuracy obtained is 96.28%. The classification result for procedure $P3$ is reported in Figure 9C. In this case, an overall accuracy of 96.0%

A

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L	R
A	294	0	0	0	6	0	0	0	0	0	0	0	0
B	0	264	6	6	0	3	3	0	0	0	9	9	0
C	0	9	279	6	0	0	3	3	0	0	0	0	0
D	0	9	9	273	0	0	3	0	6	0	0	0	3
E	3	54	3	6	201	0	0	0	15	0	18	0	0
F	0	0	15	6	0	246	6	6	3	18	0	0	0
G	0	0	9	0	0	0	279	9	3	0	0	0	0
H	0	0	3	3	0	0	3	288	0	3	0	0	0
I	0	0	6	0	0	15	0	21	258	0	0	0	0
J	0	0	0	0	0	3	0	0	0	297	0	0	3
K	3	3	0	6	0	0	0	0	0	0	288	0	0
L	0	6	0	0	0	0	0	0	0	0	0	294	0

B

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L	R
A	173	1	12	3	12	66	3	12	9	0	0	9	0
B	0	294	3	3	0	0	0	0	0	0	0	0	0
C	0	0	294	0	0	0	6	0	0	0	0	0	0
D	0	3	0	297	0	0	0	0	0	0	0	0	0
E	18	0	0	0	211	0	20	18	0	0	33	0	8
F	8	3	6	0	5	272	0	6	0	0	0	0	0
G	0	0	0	0	0	0	300	0	0	0	0	0	0
H	0	0	0	0	0	0	0	300	0	0	0	0	0
I	0	0	0	3	0	3	0	0	294	0	0	0	0
J	0	0	0	0	0	0	0	0	0	300	0	0	0
K	0	0	0	0	0	0	0	0	0	0	300	0	0
L	0	0	3	15	0	0	0	0	0	3	0	279	0

Figure 8: Class-Wise Classification Statistics on Text-Line-Level Script Datasets Using (A) GLCM and (B) Gabor Wavelet Transform.

is observed. In this last procedure *P*4, the confusion matrix is illustrated in Figure 9D, and in this case, the overall accuracy is 96.25%.

The best result is obtained for the second procedure *P*2, and the detailed class-wise performance measures are provided in Table 2. It can be seen from the confusion matrices that the *Devanagari*, *Gujarati*, *Gurumukhi*, *Oriya*, *Tamil*, *Urdu*, and *Roman* scripts have accuracies very close to 100%, whereas, the *Kannada* has the least classification accuracy and gets confused with the *Devanagari*.

It is worth mentioning that the improvement is substantial, as there is around 4% increase in the accuracy for a dataset of 3600 samples, over the best performing classifier. Here, the cases with contradictory outputs from the two classification models get combined to provide correct results for some samples. Classes, which do not have the clear majority in the results of either of the classifiers, may come out as the decision class due to its position among the top three in both the outputs and the cumulative intersection value becoming dominant. Therefore, a lot of possibilities are looked into during the combination procedure to be able to make final decisions, which can draw insights from the previous level of classification. In all the schemes that have been presented in this paper, the basic MLP classification knowledge has been incorporated as much as possible in this second level of classification to obtain improvements in accuracy for the recognition task.

4.3 Performance Analysis at Word Level

The DS procedure is again applied on a word-level dataset of 6000 images where each script contains exactly about 500 word images. The 12 script classes are numbered from **A** to **L** as described in the previous subsection. First, both the GLCM and Gabor wavelet transform feature sets are applied on the prepared

A

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	286	0	1	0	8	2	1	0	2	0	0	0
B	0	298	0	2	0	0	0	0	0	0	0	0
C	0	0	298	0	0	0	2	0	0	0	0	0
D	0	2	1	297	0	0	0	0	0	0	0	0
E	7	30	0	3	220	0	3	1	12	0	24	0
F	0	0	4	1	1	284	3	6	0	1	0	0
G	0	0	1	0	0	0	298	1	0	0	0	0
H	0	0	1	0	0	0	0	299	0	0	0	0
I	0	0	3	2	0	2	0	1	292	0	0	0
J	0	0	0	0	0	0	0	1	0	299	0	0
K	1	0	0	0	0	0	0	0	0	0	299	0
L	0	2	0	4	0	0	0	0	0	0	0	294

