*Sādhanā* Vol. 40, Part 6, September 2015, pp. 1701–1714. © Indian Academy of Sciences

# Feature selection based classifier combination approach for handwritten Devanagari numeral recognition

PRATIBHA SINGH[1,*], AJAY VERMA[1] and NARENDRA S CHAUDHARI[2]

[1]Department of Electronics and Instrumentation Engineering, Institute of Engineering and Technology DAVV, Khandwa Road, Indore 452017, India
[2]Computer Science and Engineering Department, Visvesvaraya National Institute of Technology, Nagpur 440010, India
e-mail: prat_ibh_a@yahoo.com; ajayrt@rediffmail.com; nsc0183@gmail.com

MS received 17 June 2014; revised 20 March 2015; accepted 24 April 2015

**Abstract.** In this paper a method for the recognition of handwritten Hindi numerals is presented. The paper is reporting the effectiveness of the proposed approach, which is utilizing the feature selection based on the Information theory measures. The Multilayer Perceptron (MLP) based classifier combination is used along with feature selection using two criterion functions: (i) Maximum relevance minimum redundancy and (ii) Conditional mutual information maximization. Conditional mutual information based feature selection when driving the ensemble of classifier produces improved recognition results for most of the benchmarking datasets. The improvement is also observed with maximum relevance minimum redundancy based feature selection when used in combination with ensemble of classifiers. The main contribution of the proposed method is that, the method gives quite efficient results utilizing only 10% patterns of the available dataset.

**Keywords.** Conditional mutual information maximization (CMIM); feature selection (FS); minimum redundancy maximum relevance (MRMR); mutual information (MI); ensemble; MLP.

## 1. Introduction

Handwriting recognition is a widely known pattern recognition problem considered almost solved for the isolated English text, but for the handwritten Devanagari script it is not that matured. There are several reasons for the problem considered underdeveloped, some of them are: variability in writing style, existence of multiple forms of writing the same character, existence of touching and fused characters, lack of standard benchmarking and ground truth dataset, lack of corpora and complexity of grammatical formation of the sentences. The character recognition problem can be classified as online and offline. In context of Indian language, only few attempts

*For correspondence

are made that too are limited for the recognition of isolated characters (Bajaj *et al* 2002; Pal *et al* 2007; Hanmandlu & Murthy 2007; Bhattacharya & Choudhary 2009). The process of recognition generally involves three basic steps namely: preprocessing, feature extraction and classification. For obtaining better recognition performance, we used classifier combination instead of single classifier, since the classifier which is good at classifying one class may not be good at the classification of some other class. Combination of classifiers can be attempted at three different levels: data level, feature level and classifier level or decision level. For the classification of handwritten characters the dimensionality reduction plays a very important role. There are two techniques of dimensionality reduction: feature extraction and feature selection. The feature extraction for an image is a process of transforming the image into some other linear or nonlinear plane. For better recognition it is necessary to have an efficient feature extraction method. In feature selection, a subset of features is selected from whole set on the basis of its discriminative power. The choice of a good feature subset is crucial in any classification problem. In most of the classification algorithms we extract features, out of those some of the features have more discriminating power for particular class than the others. So idea used in this study is to select different discriminative features for different classes. The dimensionality reduction technique is combined for improving the performance in terms of recognition efficiency and recognition speed in this study. Some of the recent researches (Cordella *et al* 2008), (Stefano *et al* 2014) used the combination of feature extraction and feature selection. The reason of using combination is that for the case of handwriting most of the dataset is available in the form of images and not as features. We extended this method based on class-wise feature- selection. There are two contributions of this paper: First is the introduction of an ensemble method using the feature selection based on mutual information for Devanagari handwritten numerals. Second is that, a performance comparison is established between minimum redundancy maximum relevance and conditional mutual information maximization algorithms of feature selection. Ensemble is created using class dependent feature selection approach.

The rest of the paper is organized as follows. The section 2 describes the approaches used for the feature selection and section 3 describes the method used for classifier combination. Section 4 describes the results of experiments conducted in this study and in section 5 conclusion is provided.

## 2. Feature selection

The feature selection is the process of selection, from the whole set of available features, the subset allowing the most discriminative power. It is the process of selecting a subset of relevant features for the construction of classifier model. The choice of a good feature subset is crucial in any classification problem. Feature selection methods can be classified into two types, *filters* and *wrapper* (Kohavi & John 1997). The first type is classifier independent, as they are not dedicated to a specific type of classification method. On the contrary, the *wrappers* rely on the performance of one type of classifier to evaluate the quality of a set of features. A procedure for optimal feature selection involves two components:

- Feature selection criterion: It is a feature, that allows us to judge whether one subset of features is better than another (evaluation method).
- Systematic search procedure: It allows us to search through candidate subsets of features and includes the initial state of the search and stopping criteria.

