Journal of Engineering Science and Technology
Vol. 13, No. 1 (2018) 141 - 157

IMPLEMENTATION OF REJECTION STRATEGIES INSIDE MALAYALAM CHARACTER RECOGNITION SYSTEM BASED ON RANDOM FOURIER FEATURES AND REGULARIZED LEAST SQUARE CLASSIFIER

MANJUSHA K.*, ANAND KUMAR M., SOMAN K. P.

Centre for Computational Engineering and Networking (CEN), Amrita School of Engineering- Coimbatore, Amrita Vishwa Vidyapeetham, Amrita University, India
*Corresponding Author: k_manjusha@cb.amrita.edu

Abstract

Robust and reliable recognition are indeed necessary requirements for optical character recognition systems. Distortions present in the document image and the pre-processing errors cause the optical character recognition system to apply rejection policies to achieve reliable recognition in computer assisted applications. The objective of this paper is to implement a robust and reliable character recognition system for Malayalam language. Random Fourier features classified with Regularized Least Square loss function based Regression classifier can approximate the non-linear kernel machines. Baseline Malayalam character recognition based on Random Fourier features and Regularized Least Square regression classifier is implemented in this paper. Up on this baseline character recognition system, rejection strategies are applied and are experimented with real world document images. An improvement in recognition accuracy is achieved with the simulated Malayalam character recognition system at the cost of rejecting character images having low classification score.

Keywords: Character recognition, Random Fourier features, Regularized least square classifier, Rejection approach, Accuracy - rejection curve.

1. Introduction

Optical Character Recognition (OCR) process can be applied in a wide variety of applications, to speed up data entry or to automate data collection from document images. OCR tries to convert the images of documents, captured through imaging devices to machine editable or machine understandable document format. In case

Nomenclatures

c	Number of character classes
d	Dimension of Random Fourier feature
k	Kernel function
n	Number of input data samples
S	Score Vector assigned by the classifier
W	Weight matrix in RLS classifier
X	Input data matrix in RLS classifier
Y	Label Matrix in RLS Classifier
z	Random Fourier feature vector

Greek Symbols

Φ	Mapping function inside kernel
λ	Regularization parameter in RLS classifier
$\|\|.\|\|_F$	Frobenius norm of matrix
δ	Rejection threshold

Abbreviations

ARC	Accuracy - Rejection Curve
GURLS	Grand Unified Regularized Least Squares
OCR	Optical Character Recognition
RBF	Radial Basis Function
RF	Random Fourier
RLS	Regularized Least Square
SVM	Support Vector Machine

of Malayalam language, the attempts towards building OCR system is less and a complete OCR is still in its progressing stage [1-3]. This paper concerns with the implementation of robust and reliable character recognition system for Malayalam language documents.

Nonlinear kernel machines have much importance in the research area of pattern recognition due to their excellent capability to model highly nonlinear data. Kernel trick avoids the cost of explicit mapping of input data samples to high dimensional feature space and allows classifier to work in implicit feature space of data samples. With the help of kernel trick, kernel machines easily approximate decision boundary between data classes. The dilemma of kernel machines is that, it scales quadratically with the number of training data samples (because of kernel matrix creation and storage). This computational complexity makes kernel machines inadequate to work with large scale classification problems directly. For overcoming this issue, algorithms based on random sampling have been proposed for approximating kernel matrix [4] in large scale classification problems.

Kernel functions can be approximated using Random Fourier (RF) features [5] and can be effectively utilized in classification problems. RF features are capable for approximating shift invariant kernel functions and can be incorporated with linear learning algorithms to achieve the performance level of kernel machines [5-7]. Malayalam character recognition is a large multi-class classification problem. The presence of large number of similarly shaped characters in Malayalam

language creates the need for a robust character recognizer in Malayalam OCR systems [2]. Support Vector Machine (SVM) classifier with extended architectures are evaluated with different feature spaces and are found effective in Malayalam character recognition process [8, 9]. In this paper, the Malayalam character recognition system is built by applying RF features with Regularized Least Square (RLS) regression classifier.

Generally, OCR systems produce good recognition results on good quality document images. In real world scenario, the chances of getting good quality document images are low. Document images may contain distortions introduced due to defects in the paper, defects happened during printing or defects happened during digitization process [10]. Besides that, the errors happened during pre-processing and segmentation stages in OCR system affect the overall recognition accuracy. Due to the above-mentioned distortions and errors, the recognition result obtained from well-trained classifiers (trained with very low error rate), may entirely vary from the expected result.

Reliability in recognition has to be introduced in these circumstances to improve the recognition accuracy. Instead of assigning all segmented character image samples to the highest probable class, the image samples with low classification score (confidence/ probability value of the classifier) have to be identified. In computer assisted applications of OCR, image samples with low classification score can be reported as rejected to improve the reliability in recognition rather than taking the risk of misclassifying it. Figure 1 shows the architecture of rejection based approach on character recognition process.