B

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	285	0	1	0	4	4	1	2	3	0	0	0
B	0	300	0	0	0	0	0	0	0	0	0	0
C	0	0	298	0	0	0	2	0	0	0	0	0
D	0	2	0	298	0	0	0	0	0	0	0	0
E	6	24	0	3	233	0	5	1	10	0	18	0
F	0	0	4	2	1	281	3	6	0	3	0	0
G	0	0	0	0	0	0	299	1	0	0	0	0
H	0	0	1	0	0	0	0	299	0	0	0	0
I	0	0	2	1	0	2	0	0	295	0	0	0
J	0	0	0	0	0	0	0	0	0	300	0	0
K	3	0	0	0	0	0	0	0	0	0	297	0
L	0	2	0	6	0	0	0	0	0	0	0	292

C

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	277	1	2	0	4	7	3	2	4	0	0	0
B	0	297	0	3	0	0	0	0	0	0	0	0
C	0	0	298	0	0	0	2	0	0	0	0	0
D	0	2	1	297	0	0	0	0	0	0	0	0
E	7	23	0	3	233	0	5	1	11	0	17	0
F	0	0	3	1	2	282	4	6	0	2	0	0
G	0	0	1	0	0	0	298	1	0	0	0	0
H	0	0	0	0	0	0	0	300	0	0	0	0
I	0	0	2	1	0	2	1	0	294	0	0	0
J	0	0	0	0	0	0	0	0	0	300	0	0
K	3	0	0	0	0	0	0	0	0	0	297	0
L	0	4	0	2	0	0	0	0	0	0	0	294

D

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	283	0	2	0	3	5	2	2	3	0	0	0
B	0	300	0	0	0	0	0	0	0	0	0	0
C	0	0	298	0	0	0	2	0	0	0	0	0
D	0	1	0	299	0	0	0	0	0	0	0	0
E	7	24	0	3	234	0	5	1	9	0	17	0
F	0	0	6	2	1	280	3	5	0	3	0	0
G	0	0	0	0	0	0	299	1	0	0	0	0
H	0	0	1	0	0	0	0	299	0	0	0	0
I	0	0	2	1	0	2	0	0	295	0	0	0
J	0	0	0	0	0	0	0	0	0	300	0	0
K	3	0	0	0	0	0	0	0	0	0	297	0
L	0	2	0	6	0	0	0	0	0	0	0	292

Figure 9: Class-Wise Classification Statistics Obtained by: (A) First Procedure, (B) Second Procedure, (C) Third Procedure, and (D) Fourth Procedure.

word-level dataset and script recognition accuracies of 75.28% and 89.07% are noted using MLP classifiers, respectively. The confusion matrices for the GLCM and Gabor wavelet transform feature sets are shown in Figure 10A and B, respectively.

Table 2: Detailed Performance Measures of the Individual Script Classes for the Best Performing Scheme at Text-Line Level.

Class	A	B	C	D	E	F	G	H	I	J	K	L
Precision	0.969	0.915	0.974	0.961	0.979	0.979	0.965	0.968	0.958	0.990	0.942	1.000
Recall	0.950	1.000	0.93	0.993	0.776	0.936	0.996	0.996	0.983	1.000	0.990	0.973
F-measure	0.959	0.956	0.983	0.977	0.866	0.957	0.980	0.982	0.970	0.995	0.965	0.986

A

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L	R
A	271	22	5	25	16	14	33	41	20	0	39	14	18
B	18	459	0	8	7	0	1	1	0	0	6	0	0
C	0	0	461	0	6	10	0	17	2	2	0	2	28
D	28	8	0	432	3	2	2	0	6	0	17	2	4
E	4	9	4	9	352	15	0	2	0	0	8	97	36
F	36	0	20	4	48	313	19	29	7	1	9	14	45
G	45	2	14	12	14	14	321	29	18	1	25	5	25
H	25	0	29	15	12	30	1	326	21	12	10	19	50
I	50	2	4	19	0	6	19	28	367	0	5	0	0
J	0	0	20	0	5	1	0	15	0	451	0	8	39
K	39	8	5	30	23	12	22	34	0	1	323	3	1
L	1	3	14	1	13	13	0	4	0	7	3	441	0