The process of feature selection is represented by iterative algorithm shown in figure 1.

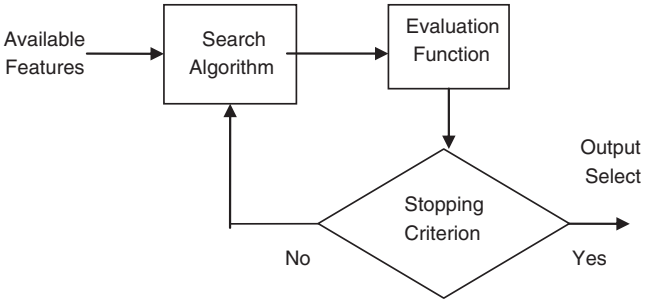


**Figure 1.** The method of feature selection, Dash & Liu (1997).

### 2.1 *Entropy and mutual information*

In feature selection problem, the relevant features have important information about the output, whereas the irrelevant features contain little information regarding the output. The objective of feature selection is to find those features that contain as much information about the output as possible. For this purpose, Shannon's information theory, (Shannon & Weaver 1949) provides a feasible way to measure the information of random variables with entropy and mutual information. Mutual Information (MI) is capable of measuring a general dependence between two features without assuming the distributions of the features, and case based reasoning requires no assumption on the different project features to derive the solutions.

The entropy H(X) is a measure of the uncertainty of a random variable X. The entropy, denoted $H(X)$, quantifies the uncertainty present in the distribution of $X$. It is defined as,

$$H(X) = -\sum_{x \in X} p(x) \log p(x). \tag{1}$$

Here, the log is to be base 2 and entropy is expressed in bits. where the lower case $x$ denotes a possible value that the variable $X$ can adopt from the alphabet $\chi$. The joint entropy of X and Y with joint pdf $p(x, y)$,

$$H(X, Y) = -\sum\nolimits_{x \in X} \sum\nolimits_{y \in Y} p(x, y) \log p(x, \ y). \tag{2}$$

When certain variables are known and others are not, the remaining uncertainty is measured by the conditional entropy,

$$H(Y \,|X) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log p(x \,|y). \tag{3}$$

Therefore, the joint entropy and conditional entropy has the following relation:

$$H(X, \ Y) = H(X) + H(Y \,|X) = H(Y) + H(X \,|Y). \tag{4}$$

The information found shared by two random variables is important in our work and it is defined as the mutual information between two variables:

$$I(X; Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \frac{\log p(x, y)}{p(x) p(y)}. \tag{5}$$

If the mutual information is large, the two variables are closely related. If the mutual information becomes zero, the two variables are independent.

### 2.2 *Filter criterion based on mutual information*

Mutual information based filter method uses an optimization criterion J which is a measure of relevance of a feature with the class to predict independently among the features. All the filter methods produce a ranking of features based on score or relevance function, (Duch 2006). The method for obtaining score can be simply a correlation between the features and class to predict. The simplest criterion stated above only tries to maximize the score between feature and class without taking redundancy among features and is known as 'MIM', mutual information maximization. The criterion does not work well when the features are interdependent. Therefore the as Maximum relevance, minimum redundancy criterion proposed by (Peng *et al* 2005) and Conditional mutual information maximization proposed by (Fleuret 2004) is used in our method for driving ensemble. The objective function using as Maximum relevance, minimum redundancy method is

$$J = \frac{1}{n}\sum\nolimits_{i=1}^{n} I(x_i;\ y) - \frac{1}{n2}\sum\nolimits_{i=1}^{n} I(x_i;\ x_j), \tag{6}$$

where $x_i$ is the input, and $y$ is the output and n is the number of features. The objective function using CMIM is

$$J(x_k) = \min_{xj \in s}\left[I(x_k; (Y \,|x_j))\right] \tag{7}$$

where $S$ represent the set of currently selected features.

## 3. Models of classifier combination

### 3.1 *The classifier model used*

Multi Layer Perceptron is used as classifier. The architecture of Multi Layer Perceptron (MLP) consists of input layer, output layer and hidden layer. Single hidden layer Perceptron gives universal approximation in many pattern recognition applications. The output vector for a single layer Perceptron is given by

$$f(x) = G\left(b^{(2)} + W^{(2)}\left(s\left(b^{(1)} + W^{(1)}x\right)\right)\right), \tag{8}$$

where $b^{(1)}$, $b^{(2)}$ are the bias vectors at the hidden and output layers, $W^{(1)}$, $W^{(2)}$ are the weight matrices at the respective nodes and s, G are the activation functions. For a classification problem if $(x^{(\mathrm{i})}, y^{(\mathrm{i})})$ is the training vector, where $x^{(i)} \in \Re^D$, a D-dimensional training vector and $y^{(i)} \in \{1, \ldots, L\}$. For the prediction function $f(x)$ given in Eq. 8, the zero-one loss function is given by

$$\ell_{0,1} = \sum\nolimits_{i=0}^{|D|} I_{f(x^{(i)}) \neq y^{(i)}}, \tag{9}$$

where $I$ is the indicator function given by

$$I_x = \begin{cases} 1 & \text{if } x \text{ is true} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$$f(x) = \arg\max_k P(Y = k\,|x, \theta), \tag{11}$$