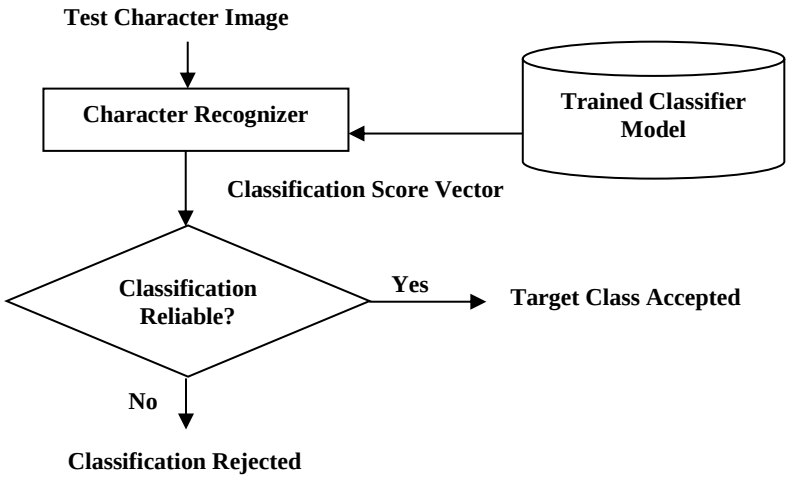


Fig. 1. Flow chart representing rejection based classification approach.

A crucial decision from classifier, which leads to misclassification, occurs mainly due to two reasons. When the data sample is not present in the data set (outlier class or result of segmentation error or noise present in the document), the classifier may not be able to identify the character class and the classifier assigns the data sample a very low classification score. Another reason for misclassification is due to overlapping decision boundary between classes (due to similarity between classes) and the classification score for two or more classes are almost same.

Designing rejection strategies to achieve reliable classification in the above-mentioned circumstances is really difficult. The misclassification rate should decrease monotonically with rejection rate (number of image samples rejected with respect to the total number of image samples) by identifying those critical data samples. Optimal error rate with rejection trade-off for recognition can be calculated and if the conditional probability density of both rate is known [11]. But in almost all real applications, the above-mentioned probability densities are unknown and the rejection strategy is usually derived from the confidence or reliability measure provided by the classifier for the training data samples [12-15].

In this paper, two rejection strategies based on the classification score is implemented to achieve reliable character recognition for Malayalam OCR system developed with RF features on Regularized Least Square (RLS) regression Classifier. The performance of the implemented Malayalam OCR system with the rejection strategies is evaluated on the real-world document images to analyse the effectiveness of proposed approach in reliable classification context.

The base-line Malayalam character recognition system built with RF features and RLS classifier is discussed in Section 2. Up on which the classification rules based on rejection strategies described in Section 3 are evaluated. Different experiments conducted with the base-line recognition system and rejection strategies are described in Section 4. Finally, the conclusion discusses the work mentioned in this paper and outlines for future work.

2. Baseline Malayalam Character Recognition System

Malayalam language belongs to Dravidian family of languages and is the official language of the state Kerala [2]. Malayalam language includes large number of character classes with Vowels (V), Consonants (C), Half-Consonants (known as chillu), Vowel Modifiers and Compound characters. Besides the large number of character classes, script revision happened over time and the existence of non-standard font styles are the main challenges in Malayalam character recognition problem [2]. For implementing a robust character recognition system, in this paper we have used RF features along with RLS classifier.

2.1. Malayalam character image database

For implementing the character recognition system for printed Malayalam language documents, a character image database (Mal_CharDB) is created by using direct (collecting character images from real document images) and synthetic (creating character images by applying various styles and size in text form of Malayalam characters) approaches. Each character image is resized to 32×32. Mal_CharDB consists of 130 different character classes which includes independent and dependent vowels, consonants, some commonly used compound characters of Malayalam language and digits (0-9). Mal_CharDB contains totally 24553-character images. From each character class, 75% of images are taken for implementing training process and the rest is considered for testing.

2.2. Random Fourier (RF) features

Random Fourier features are inspired from the randomization algorithms for approximating kernel functions [5]. Kernel functions define a convenient way for calculating an inner product between the data samples without explicitly lifting the data samples to the higher dimensional space. RF features relies on the fact that the data samples can be mapped in to randomized feature space having lower dimension, so that the inner product between the data samples in randomized feature space approximates positive definite translation invariant kernel functions. Let x and y be the data points and Φ be the mapping for lifting data points to higher dimension, then the kernel function k can be defined as shown in Eq. (1).

$$k(x, y) = \langle \phi(x), \phi(y) \rangle \tag{1}$$

As per Brochner's theorem, the Fourier transform of a shift invariant kernel $k(x-y)$ is a proper probability distribution function if it is properly scaled. Let $p(\omega)$ be the Fourier transform of k, then $k(x,y)$ can be written as in Eq. (2).

$$k(x, y) = k(x - y) = \int_{-\infty}^{+\infty} p(\omega) e^{j\omega^T (x-y)} d\omega \tag{2}$$

As $p(\omega)$ is a probability distribution function, the expected value of $e^{j\omega^T (x-y)}$ is the unbiased estimate of $k(x,y)$ only if ω is drawn from the probability distribution p. The multi-variate vector ω_i, can be generated independently from p. $E(e^{j\omega^T (x-y)})$ can be approximated using the generated ω_i as shown in Eq. (4).