B

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L	R
A	405	4	2	14	1	3	24	15	6	0	14	12	1
B	1	477	0	5	0	0	1	0	0	0	3	13	1
C	1	0	495	0	0	0	1	1	1	0	0	1	4
D	4	6	0	473	0	0	0	1	0	0	1	15	0
E	0	0	0	0	491	6	0	1	0	0	0	2	0
F	0	0	2	0	20	454	0	16	0	7	0	0	2
G	2	0	7	2	0	1	439	14	18	0	15	3	5
H	5	0	5	1	15	32	1	379	16	5	16	15	8
I	8	1	4	1	0	0	13	4	464	0	4	1	0
J	0	0	0	0	2	4	0	2	0	489	0	3	24
K	24	2	6	7	2	0	31	8	7	0	402	11	2
L	2	0	22	24	20	12	6	10	18	1	10	376	0

Figure 10: Class-Wise Classification Statistics on Word-Level Script Datasets Using (A) GLCM and (B) Gabor Wavelet Transform.

Procedures *P*1 to *P*4 are applied on the classification results from the word level data. Figure 11A–D shows the classification results for the combination procedures discussed in Section 3. *P*1 gives an accuracy of 94.53%, followed by 93.17% for *P*2. *P*3 and *P*4 have overall accuracies of 94.20% and 94.27%, respectively.

The highest recognition accuracy at word level is obtained for the first procedure *P*1. Table 3 enlists the detailed class-wise performance measures attained for this best case.

In some cases, where one of the dominating class confidence values is considerably higher than the other highest class confidence from another source, the combination reflects the change in favor of the higher confidence class. Hence, the highly confident but wrong decisions do not get altered. Incorrect classifications are also seen in cases where the top three classes do not cover most of the unit spectrum of confidence values for that classifier output.

In order to compare the performance of the DS theory-based procedures with other popular rule-based classifier combination approaches, algorithms like the Borda count [41], sum rule, product rule, and max rule are also implemented. The confidence outputs of the two MLP classifiers are the inputs for the combination procedures, and a new set of output confidence values are obtained. The Borda count algorithm ranks the two input sources and adds the rank for each output class. The class having the best rank is selected as the decision. The sum rule adds the confidence values given to the class from multiple sources. The class

A

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	451	6	4	2	5	2	7	3	6	0	6	8
B	1	485	0	4	0	0	0	1	0	0	2	7
C	0	0	496	0	0	1	0	1	1	0	0	1
D	4	3	0	484	0	0	0	1	1	0	3	4
E	0	0	0	0	494	4	0	0	0	0	2	0
F	0	0	4	1	14	460	2	9	0	3	0	7
G	6	0	2	2	1	0	454	10	11	1	13	0
H	8	0	7	3	6	6	5	448	8	3	0	6
I	6	2	1	5	0	0	5	6	469	1	5	0
J	0	0	4	0	0	0	0	2	0	492	1	1
K	5	1	5	6	1	0	6	6	2	1	466	1
L	1	1	4	5	9	1	1	1	0	1	3	473

B

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	427	6	4	8	5	2	10	9	9	0	9	11
B	4	476	0	5	1	0	0	1	0	0	5	8
C	1	0	492	0	0	0	0	6	1	0	0	0
D	4	3	0	483	0	0	1	1	1	0	3	4
E	1	3	0	2	481	3	0	0	0	0	1	9
F	2	0	5	1	14	457	2	16	0	1	0	2
G	3	0	3	3	1	2	460	8	7	1	12	0
H	9	0	7	4	7	14	5	426	12	3	7	6
I	10	1	1	3	0	0	9	7	463	1	5	0
J	0	0	3	0	0	0	0	5	0	489	1	2
K	5	1	5	7	4	0	8	9	2	1	456	2
L	0	1	3	6	3	0	1	2	0	2	2	480