**Table 1.** Steps in gradient descent algorithm.

| Gradient descent algorithm |
|---|
| while True<br>• Loss=f(parameters)<br>• Find derivative of loss with respect to parameters or compute gradient<br>• Modify parameters by-learning rate * derivative of loss with respect to parameters<br>• if <stopping condition is met>:<br>• return parameters |

where $\theta$ is the set of all parameters of the given model. The objective of the training is to minimize the loss function. But the zero-one loss function is not differentiable therefore negative Log-likelihood of loss function minimization is used as the objective of the training.

$$NLL\,(\theta,\ D) = -\sum_{i=0}^{|D|} P\Big(Y = y^{(i)} \Big| x^{(i)}\ , \theta\Big). \tag{12}$$

Weights are updated using gradient of the error surface defined by loss function. Gradient is estimated from the training data using Gradient descent algorithm (given in table 1).

### 3.2 *The classifier combination model used*

Classifier combination is the process of combining classifiers, at data level, at feature level and at classifier level or at decision level. The combination can be applied for different type of classifiers. There are many reasons for using classifier combinations. Firstly, it improves the overall accuracy. Secondly, it makes the overall classifier more robust. In this study, combination of measurement level output of classifiers is considered. Three multilayer Perceptrons are used as a base classifier, in which the random shuffling of patterns is used to obtain diversity in training samples. The experiments are performed with 10 neurons, 20 neurons and 30 neurons in the hidden layer. The transfer function used in the output layer is 'tansigmoid'. MLP is trained with 'trainelm' based back propagation training algorithm. The combination of classifiers is made using decision combination based approach where the measurement level outputs are combined using various combination rule. After calculating the measurement level output i.e. posterior probabilities $\{p_{ij}(x)$ for $i = 1,\ m;\ j = 1,\ c\}$ for a $m$ classifiers and $c$ classes, fixed combining rules are used for decision combiner, (Kittler *et al* 1998). The confidence $q_j(x)$ for class $j$ is computed by

$$q'_j(x) = \text{Rule}_i(p_{ij}(x)), \tag{13}$$

$$q_j(x) = \frac{q'_j(x)}{\sum_j q'_j(x)}. \tag{14}$$

The following combiners are used as rule in Eq. 13: *majority voting, bayes, decision template, Dempster Shafer*. Majority voting based classifier combination is the simplest method in which final decision is that class for which maximum (greater than N/2) participating classifiers vote. Bayes combination is the method in which an intermediate space made of posterior probability of classifiers train the final classifier. These combination schemes are explained in the forthcoming sub-sections.

3.2a *Majority voting principle*: Majority voting is one of the widely used non-trainable combiner. To compute the final score for a particular class, here we simply count the number of classifiers selecting that particular class. The rule can also be applied to rank level classifier which outputs class labels or class ranks. In fact, it does not use the scores, and just uses the class labels. If only class labels are obtained from base classifiers, majority voting is the optimal rule under following minor assumptions: (1) The number of classifiers are odd and the problem is a binary classification problem, (2) The probability of each classifier for choosing any class is equal for an instance, (3) Base classifiers are independent, (Polikar 2006). Majority voting based classifier combination is the simplest method in which final decision is that class for which maximum (greater than N/2) participating classifier vote, where N is the number of classifiers.

3.2b *Decision templates*: The method based on decision template, (Kuncheva *et al* 2001) firstly creates DT for each class using training data. The decision profile based on the testing data is compared using some distance measure with the decision templates stored of each class. The closest DT defines the class label for unknown pattern.

For MLP based classifier the output is a considered as continuous output defining degree of confidence or the posterior probability estimate for each class. Let us consider the $x$ be the feature vector for input pattern such that $\mathrm{x} \in \Re^n$. For a $C$ class problem the if label is given by $\Omega = \{\omega_1, \omega_2, \omega_3 \ldots \omega_\mathrm{C}\}$ and the decision vector of $L$ classifiers is given by $\mathrm{D}=\{\mathrm{D}_1,\ldots \mathrm{D}_\mathrm{L}\}$. Where each classifier decision$D_i$ shows $C$ degree of support which lie in the interval [0, 1] i.e.,

$D_i$:$\Re^n \rightarrow$ [0, 1] then decision profile is given by

$$\mathrm{DP}(x) = \begin{bmatrix} d_{1,1}(x) \cdots & d_{1,j}(x) \cdots & d_{1,C}(x) \\ d_{i,1}(x) \cdots & d_{i,1}(x) \cdots & d_{i,C}(x) \\ d_{L,1}(x) \cdots & d_{L,1}(x) \cdots & d_{L,C}(x) \end{bmatrix}, \tag{15}$$

where $d_{i,j}(x)$ represents the support degree that the classifier $Di$ gives to $x$being from the class $j$.