$$\int_{-\infty}^{+\infty} p(\omega) e^{j\omega^T (x-y)} d\omega = E(e^{j\omega^T (x-y)}) \tag{3}$$

$$E(e^{j\omega^T (x-y)}) \approx \frac{1}{d} \sum_{i=1}^{d} e^{j\omega_i^T (x-y)} \tag{4}$$

The RHS of Eq. (4) can be expanded and is equivalent to the inner product in the function space of z (exponentials of projected data samples to ω_i vectors sampled from p) and the resulting inner product approximates the $k(x-y)$.

$$\frac{1}{d} \sum_{i=1}^{d} e^{j\omega_i^T (x-y)} = \frac{1}{d} \sum_{i=1}^{d} \langle e^{j\omega_i^T x}, e^{j\omega_i^T y} \rangle \tag{5}$$

$$= \sum_{i=1}^{d} \langle \frac{1}{\sqrt{d}} e^{j\omega_i^T x}, \frac{1}{\sqrt{d}} e^{j\omega_i^T y} \rangle \tag{6}$$

$$= \left\langle \begin{bmatrix} \frac{1}{\sqrt{d}} e^{j\omega_1^T x} \\ \frac{1}{\sqrt{d}} e^{j\omega_2^T x} \\ \cdots\cdots \\ \frac{1}{\sqrt{d}} e^{j\omega_d^T x} \end{bmatrix}, \begin{bmatrix} \frac{1}{\sqrt{d}} e^{j\omega_1^T y} \\ \frac{1}{\sqrt{d}} e^{j\omega_2^T y} \\ \cdots\cdots \\ \frac{1}{\sqrt{d}} e^{j\omega_d^T y} \end{bmatrix} \right\rangle \tag{7}$$

$$= \left\langle z(x), z(y) \right\rangle \tag{8}$$

$$\approx \left\langle \phi(x), \phi(y) \right\rangle \tag{9}$$

The first vector *z(x),* inside the inner product in Eq. (8) represents lifting of data sample *x* to randomized feature space of dimension *d*. In order to avoid the computation of complex numbers, the data point can be projected on to cosine and sine bases separately and can be appended together to represent (2**d*) dimension vector. So, the resultant *z(x),* can be represented as in Eq. (10).

$$z(x) = \frac{1}{\sqrt{d}} \begin{bmatrix} \cos(\omega_1^T x) \\ ..... \\ \cos(\omega_d^T x) \\ \sin(\omega_1^T x) \\ ..... \\ \sin(\omega_d^T x) \end{bmatrix} \tag{10}$$

2.3. Regularized least-squares regression (RLS) classifier

A simple linear classification algorithm can be used to approximate the performance of non-linear kernel machines by classifying the extracted RF features. Classifier based on Regularized Least Squares loss function can be used for this purpose. Regularized least squares multiclass classification is based on the optimization function which minimizes the average loss in classification [16].

In our multi class classification problem, let *c* be the total number of classes. *X* is the data matrix created by appending all training data samples together. If *d* represents the feature dimension of data samples, then *X* have dimension *n*×*d*, where *n* is the total number of data samples. Let *Y* be the label matrix of corresponding data samples in *X*, having dimension *n*×*c*. In *Y*, each row represents the label vector for data samples with +1 in the position of correct label index and all other entries as -1. The optimization function formulation for RLS classifier is as shown in Eq. (11), where *W* represents the weight matrix with dimension *d*×*c*, which needs to be optimized and λ is the regularization parameter.

$$\min_{W^{d\times c}} \frac{1}{n}\|Y - XW\|_F^2 + \lambda \|W\|_F^2 \tag{11}$$

where, $\|.\|_F$ denotes the *Frobenius* norm of matrix. *n* can be multiplied with λ which is again a scalar, so in further equations, λ represents *(n* λ).*

Let,

$$f(W) = \|Y - XW\|_F^2 + \lambda \|W\|_F^2 \tag{12}$$

$$= Tr\left(\left(Y - XW\right)^T \left(Y - XW\right)\right) + Tr\left(\lambda W^T W\right) \tag{13}$$

$$= Tr\left(Y^T Y\right) + Tr\left(W^T X^T XW\right) - Tr\left(Y^T XW\right) - Tr\left(W^T X^T Y\right) + \lambda Tr\left(W^T W\right) \tag{14}$$

The optimum value for W represented as W^*, can be found by equating the differential of $f(W)$ with zero.

$$\frac{\partial f(W)}{\partial W} = 2\left(X^T X\right)W - X^T Y - X^T Y + 2\lambda W = 0 \tag{15}$$

$$\left(X^T X + \lambda I\right)W = X^T Y \tag{16}$$

$$\Rightarrow W^* = \left(X^T X + \lambda I\right)^{-1} X^T Y \tag{17}$$

Besides Eq. (17), Cholesky factorization can be applied to solve the linear system of Eq. (16) to find W^*. The classification label of test data sample can be found by projecting the data sample x^* (the extracted RF features) to W^* and selecting the label of that class which have the highest projection value. For applying RLS classifier in recognition experiments, GURLS (Grand Unified Regularized Least Squares) package [17] is used, which contains routines for selecting best possible classification model through automatic parameter selection from training data samples.