C

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	449	6	3	4	5	1	6	7	9	0	5	5
B	4	478	0	5	2	0	0	0	0	0	5	6
C	1	0	491	0	0	2	0	5	1	0	0	0
D	4	3	0	483	0	0	1	1	1	0	3	4
E	0	2	0	0	494	3	0	0	0	0	1	0
F	3	0	4	1	13	458	1	15	0	2	1	2
G	4	0	4	3	1	2	456	10	8	1	11	0
H	7	0	5	7	5	7	5	451	3	4	0	6
I	5	1	2	3	0	0	6	5	474	1	3	0
J	0	0	4	0	0	0	0	3	0	491	1	1
K	7	1	5	5	4	0	8	9	2	1	457	1
L	0	5	3	8	5	1	1	2	1	2	2	470

D

Class/Class	A	B	C	D	E	F	G	H	I	J	K	L
A	449	6	3	4	5	1	6	7	9	0	5	5
B	4	482	0	3	2	0	0	0	0	0	5	4
C	1	0	491	0	0	2	0	5	1	0	0	0
D	4	3	0	483	0	0	1	1	1	0	3	4
E	0	2	0	0	494	3	0	0	0	0	1	0
F	3	0	4	1	13	458	1	15	0	2	1	2
G	4	0	4	3	1	2	456	10	8	1	11	0
H	7	0	5	7	5	7	5	451	3	4	0	6
I	5	1	2	3	0	0	6	5	474	1	3	0
J	0	0	4	0	0	0	0	3	0	491	1	1
K	7	1	5	5	4	0	8	9	2	1	457	1
L	0	5	3	8	5	1	1	2	1	2	2	470

Figure 11: Class-Wise Classification Statistics Obtained by (A) First Procedure, (B) Second Procedure, (C) Third Procedure, and (D) Fourth Procedure.

with the maximum confidence is the result of the combination. Similarly, the product rule accumulates the products of the confidence values of the classes and maximizes it for its decision. The max rule selects the class corresponding to the highest confidence value among all the sets of confidence score outputs.

Table 3: Detailed Performance Measures of the Individual Script Classes for the Best Performing Scheme at Word Level.

Class	A	B	C	D	E	F	G	H	I	J	K	L
Precision	0.936	0.974	0.941	0.945	0.932	0.970	0.946	0.918	0.942	0.980	0.930	0.931
Recall	0.902	0.970	0.992	0.968	0.988	0.920	0.908	0.896	0.992	0.984	0.932	0.946
F-measure	0.918	0.971	0.965	0.956	0.959	0.944	0.927	0.907	0.966	0.981	0.930	0.938

Table 4 records the accuracy obtained at the at both the text-line level and word level script data by these procedures.

All the classifier combination procedures perform well in improving the overall classification accuracy of the script recognition task on the aforementioned datasets. This means that the classifiers provide complementary sets of information that get combined effectively. In order to justify this hypothesis, correlation analysis is performed to arrive at the Edward Spearman correlation coefficient [2] that provides a measure of the rank level correlation. Edward Spearman's formula for calculating the rank correlation coefficient (R_c) is given as follows:

$$R_c = 1 - \frac{6\sum_{i=1}^{n} D_i^2}{n(n^2-1)} \tag{24}$$

where D is the difference between the ranks of two classifiers, and n is the number of classes in each classifier. Values of 0.435 and 0.358 are obtained for the text-line level and word level classification outputs, respectively. The low correlation values statistically validate the combination of these two sources of information.

The combination of the GLCM and Gabor wavelet transform using the present DS theory is now compared with their individual as well as natural combination. The classification is done using the MLP classifier, and

Table 4: Performance Comparison of the Present Classifier Combination Technique with Other Combination Techniques at Both Text-Line and Word Levels (Bold Style Shows the Best Case).

Classifier combination methodology	Recognition accuracy (%)	
	Text-line level	Word level
Borda count	92.86	90.28
Sum rule	95.91	93.07
Product rule	94.55	91.95
Max rule	96.0	89.68
Best result using DS theory	**96.28**	**94.53**

Table 5: Performance Comparison of the Present Feature Set with Their Combination at Both Text-Line and Word Levels (Bold Style Shows the Best Case).