The decision template is calculated as a mean of decision profile for each individual class. For $j^{\mathrm{th}}$ class having training data as $Z$

$$DT_j = \frac{1}{N_j} \sum_{\substack{zk \in \omega j \\ zk \in Z}} DP(z_k). \tag{16}$$

For the classification of unknown sample similarity between DP(x) and DT for each class is calculated and the closest is assigned as the class label.

3.2c *Dempster-Shafer rule based classifier combination*: Dempster–Shafer (DS) method is based on the evidence theory, proposed by Glen Shafer as a way to represent cognitive knowledge. Here the probability is obtained using belief function instead of using the Bayesian distribution. Probability values are assigned to a set of possibilities instead of unique events. Its appeal is in the fact that they code evidences rather than propositions. It provides a simple method of combining evidences from different sources (Dempster rule) without any a priori distribution, (Ahmadzadeh *et al* 2000). The method of training is same as decision template i.e., DT is calculated using the training data. The method is different from the DT based combination in the way that instead of calculating the similarity between the DP and DT here we calculate the closeness of each pattern

classifier output with the decision template in order to obtain the belief degree for each classifier to each of the respective classes.

Let $DT_i^j$ be the $i$th row of decision template $DT_j$ and $D_i(x)$ be the output of classifier such that $D_i(x) = \left[d_{i,1}(x) \ldots d_{i,c}(x)\right]^2$ the ith row of decision profile $DP(x)$.The proximity $\phi$ between $DT_i^j$ and $D_i(x)$ for input pattern x is calculated by

$$\phi_{j,i}(x) = \frac{\left(1 - \left\| DT_j^i - D_i(x) \right\|^2\right)^{-2}}{\sum_{k-1}^{k-c} \left(1 + \left\| DT_k^i - D_i(x) \right\|^2\right)^{-2}}, \tag{17}$$

where $\|.\|$ is the matrix norm. So for each decision template we have L proximities. Using the last equation we can calculate for every class $j$ =1,. . .c and for every classifier $i$ = 1,. . .L following belief degrees,

$$b_j(D_i(x)) = \frac{\phi_{j,i}(x) \prod_{k=1} \left(1 - \phi_{k,i}(x)\right)}{\left(1 - \phi_{j,i}(x)\right)\left[1 - \prod_{k=1} \left(1 - \phi_{k,i}(x)\right)\right]}. \tag{18}$$

The final support degree is given by

$$\mu_j(x) = \prod_{i=1}^{L} b_{j\ D_i\ (X)}. \tag{19}$$

## 4. Experimental comparison

For experimental evaluation of the proposed method we obtained the dataset of Devanagari numerals from Indian Statistical Institute Kolkata, (Bhattacharya & Choudhary 2009), Intelligent system group Noida, (Kumar *et al* 2013) and from the research performed by (Dongre & Mankar 2012). The information about these dataset are given is the following subsections.

### 4.1 *CVPR-ISI dataset*

This dataset is available to the global research community since 2009 and is developed by Computer Vision & Pattern Recognition unit of Indian Statistical Institute Kolkata. The Devanagari numeral database includes samples collected from mail pieces and job application forms through specially designed form for data collection. The dataset consists of 22,556 images stored in 'tif format collected from 1,049 writers.

A few samples from CVPR ISI numeral dataset is shown in table 2. The maximum efficiency of previous reported result obtained using 64 dimensional feature vector is 96.68% in single stage while 99.04 % accuracy for multistage classifier, (Bhattacharya & Choudhary 2009). Their validation set is obtained by randomly selecting 2,000 characters while training set was made up of 16,794 images and test set was made up of 3,762 images from the whole set.

### 4.2 *CPAR -2012 dataset (Centre for pattern analysis and recognition)*

This dataset is available since the year 2012 to the research community and is developed by Intelligent system group Noida. This is the largest dataset available for the handwritten numerals

**Table 2.** Sample images from CVPR dataset.

| Script digit | Devanagari numeral images from ISI dataset (Bhattacharya & Choudhary 2009) |
|---|---|
| 0 |  |
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| 8 | |
| 9 | |

consisting of 35,000 images. The data is collected from diverse population strata of 2,000 writers from various states of India having different religions. Table 3 gives the detailed number of samples in each of the 11 classes of numeral dataset. There are two ways of writing digit '9' in Hindi, therefore the numbers of classes in this dataset are eleven, this is not the case with the other two datasets.