3. Applying Rejection Approach in Recognition

During training, the RLS classifier is provided with the label matrix Y, in which each row represents the label vector, corresponds to the data sample in X matrix. The label vector is of size $1\times c$, where c is the total number of different classes considered. If the particular data sample belongs to i^{th} class, then the i^{th} entry in the label vector will be +1 and rest of the entries will be -1. During training, the RLS classifier tries to minimize the squared error in prediction by optimizing the weight matrix W and finds W^*. During testing for each test data sample, the RLS classifier predicts a score vector of size $1\times c$. The i^{th} entry in the score vector will be the classification confidence score assigned for the test data sample to belong to i^{th} class. The ideal situation in the multi-class RLS classifier for test data sample is that if the data sample belong to the class i, then the classification score provided for class i, will be 1 and for all other classes except i the score should be -1. But in real cases, the classification score differ from the ideal situation and the score can vary from 1 or -1 and the classification score provided by the RLS classifier will be in the range $(-1-\alpha, 1+\beta)$. Three classification rules are formulated based on the classification confidence score vector S generated by the RLS classifier and are described in section 3.1, 3.2 and 3.3.

3.1. Zero-rejection (Max_Rule)

In order to take the decision about target class of data sample depending on the classification score, the most commonly used one is to assign the data sample to the class with highest score in the classification score vector. This classification rule can be termed further in paper as *Max_Rule*. This rule will assign all the data samples with a target class label without rejection. *Max_Rule* doesn't provide reliability in classification because, even if the score of target class is very low, it still assigns the data sample with that class label. This approach may be intended in those applications where the computer assistance in recognition is not available (recognition verification facility is not available). If S

is the classification score vector $1\times c$ assigned for the data sample, then the target class label assigned for that data sample can be represented as shown in Eq. (18).

$$Max\_Rule(S) = i;\ \text{ if }\ S(i) > S(j) \text{ for } \forall j \neq \text{i},\ j \in [1,c] \tag{18}$$

3.2. Rejection based on score value (SR_Max_Rule)

Max_Rule can be modified such that instead of assigning target classes to all data samples, data samples classified with a maximum score value less than the rejection threshold δ, are rejected. According to *SR_Max_Rule*, classification score vector with its maximum value above δ only termed as reliable classification and all the data samples with classification score vector with its maximum value less than or equal to δ are rejected data samples. *SR_Max_Rule* divides the data samples to two regions accepted, rejected and assigns target class labels to only those data samples which reside in the accepted region and it assigns -1 value for data samples in rejected region. Equation (19) represents *SR_Max_Rule.*

$$SR\_Max\_Rule(S) = \begin{cases} Max\_Rule(S) & ;\text{ if } \max(S) > \delta \\ -1 & ;\text{ if } \max(S) \leq \delta \end{cases} \tag{19}$$

3.3. Rejection based on difference in score (DR_Max_Rule)

Instead of using maximum value in classification score vector, the difference in classification score between the first and second maximum value in classification score vector can be used for evaluating reliability in recognition. Let S_1 represents the highest classification score value inside S and S_2 represents the second highest score value inside S. Then the classification of that data sample is considered as reliable only if the difference between S_1 and S_2 is greater than distance-reject threshold δ_d. If the classes have overlapping decision boundary and in situation where the classifier have to take critical decision in between those classes, then this rejection strategy will reject those data samples instead of misclassifying it.

$$DR\_Max\_Rule(S) = \begin{cases} Max\_Rule(S) & ;\text{ if } (S_1 - S_2) > \delta_d \\ -1 & ;\text{ if } (S_1 - S_2) \leq \delta_d \end{cases} \tag{20}$$

3.4. Proposed rejection based approach

For the classification rules described in section 3.2 and 3.3 rejection threshold has to be estimated from the validation dataset. The rejection threshold estimation is done based on the Accuracy - Rejection curve (ARC). The rejection thresholds are estimated as follows. The algorithm is based on selecting rejection threshold with desirable recognition accuracy. The algorithm can be modified such that rejection threshold can be selected within the desirable rejection rate.

In the above algorithm, 'Correct' and 'Number' represents functions which calculates the number of correctly classified images and total number of images respectively among the dataset passed through parameters. The above algorithm iterates through different rejection thresholds and selects the minimum threshold value which achieve the desirable recognition accuracy among the accepted images.

Rejection Threshold Estimation based on desirable recognition accuracy

Input: 1) Classification score vector *S* of character images in Validation dataset
2) Desirable Accuracy *DAcc*

Algorithm:

1. For each image in validation dataset (*Val*) calculate *FinalScore*.
   In case of *SR_Max_Rule*, *FinalScore* = max(*S*)
   In case of *DR_Max_Rule*, *FinalScore* = (S_1-S_2)
2. Set initial rejection threshold, $\delta = 0$
3. Decide Accepted Images *Accept*, whose *FinalScore* > δ
   Calculate recognition accuracy *Acc*, among accepted images
   Acc = Correct(*Accept*)/Number(*Accept*)×100
4. Decide Rejected Images *Reject*, whose *FinalScore* <= δ
   Calculate rejection rate *RejRate*
   RejRate = Number(*Reject*)/Number(*Val*)×100
5. If *Acc* >= *DAcc*, go to step 6.
   Else increment $\delta = \delta + \Delta\delta$, go to step 3.
6. Stop

In rejection based approaches, the estimation of rejection thresholds is very crucial. The main challenge is that the rejection threshold should be chosen such that all the correctly classified images should be accepted while all the misclassified images should be rejected. Depending on the validation dataset, the rejection threshold estimated may change and thus can affect the overall performance of rejection based recognition systems.