Features/methodology	Recognition accuracy (%)	
	Text-line level	Word level
GLCM	90.58	75.28
Gabor wavelets transform	92.06	89.07
GLCM + Gabor wavelet transform	94.30	91.95
First procedure using DS theory	**96.28**	**94.53**
Second procedure using DS theory	95.91	93.17
Third procedure using DS theory	96.00	94.20
Fourth procedure using DS theory	96.25	94.27

the performance comparison is shown in Table 5. It can be observed from Table 5 that the best recognition accuracy is achieved by the first procedure using the DS theory as it outperforms all other combinations. It is to be also noted that all the four procedures proposed in this paper outperform the combination of the GLCM and Gabor wavelet transform feature set. This validates the suitability of the proposed methodology using the DS theory in improving the classification accuracy for solving the problem of handwritten *Indic* script identification.

5 Conclusion

This is the first application of the DS theory-based multi-classifiers' information fusion in the domain of handwritten *Indic* script recognition. According to the result, a considerable amount of improvement over the original recognition accuracy provided by either of the classifiers has been observed. From the four closely related schemes, the increment in the range of 4–6% over the best classification results of the MLP classifier on sample database sizes of 3600 handwritten text-line and 6000 handwritten word images has been reported. Uniting these classifier outputs for feature sets in this way can, thus, help to augment the recognition ability of the overall system and make the final decision. The process, described here, combines the classifiers' results on two feature sets using the DS rule. In the future, we can think of designing a tree-like hierarchical structure to combine more number of classifiers' outcomes. It may combine two classifiers' outputs at a time and then report the class-wise confidences that the combination rule provides. At the next level, these intermediate results can be combined again moving to a higher level of the structure. Finally, a decision can be made, thus, incorporating a lot of pertinent contributing sources of information in the form of classifiers as well as feature sets.

Acknowledgment: The authors are thankful to the Center for Microprocessor Application for Training Education and Research (*CMATER*) and the Project on Storage Retrieval and Understanding of Video for Multimedia (*SRUVM*) of the Computer Science and Engineering Department, Jadavpur University, for providing infrastructure facilities during the progress of the work. The authors of this paper are also thankful to all those individuals who willingly contributed in developing the handwritten *Indic* script database used in the current research.

Bibliography

[1] S. Basu, R. Sarkar, N. Das, M. Kundu, M. Nasipuri and D. K. Basu, Handwritten Bangla digit recognition using classifier combination through DS technique, in: *Proc. of 1st International Conference on Pattern Recognition and Machine Intelligence (PReMI)*, Springer, LNCS 3776, Kolkata, India, pp. 236–241, 2005.

[2] A. G. Bluman, *Elementary statistics: a step by step approach*, 7th Edition, McGraw Hill, New York, 2009.

[3] A. Busch, W. W. Boles and S. Sridharan, Texture for script identification, *IEEE Trans. Pattern Anal. Mach. Intell.* **27** (2005), 1720–1732.

[4] S. Chanda, S. Pal, K. Franke and U. Pal, Two-stage approach for word-wise script identification, in: *Proc. of 10th IEEE International Conference on Document Analysis and Recognition (ICDAR)*, Barcelona, Catalonia, Spain, pp. 926–930, 2009.

[5] S. Chaudhury, G. Harit, S. Madnani and R. B. Shet, Identification of scripts of Indian languages by combining trainable classifiers, In: *Proc. of Indian conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, Bangalore, India, 2000.

[6] B. B. Chaudhuri and U. Pal, An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi), In: *Proc. of 4th IEEE International Conference on Document Analysis and Recognition (ICDAR)*, Ulm, Germany, pp. 1011–1015, 1997.

[7] J. G. Daugman, Uncertainty relation for resolution in space, spatial-frequency, and orientation optimized by two-dimensional visual cortical filters, *J. Opt. Soc. Am.* **2** (1985), 1160–1169.

[8] A. P. Dempster, A generalization of Bayesian inference, *J. R. Stat. Soc. B* **30** (1968), 205–247.

[9] D. Dhanya, A. G. Ramakrishnan and P. B. Pati, Script identification in printed bilingual documents, *Sadhana* **27** (2002), 73–82.