The third dataset used for the experiment is developed by (Dongre & Mankar 2012), available since the year 2012. It consists of 5,137 symbols of numerals stored in 'tiff' format.

The flow diagram of the proposed approach is shown in figure 2. Recognition results for the three datasets are obtained for the combination of classifier. For each classifier directional histogram features is obtained by dividing each image pattern into nine zones. Ranked list of features is generated by two feature selection algorithms: one based on maximum relevance minimum redundancy and the other based on the mutual information based conditional likelihood maximization. Selection of features is done using class specific method for which the feature vector is generated by converting an n-class problem into n binary class problems. For each binary problem a set of features corresponds to positive class of that binary problem. The resultant output of all such n binary problems is combined and used for training.

**Table 3.** Number of samples in each class of CPAR-12 dataset.

| **Image** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9**[1] | **9**[2] |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Train | 2,280 | 2,280 | 2,280 | 2,280 | 2,280 | 2,280 | 2,280 | 2,280 | 2,280 | 2,280 | 1,200 |
| Test | 1,012 | 1,012 | 1,012 | 1,012 | 1,012 | 1,012 | 1,012 | 1,012 | 1,012 | 1,012 | 880 |
| Total | 3,292 | 3,292 | 3,292 | 3,292 | 3,292 | 3,292 | 3,292 | 3,292 | 3,292 | 3,292 | 2,080 |

[1]indicates the first representation of writing '9'
[2]indicates the second representation of writing '9'


**Figure 2.** Proposed method.

Experiments are performed on three handwritten numeral dataset mentioned earlier. The samples from each of the dataset are chosen randomly and are equal to 200 images per class. The dataset is divided into training and test set using 5-fold cross validation technique. The training set as well as the test set are undergone various preprocessing steps namely binarization, size normalization, filtration, and boundary extraction. The direction based edge features are extracted by partitioning the bounding box of image into nine part of equal size and calculating the histogram in each part by quantizing it into 8-directions.

### 4.3 *Feature set generation*

Gradient based feature sets are generated by applying 'sobel' edge detection algorithm on each pixel of the image. The gradient vector is then quantized into eight directions by vector decomposition using parallelogram rule. In this method the gradient vector is decomposed into two nearest directional planes using parallelogram vector division rule. The parallelogram quantization method gives less quantization error so we have taken this method for quantizing gradient vector.

The calculated gradient of the image is decomposed into four, eight or sixteen directional planes. For our analysis we have taken eight directional plane. We have accumulated the magnitude of gradient in eight discrete directions for each of the subsection of original image. The components of gradient vector are given by the following equation.

$$\begin{aligned} gx(x, y) &= f(x+1,\ y-1) + 2f(x+1, y) + f(x+1, y+1) - f(x-1, y-1) \\ &\quad -2f(x-1, y) - f(x-1, y+1), \\ gy(x, y) &= f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1) - f(x-1, y-1) \\ &\quad -2f(x, y-1) - f(x+1, y-1). \end{aligned} \tag{20}$$

The MLP based classifier is used for experiments in this study. It is having 30 nodes in hidden layer and 'tansig' is used as activation function in the output layer. The three different MLP classifiers having same number of hidden nodes and activation function are combined to form the committee of classifiers. The type of samples are different for each of these classifiers. Moreover, for each of the classifiers the samples are generated by randomly duplicating some of the samples. This is done to achieve the diversity among all the three classifiers. The experiments are conducted for the dataset of each of three classifier using features generated by feature selection algorithm. Feature selection algorithm in this study is defined as "class dependent mutual information based" method. In this method features are selected in a class dependent manner by converting 10-class problem into a ten 2-class problems. Performance comparison is done for two methods of feature selection namely "minimum redundancy maximum relevance" and *conditional mutual information maximization*. The results of the three classifiers are combined using algorithm developed by (Bagheri *et al* 2013). The methods used in the combinations are: (i) Majority voting, (ii) Bayes combination, (iii) Decision template and (iv) Dempster Shafer. The performance in terms of recognition efficiency is presented in tables 4–6 for all the three datasets. The first column is indicating the number of features given to the optimization algorithm for selecting features according to the criterion given by Eq. 6 and 7. Table 4 represents the result for the CPAR- 2012 numeral dataset. The performance is given in terms of recognition rate which is defined as the ratio of correctly classified test pattern to the total number of