4. Experimental Results and Discussion

Base line character recognition system (Zero rejection) with RF features and RLS multi-class classifier is built on MATLAB environment. 19162-character images from Mal_CharDB are used for training purpose and 5391-character images (validation dataset) are used for evaluating the recognition system. Accuracy of recognition is calculated as the percentage of correctly classified test character images among the total tested images. Likewise, misclassification rate is the percentage of incorrectly classified character images among the total tested images.

4.1. Experiment 1: RF - RLS based character recognition

The first experiment is to find the suitable dimension of RF feature that maximizes the accuracy of recognition system with respect to the Malayalam character image database. RF feature, *z(x)* is extracted from the character images as defined in Eq. 10. The dimension of *z(x)* is determined by the number of random vectors sampled from the Fourier transform of Radial Basis Function (RBF). If the number of random vectors is *d*, then *2*d* will be the size of RF feature vector. RLS classifier model is built on RF features extracted from character images for different values of *d*, and the accuracy of classification is evaluated over test character images. The accuracy of recognition changes with the change in random vectors taken from the probability distribution, so instead of taking single recognition accuracy corresponds to particular *d*, average recognition accuracy is calculated from the 10 trials by changing the random

vectors. Figure 2 shows the variation in average recognition accuracy with the increase in *d*.

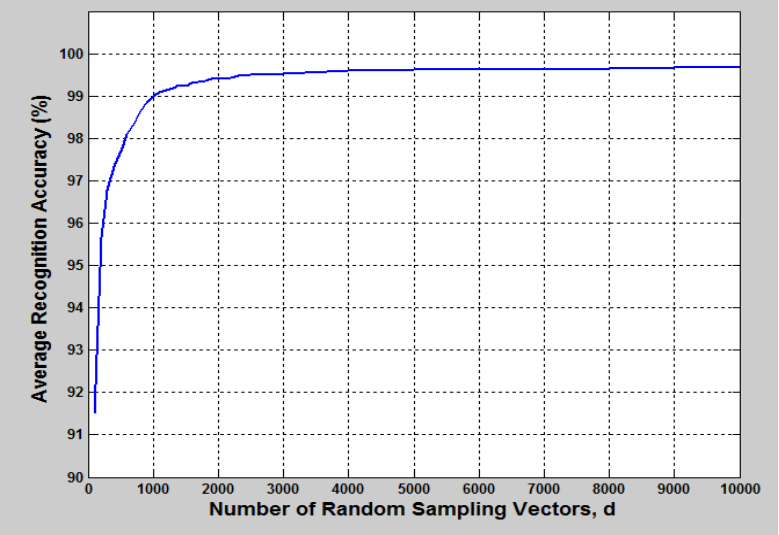


Fig. 2. Average validation accuracy obtained for the Malayalam character recognition system is plotted against the number of random sampling vectors.

The average recognition accuracy of the system among the validation dataset increases exponentially with the increase in dimension *d*. As the value of *d* approaches value 1000, the recognition accuracy near 99% is achieved and after that the increase in recognition accuracy with increase in value of *d*, is at a very slow rate. Table 1 shows the average recognition accuracy achieved for different values of *d* starting from 1000 till 10,000 with an increase of 1000 in *d* value. Even in higher dimensions, the recognition accuracy is still improving at the cost of heavy computation. Till *d*=5000, there is noticeable improvement in accuracy with the increase in feature dimension. But the improvement in accuracy is very low and the increase in accuracy is only 0.03 when the feature dimension is lifted from 5000 to 10,000. Further in our experiments, we are fixing the dimension of *d* as 5000 to avoid heavy computations.

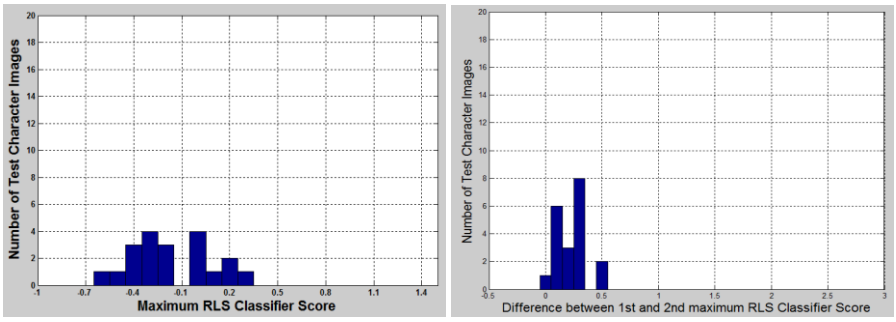
The recognition score corresponds to the target class assigned for the validation images by the recognition system is analysed. Figures 3(a) and (b) shows histogram plot of the highest recognition score and the difference between the first and second high recognition scores respectively in case of misclassified character images in validation dataset. For most of the misclassification cases, the recognition score is concentrated on the lowest region of graph in both the cases. The misclassification happened even in presence of high RLS recognition score is in case of similarly shaped characters. From these histogram graphs it is pretty sure that rejection approaches based on recognition score may clearly detect most of the misclassification happened in the outcome of character recognition system.