[10] R. C. Gonzalez and R. E. Woods, *Digital image processing*, Vol. I, Prentice-Hall, India, 1992.

[11] R. M. Haralick and L. Watson, A facet model for image data, *Comput. Vis. Graph. Image Process* **15** (1981), 113–129.
[12] R. M. Haralick, K. Shanmungam and I. Dinstein, Textural features of image classification, *IEEE Trans. Syst. Man Cybern.* **3** (1973), 610–621.
[13] P. S. Hiremath, S. Shivshankar, J. D. Pujari and V. Mouneswara, Script identification in a handwritten document image using texture features, in: *Proc. of IEEE 2nd International Conference on Advance Computing*, Patiala, India, pp. 110–114, 2010.
[14] T. K. Ho, *A theory of multiple classifier systems and its application to visual word recognition*, PhD thesis, State University of New York at Buffalo, 1992.
[15] T. K. Ho, J. J. Hull and S. N. Srihari, Decision combination in multiple classifier systems, *IEEE Trans. Pattern Anal. Mach. Intell.* **16** (1994), 66–75.
[16] J. Hochberg, L. Kerns, P. Kelly and T. Thomas, Automatic script identification from images using cluster-based templates, in: *Proc. of the 3rd IEEE International Conference on Document Analysis and Recognition*, Montreal, Canada, Vol. 1, pp. 378–381, 1995.
[17] J. Hochberg, L. Kerns, P. Kelly and T. Thomas, Automatic script identification from images using cluster based templates, *IEEE Trans. Pattern Anal. Mach. Intell.*, **19** (1997), 176–181.
[18] G. D. Joshi, S. Garg and J. Sivaswamy, Script identification from Indian documents, in: *Lecture Notes in Computer Science: International Workshop on Document Analysis Systems*, Nelson, LNCS-3872, pp. 255–267, Feb. 2006.
[19] J. Kittler, M. Hatef, R. Duin and J. Matas, On combining classifiers, *IEEE Trans. Pattern Anal. Mach. Intell.* **20** (1998), 226–239.
[20] D. Lee, *A theory of classifier combination: the neural network approach*, PhD thesis, State University of New York at Buffalo, 1995.
[21] T. S. Lee, Image representation using 2D Gabor wavelets, *IEEE Trans. Pattern Anal. Mach. Intell.* **18** (1996), 1–13.
[22] E. Mandler and J. Schuerman, *Pattern recognition and artificial intelligence*, Elsevier Science Publishers, Amsterdam, North-Holland, 1988. DOI: 10.1002/bimj.4710320512.
[23] C. Nadal, R. Legault and C. Y. Suen, Complementary algorithms for the recognition of totally unconstrained handwritten numerals, in: *Proc. of 10th IEEE International Conference on Pattern Recognition*, Atlantic City, New Jersey, USA, Vol. A, pp. 434–449, 1990.
[24] J. Ni, J. Luo and W. Liu, 3D palmprint recognition using Dempster-Shafer fusion theory, *J. Sens.* **2015** (2015), Article ID: 252086, 1–7. DOI: http://dx.doi.org/10.1155/2015/252086
[25] M. C. Padma and P. A. Vijaya, Global approach for script identification using wavelet packet based features, *Int. J. Signal Process. Image Process. Pattern Recognit.* **3** (2010), 29–40.
[26] U. Pal and B. B. Chaudhuri, Identification of different script lines from multi-script documents, *Image Vis. Comput.* **20** (2002), 945–954.
[27] U. Pal, S. Sinha and B. B. Chaudhuri, Word-wise script identification from a document containing English, Devnagari and Telugu Text, in: *Proc. of 2nd National Conference on Document Analysis and Recognition (NCDAR)*, PES, Mandya, Karnataka, India, pp. 213–220, 2003.
[28] U. Pal, S. Sinha and B. B. Chaudhuri, Multi-script line identification from Indian documents, in: *Proc. of 7th IEEE International Conference on Document Analysis and Recognition (ICDAR)*, Edinburgh, Scotland, UK, pp. 880–884, 2003.
[29] P. B. Pati and A. G. Ramakrishnan, Word level multi-script identification, *Pattern Recognit. Lett.* **29** (2008), 1218–1229.
[30] G. Shafer, *A mathematical theory of evidence*, Princeton University Press, Princeton, New Jersey, ISBN: 9780691100425 1976.
[31] M. Shoyaib, M. Abdullah-Al-Wadud and O. Chae, A skin detection approach based on the Dempster-Shafer theory of evidence, *Int. J. Approx. Reason.* **53** (2012), 636–659.
[32] P. K. Singh, I. Chatterjee and R. Sarkar, Page-level handwritten script identification using modified log-Gabor filter based features, in: *Proc. of 2nd IEEE International Conference on Recent Trends in Information Systems (ReTIS)*, Kolkata, India, pp. 225–230, 2015.
[33] P. K. Singh, R. Sarkar and M. Nasipuri, Offline script identification from multilingual Indic-script documents: a state-of-the-art, *Comput. Sci. Rev.* (*Elsevier*), **15–16** (2015), 1–28.
[34] P. K. Singh, S. K. Dalal, R. Sarkar and M. Nasipuri, Page-level script identification from multi-script handwritten documents, in: *Proc. of 3rd IEEE International Conference on Computer, Communication, Control and Information Technology (C3IT)*, Kolkata, India, pp. 1–6, 2015.
[35] P. K. Singh, R. Sarkar, M. Nasipuri and D. Doermann, Word-level script identification for handwritten Indic scripts, in: *Proc. of 13th IEEE International Conference on Document Analysis and Recognition (ICDAR)*, Tunis, Tunisia, pp. 1106–1110, 2015.
[36] P. K. Singh, S. Das, R. Sarkar and M. Nasipuri, Line parameter based word-level Indic script identification system, in: *International Journal of Computer Vision and Image Processing*, PhD thesis, IGI Global Publishers, Hershey, Pennsylvania 17033-1240, USA, Vol. 6, Issue 2, pp. 18–41, 2016.
[37] A. L. Spitz, Determination of the script and language content of document images, *IEEE Trans. Pattern Anal. Mach. Intell.* **19** (1997), 234–245.
[38] C. Y. Suen, C. Nadal, T. Mai, R. Legault and L. Lam, Recognition of totally unconstrained handwritten numerals based on the concept of multiple experts, in: *Proc. of International Workshop on Frontiers in Handwriting Recognition*, Montreal, Canada, pp. 131–143, Apr. 2–3 1990.