**Table 4.** Recognition performance for dataset CPAR-2012 numeral.

| Combination method / Number of features | MV | Bayes | DP | Dempster Shafer | MV | Bayes | DP | Dempster Shafer |
|---|---|---|---|---|---|---|---|---|
| | **Method of FS – CMIM** | | | | **Method of FS – MRMR** | | | |
| 70 | 0.9927 | 0.981 | 0.9947 | **0.995** | 0.9945 | 0.9832 | **0.9952** | **0.9957** |
| 68 | 0.9927 | 0.9807 | 0.9923 | 0.9927 | 0.9933 | 0.983 | 0.9945 | 0.9947 |
| 64 | 0.9937 | 0.9802 | 0.995 | **0.9955** | 0.9927 | 0.9788 | 0.9942 | 0.9942 |
| 54 | 0.9915 | 0.9827 | 0.9922 | 0.9925 | 0.9918 | 0.9807 | 0.9928 | 0.9932 |
| 48 | 0.9892 | 0.9757 | 0.9893 | 0.9898 | 0.9898 | 0.9753 | 0.9915 | 0.9922 |
| 36 | 0.9853 | 0.9697 | 0.9883 | 0.9875 | 0.984 | 0.9688 | 0.987 | 0.9868 |
| 28 | 0.977 | 0.9593 | 0.9805 | 0.9817 | 0.9773 | 0.9608 | 0.9802 | 0.9802 |
| 20 | 0.97 | 0.9557 | 0.9728 | 0.9728 | 0.9662 | 0.953 | 0.968 | 0.9685 |
| 15 | 0.9487 | 0.9383 | 0.9533 | 0.953 | 0.9477 | 0.9377 | 0.95 | 0.95 |
| 10 | 0.9022 | 0.8995 | 0.9078 | 0.9092 | 0.9018 | 0.898 | 0.9048 | 0.9048 |
| 5 | 0.7502 | 0.7502 | 0.7588 | 0.7595 | 0.7212 | 0.7237 | 0.7277 | 0.7268 |

**Table 5.** Recognition performance for (Dongre & Mankar 2012) dataset.

| Combination method / Number of features | MV | Bayes | DP | Dempster Shafer | MV | Bayes | DP | Dempster Shafer |
|---|---|---|---|---|---|---|---|---|
| | **Method of FS – CMIM** | | | | **Method of FS – MRMR** | | | |
| 70 | 0.9883 | 0.9778 | 0.989 | 0.9893 | 0.9885 | 0.9768 | **0.9918** | **0.9915** |
| 68 | 0.9887 | 0.9795 | 0.9895 | 0.9898 | 0.9885 | 0.9763 | 0.9893 | 0.9897 |
| 64 | 0.988 | 0.9782 | 0.9892 | 0.9893 | 0.9863 | 0.9762 | 0.9887 | 0.989 |
| 54 | 0.9837 | 0.971 | 0.9848 | 0.9847 | 0.9833 | 0.9702 | 0.9867 | 0.9867 |
| 48 | 0.9837 | 0.97 | 0.9852 | 0.9848 | 0.9823 | 0.9695 | 0.9813 | 0.9818 |
| 36 | 0.9782 | 0.9672 | 0.9798 | 0.98 | 0.978 | 0.9662 | 0.979 | 0.979 |
| 28 | 0.9712 | 0.956 | 0.9713 | 0.971 | 0.9672 | 0.9567 | 0.969 | 0.9695 |
| 20 | 0.9502 | 0.9437 | 0.9543 | 0.9545 | 0.951 | 0.9425 | 0.9545 | 0.9563 |
| 10 | 0.8832 | 0.882 | 0.8883 | 0.8887 | 0.8722 | 0.8735 | 0.878 | 0.8767 |
| 5 | 0.7358 | 0.7358 | 0.7408 | 0.7415 | 0.7188 | 0.7195 | 0.7195 | 0.7185 |

test patterns. The rates are obtained as a function of number of features selected for driving the ensemble. First four recognition result are giving the performance of combined classifier when the feature selection criterion is CMIM, while the next four recognition results are giving the performance of classifier combination scheme when the feature selection criterion is MRMR. The four combining rules as discussed in section 3.2 are used for classifier combination using measurement level output of the three classifiers. The performance of Dongre's dataset and the CVPR-ISI dataset are given in table 5 and table 6, respectively.

The performance comparison for all the three dataset is given in figure 3 which shows comparison of (1) single classifier, (2) combination of classifier with DS ensemble, (3) DS ensemble driven by CMIM method of feature selection and (4) DS ensemble driven by MRMR method of feature selection. The obtained results show that the best result obtained when DS ensemble is