For testing purpose, 67 document images collected from various sources are considered. Level-set based active contour method [18] is used for segmenting characters from the document images. Among the segmented 22,712-character

images, 833 images are representing characters that are not present in Mal_CharDB (These are denoted as NDB). 483 images have segmentation error and are denoted as SE. SE and NDB test character images comes under error data samples (ERROR). Image pixel value (IMG) can be used directly as features and are proved feature descriptors in character recognition process [8]. Histogram of Oriented Gradients (HOG) is capable of producing strong feature descriptor in image classification tasks [19]. IMG and HOG features are compared with RF features and the recognition accuracy obtained on test dataset is listed in Table 2. In order to classify the IMG and HOG features, Support Vector Machine (SVM) classifier (linear and RBF kernel) is utilized. RF feature performs better than the other recognizers with 88.08%.

Table 1. Average recognition accuracy of the character recognition system based on RF features for different *d* values on validation dataset

No. of random sampling vectors, *d*	Average recognition\ accuracy, %
1000	99.00
2000	99.43
3000	99.52
4000	99.59
5000	99.63
6000	99.63
7000	99.63
8000	99.65
9000	99.66
10000	99.66



(a) **(b)**

Fig. 3. (a) Highest classification score for misclassified character images in validation dataset, (b) Difference between highest and second highest classification score for misclassified validation character images.

Table 2. Recognition accuracy on test dataset.

Feature	Classifier	Recognition accuracy (%)
IMG	Linear SVM	87.21
IMG	RBF SVM	87.59
HOG	Linear SVM	87.74
HOG	RBF SVM	80.82
RF	RLS	88.08

4.2. Experiment 2: Estimating rejection thresholds

This experiment tries to estimate the optimal rejection threshold value for *SR_Max_Rule* and *DR_Max_Rule* by analysing their effect on the recognition outcome of baseline recognition system. In *SR_Max_Rule*, Test character images are accepted only if the maximum RLS classification score assigned for it is greater than the rejection threshold and recognition accuracy is calculated among the accepted character images. In case of *SR_Max_Rule*, Accuracy - Rejection curve can be plotted for different rejection thresholds based on the recognition outcome on validation dataset and is shown in Fig. 4.

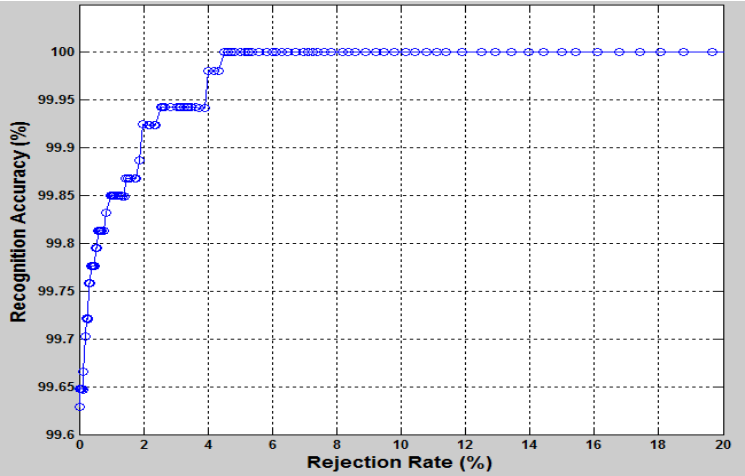


Fig. 4. Accuracy - Rejection curve plotted for *SR_Max_Rule* in recognition outcome obtained from RLS classifier on validation dataset.

Rejection threshold can be selected from this curve either by selecting the desired recognition accuracy among accepted character images or by limiting the rejection rate of the system to a particular value. In Fig. 4., the recognition accuracy is increasing at the cost of increase in rejection rate. The relation between rejection rate and recognition accuracy is monotonic. The recognition accuracy of 100% is achieved with *SR_Max_Rule* by rejecting 4.23% of test character images at a rejection threshold of 0.26. The rejected test character images can be labelled as unreliable and presented to the user for easy error correction.

In *DR_Max_Rule*, the difference between the first and second maximum classification scores assigned for the test character images by the RLS classifier are calculated and based on this difference the character images are rejected. The idea is that if the classifier has clearly distinguished the test character image to belong to a particular class rather than the other, then the classification score for the target class assigned by RLS classifier will be very high compared to classification score of other classes. The recognition accuracy among accepted validation character images and rejection rate according to *DR_Max_Rule* for different rejection thresholds is plotted in Fig. 5. At difference reject threshold 0.48, the system obtained 100% recognition accuracy among accepted test character images by rejecting 1.52% of total test character images. Compared to

SR_Max_Rule, *DR_Max_Rule* obtained 100% recognition accuracy by rejecting very less character images.