[39] T. N. Tan, Rotation invariant texture features and their use in automatic script identification, *IEEE Trans. Pattern Anal. Mach. Intell.* **20** (1998), 751–756.

[40] S. Tulyakov, S. Jaeger, V. Govindaraju and D. Doermann, Review of classifier combination methods, in: S. Marinai and H. Fujisawa (eds.), *Machine Learning in Document Analysis and Recognition, SCI*, Vol. 90, pp. 361–386, Springer, Heidelberg, 2008.

[41] M.Van Erp, L. G. Vuurpijl and L. Schomaker. An overview and comparison of voting methods for pattern recognition, in: *Proc. of 8th International Workshop on Frontiers in Handwriting Recognition (IWFHR-8)*, pp. 195–200, Niagara-on-the-Lake, Canada, 2002.

[42] S. Wood, X. Yao, K. Krishnamurthi and L. Dang, Language identification for printed text independent of segmentation, In: *Proc. of IEEE International Conference on Image Processing*, Washington, DC, USA, Vol. 3, pp. 428–431, 1995.

[43] L. Xu, A. Krzyzak and C. Suen, Methods of combining multiple classifiers and their applications to handwritten recognition, *IEEE Trans. Syst. Man Cybern.* **SMC-22** (1992), 418–435.

[44] R. R. Yager, On the Dempster-Shafer framework and new combination rules, *Inf. Sci.* **41** (1987), 93–137.

[45] B. Zhang and S. N. Srihari. Class-wise multi-classifier combination based on Dempster-Shafer theory, in: *Proc. of 7th IEEE International Conference on Control, Automation, Robotics and Vision, 2002*, ICARCV 2002, Marina Mandarin, Singapore, Vol. 2, pp. 698–703, 2002.