**Table 6.** Recognition performance of CVPR-ISI dataset.

| Combination method / Number of features | MV | Bayes | DP | Dempster Shafer | MV | Bayes | DP | Dempster Shafer |
|---|---|---|---|---|---|---|---|---|
| | **Method of FS – CMIM** | | | | **Method of FS – MRMR** | | | |
| 70 | 0.987 | 0.9793 | 0.9923 | 0.9928 | 0.9907 | 0.9788 | **0.9937** | 0.9928 |
| 68 | 0.9844 | 0.975 | 0.991 | 0.9915 | **0.9915** | **0.9793** | 0.9933 | 0.993 |
| 64 | 0.9843 | 0.9773 | 0.9902 | 0.9905 | 0.991 | 0.9742 | 0.9923 | 0.992 |
| 54 | 0.9845 | 0.9782 | 0.9907 | 0.9912 | 0.9873 | 0.9728 | 0.9892 | 0.9898 |
| 48 | 0.9808 | 0.9703 | 0.988 | 0.9873 | 0.9858 | 0.9733 | 0.9878 | 0.988 |
| 36 | 0.9747 | 0.965 | 0.9833 | 0.9833 | 0.9777 | 0.9608 | 0.979 | 0.9792 |
| 28 | 0.9597 | 0.9495 | 0.9722 | 0.972 | 0.971 | 0.9518 | 0.9722 | 0.9727 |
| 20 | 0.9442 | 0.9418 | 0.9582 | 0.959 | 0.9557 | 0.9423 | 0.9578 | 0.9573 |
| 15 | 0.9133 | 0.921 | 0.929 | 0.9293 | 0.924 | 0.918 | 0.9288 | 0.9285 |
| 10 | 0.8581 | 0.8652 | 0.8762 | 0.8763 | 0.8773 | 0.8752 | 0.8858 | 0.8852 |
| 5 | 0.6894 | 0.7035 | 0.7113 | 0.7083 | 0.6932 | 0.6923 | 0.7005 | 0.6977 |

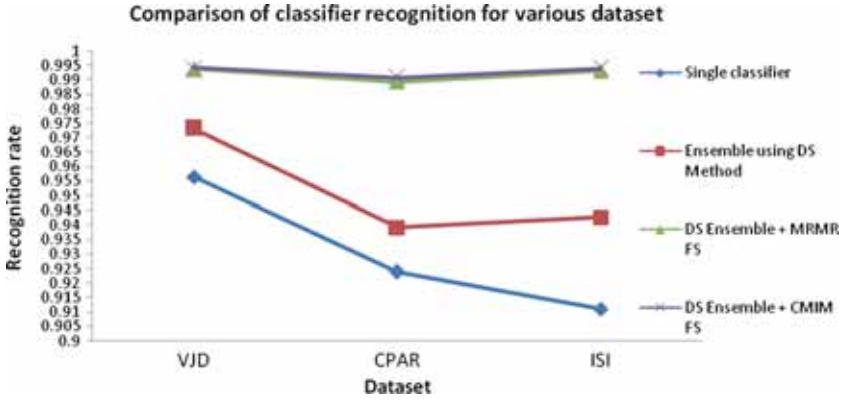


**Figure 3.** Performance comparison of proposed methods over single classifier.

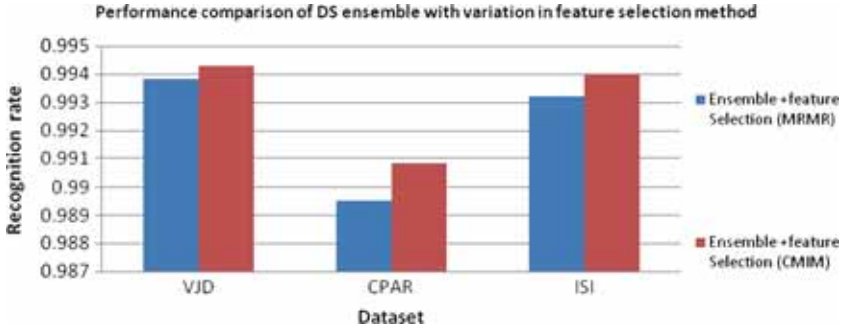


**Figure 4.** Performance comparison of MRMR and CMIM feature selection.

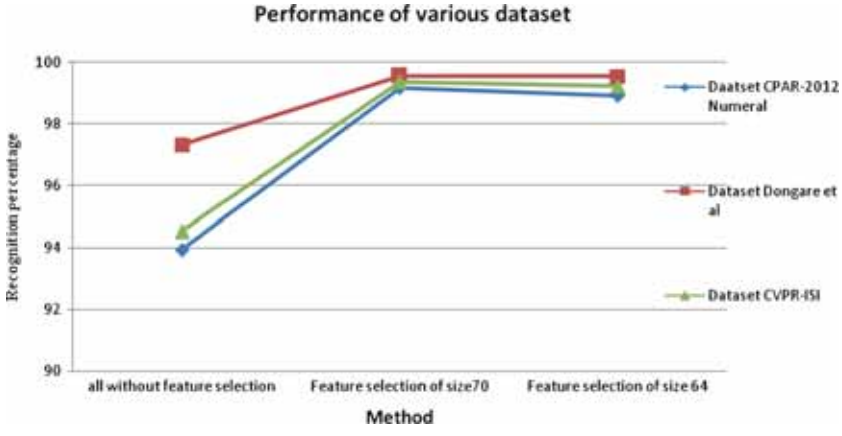


**Figure 5.** Performance comparison for various dataset.

**Table 7.** Performance comparison with previously established benchmarks.

| Dataset | Previously reported result (%) | Our results (%) |
|---|---|---|
| CPAR-2012 Numeral (Kumar *et al* 2013) | 97.87 | 99.57 |
| CVPR- ISI Numeral (Bhattacharya & Choudhary 2009) | 99.04 | 99.37 |
| Dataset (Dongre & Mankar 2012) | 72.87 | 99.18 |