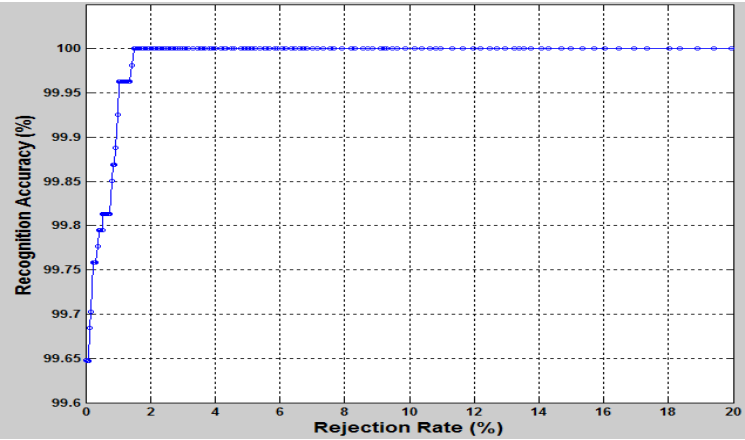


Fig. 5. Accuracy - rejection curve plotted for DR_Max_Rule in recognition outcome obtained from RLS classifier on validation dataset.

4.3. Experiment 3: Applying rejection approach in recognition

The aim of this experiment is to evaluate the *Max_Rule*, *SR_Max_Rule* and *DR_Max_Rule* in real document image recognition. On the test dataset (containing 22712 images) described at the end of Experiment 1, the classification rules are evaluated. ERROR detection rate is calculated as the percentage of ERROR images rejected by the classification rule among all the ERROR images present in the dataset. All 22,712-character images are tested with the same recognition system and the classification scores obtained from the RLS classifier is passed to *Max_Rule*, *SR_Max_Rule* and *DR_Max_Rule*. The rejection threshold estimated from Experiment 2 for *SR_Max_Rule* and *DR_Max_Rule* are used. The recognition accuracy among the accepted reliable classification and among all the tested character images is calculated. The rejection rate acquired for the classification rules along with recognition accuracy are tabulated in Table 3.

The *Max_Rule* could classify the tested character images with 88.08% without rejecting any character image. This is the actual classification accuracy of the implemented character recognition system. *Max_Rule* is not checking the reliability of classification instead assigns target label for all tested character images. Recognition accuracy of *SR_Max_Rule* among accepted test character images is 97.62% and the rule rejected 29.15% of all the tested character images. The same rule could reject 99.09% of the ERROR data samples present in the dataset. The rejection rate of *DR_Max_Rule* is 14.75%, which is only half of that of *SR_Max_Rule* and could achieve 96.03% recognition accuracy among accepted character images.

Among the ERROR character images, *DR_Max_Rule* rejected 90.65% correctly. *SR_Max_Rule* performs better than *DR_Max_Rule* on identifying the ERROR test character images but at the cost of high rejection rate. Combination of *SR_Max_Rule* and *DR_Max_Rule* is also evaluated on the test dataset. A slight

improvement in recognition accuracy is obtained but with slight increase in rejection rate compared to both the rules.

Table 3. Performance of different classification rules in real world document image recognition.

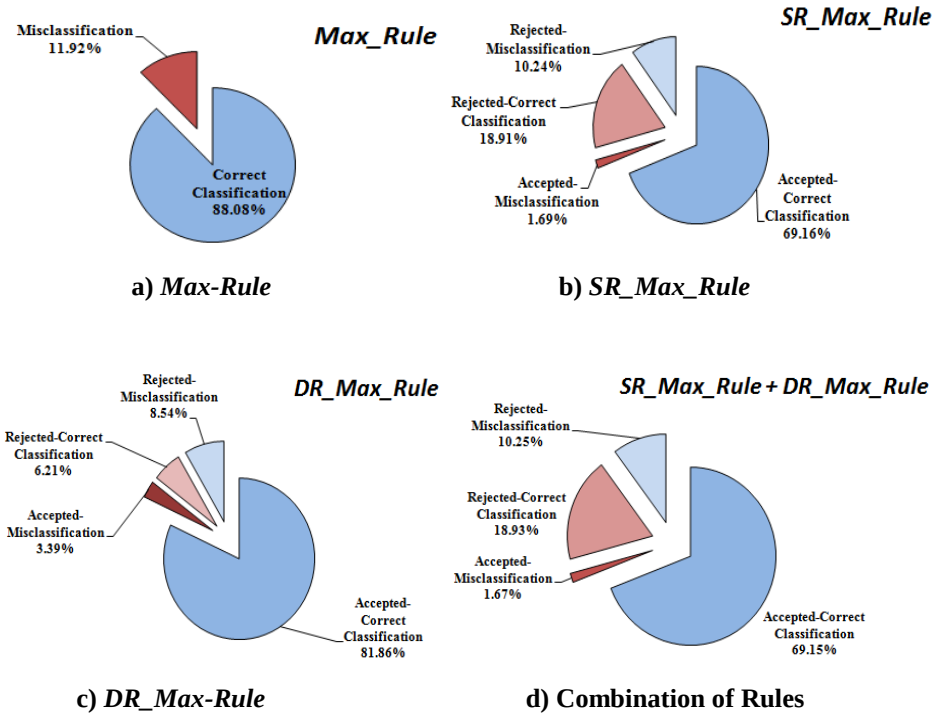
Classification Rule	Rejection rate (%)	Recognition accuracy		ERROR detection rate (%)
		Among accepted images	Among all test images	
Max_Rule	-	88.08	88.08	-
SR_Max_Rule	29.15	97.62	69.17	99.09
DR_Max_Rule	14.75	96.03	81.86	90.65
SR_Max_Rule + DR_Max_Rule	29.18	97.64	69.15	99.09