used with the combination of CMIM based feature selection algorithm. The performance comparison for the CMIM based feature selection is compared with MRMR based feature selection for DS ensemble as shown in figure 4. The performance of combiner with and without feature selection for two different feature length is shown in figure 5. The obtained results verifying that feature selection gives better performance. A comparison of obtained result with the previously reported results by other researchers is tabulated in table 7.

## 5. Conclusion

The proposed framework is quite effective in reducing the error rate for the recognition of handwritten samples of Devanagari dataset. By using the ranking generated by class specific feature selection method based on mutual information the improvement in the recognition efficiency is observed. The recognition efficiency is improved for the proposed ranking based on mutual information by 4–5% for CPAR-12 when compared with without feature selection. For CVPR Numeral Dataset the improvement of recognition efficiency with and without feature selection is also around 4%. For dataset of (Dongre & Mankar 2012) efficiency improved by 2%. However, the method of feature selection is quite computationally complex and therefore experimented for small number of classes with reduced sample size. The proposed approach is effective for Devanagari character recognition because for the Devanagari script only a few benchmarking dataset and other resources are available. This method is very effective for dataset of less number of samples.

## References

Ahmadzadeh M, Petron M and Sasikala K 2000 The Depster-Shafer combination rule as a tool to classifier combination. *Geoscience and Remote Sesing, IEEE International Symposium. Proc. IGARSS*, (2429–2431)

Bagheri M, Montazar G and Kabir E 2013 A subspace approach to error-correcting output coding. *Pattern Recog. Lett.* 34: 176–184

Bajaj R, Dey L and Chaudhary S 2002 Devnagari numeral recognition by combining decision of multiple connectionist classifiers. *Sadhana* 27: 59–72

Bhattacharya U and Choudhary B B 2009 Handwritten numeral databases of Indian scripts and multistage recognition of mixed numerals. *IEEE Trans. Pattern Anal. Mach. Intell.* 31(3): 444–457

Cordella L, De Stefano C, Fontanella F and Marrocco C 2008 A feature selection algorithm for handwritten character recognition. *19th International Conference on Pattern Recognition*, (1–4)

Dash M and Liu H 1997 Feature selection for classification. *Intell. Data Anal.* 1: 131–156

Dongre V J and Mankar V H 2012 Development of comprehensive devnagari numeral and character database for offline handwritten character recognition. *Applied Computational Intelligence and Soft Computing* 2012: 1–5

Duch W 2006 Filter Methods, in *Feature Extraction: Foundations and Applications*. Springer-Verlag New York, Inc. Secaucus, NJ, USA: Studies in Fuzziness & Soft Computing, chapter 3, pp. 89–117

Fleuret F 2004 Fast binary feature selection with conditional mutual information. *J. Mach. Learn. Res.* 5: 1531–1555

Hanmandlu M and Murthy O 2007 Fuzzy model based recognition of handwritten numerals. *Pattern Recog.* 40: 1840–1854

Kittler J, Hatef M and Duin R W 1998 On combining classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.* 20(3): 226–239

Kohavi R and John G H 1997 Wrappers for feature subset selection. *Artif. Intell.* 97(1–2): 273–324

Kumar R, Kumar A and Ahmed P 2013 A benchmark dataset for devnagari document recognition research. *Recent advances in telecommunications, signals and systems*, (258–263)

Kuncheva L I, Bezdek J C and Duin R 2001 Decision template for multiple classifier fusion: An experimental comparision. *Pattern Recog.* 34: 299–314

Pal U, Wakabayashi T, Sharma N and Kimura F 2007 Handwritten Numeral Recognition of Six Popular Indian Scripts. *Ninth International Conference on Document Analysis and Recognition*, (749–753) Parana

Peng H, Long F and Ding C 2005 Feature selection based on mutual information: Criteria of maxdependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* 27(8): 1226–1238

Polikar R 2006 Ensemble based systems in decision making. *Circuits Syst. Mag.* 6(3): 21–45

Shannon C and Weaver W 1949 *The mathematical theory of communication*. Urbana, IL: University of Illinois Press

Stefano C D, Fontanella F, Marrocco C and Scotto di Freca A 2014 A GA-based feature selection approach with an application to handwritten character recognition. *Pattern Recog. Lett.* 35: 130–141