The rejection rules are not actually improving the recognition accuracy of the baseline system; rather it helps to identify probable misclassifications in recognition outcome. Thus, rejection approaches help in finding those unreliable classifications and opens an opportunity to improve recognition performance through further processing.

The overall performance of the classification rules on the recognition outcome of test dataset is visualized in Fig. 6. With the baseline recognition system, the recognition accuracy obtained without rejecting any character image (with *Max_Rule*) is 88.08%. This implies that the 11.92%-character images were misclassified during recognition. If the rejection rules could detect these misclassifications correctly then there is a chance for improving the accuracy of baseline character system by applying further processing on these rejected character images. Ideally the rejection rules should reject all the misclassification and should accept all correct classifications. *SR_Max_Rule* rejected 29.15% of test character images among that 10.24% were misclassified character images in recognition process. This implies *SR_Max_Rule* could not detect 1.69% misclassified images. *DR_Max_Rule* could detect only 8.54% in 11.92% misclassified character images, which implies 3.38% of misclassifications got accepted with *DR_Max_Rule*. The combination of both rules could detect 10.25% misclassified character images and it reduced the misclassified character images not detected to 1.67%.

Along with the detection of unreliable classification, the other measure used for evaluation of rejection rule is the presence of correct classifications in rejected region. Even though further processing is possible in rejected character images, the presence of correctly classified images in rejected region should be as low as possible. *SR_Max_Rule* can only detect 69.17% in 88.08% correctly classified character images. 18.91% correctly classified character images got rejected through *SR_Max_Rule* whereas *DR_Max_Rule* rejected only 6.21% correctly recognized character images. As further processing can be done on the rejected character images what actually matters is the misclassifications present among the accepted images, so *SR_Max_Rule* is suitable rather than *DR_Max_Rule* even the rejection rate is double than that of *DR_Max_Rule*.

The misclassifications present among the accepted images for the combined rule is mainly happened due to the similarity in shape between the character classes. The error due to similarly shaped classes can be reduced by re-checking applied for those particular classes. The risk involved in rejecting character images is less compared to misclassifying. Further classification, applying language information during post-processing are the possible actions that can be done on rejected character images.



a) *Max-Rule* **b) *SR_Max_Rule***

c) *DR_Max-Rule* **d) Combination of Rules**

Fig. 6. Performance analysis of the classification rules on the recognition outcome of baseline recognition system in real document recognition.

5. Conclusion and Future work

Reliable recognition is one of the necessary requirements in most of the pattern recognition applications. Rejection strategies can be applied on the recognition outcome to identify unreliable classifications. In this paper, we experiment the rejection strategies in Malayalam character recognition system to achieve reliable recognition results. For implementation purpose, an image database (Mal_CharDB) is created with 130 different character classes. Baseline Malayalam recognition system is created by using Random Fourier (RF) features and Regularized Least Square (RLS) multi-class classifier. At RF feature dimension 5000, the baseline recognition system achieved 99.63% recognition accuracy on Mal_CharDB.

Histogram analysis of classification score obtained for the images shows, most of the misclassification occurred in the lower region of classification score.

Two rejection rules are experimented in this paper; first one is based on the highest classification score value (*SR_Max_Rule*) and the other is based on the difference between first and second maximum classification score (*DR_Max_Rule*). The rejection threshold values for the two rules are calculated from the Accuracy - Rejection curve. The effectiveness of rejection rules is evaluated on segmented images extracted from real world document images. *SR_Max_Rule* could achieve 97.62% recognition accuracy among accepted character images by rejecting 29.15% of the character images. 99.09% of the ERROR character images in the real-world test dataset got detected in rejected images. *DR_Max_Rule* have less rejection rate of 14.75% and could detect most of the correctly classified character images. But as the focus of the paper is on detecting the misclassified and ERROR character images through rejection methods, *SR_Max_Rule* is performing better than *DR_Max_Rule*. The combination of both the rules is applied and could achieve slightly better rejected misclassification rate compared to *SR_Max_Rule*. Analysis on misclassification present in accepted character images, explores that these misclassifications occurred mostly due to the high similarity in character shapes. Further classification or applying character context information on rejected character images may improve the recognition accuracy of baseline character recognition system further. Future work includes improving recognition accuracy with the help of multiple classifier decision or language modelling.

References

1. Pal, U.; and Chaudhuri, B.B. (2004). Indian script character recognition: A survey. *Pattern Recognition*, 37(9), 1887-1899.
2. Govindaraju, V.; Setlur, S. editors. (2009). *Guide to OCR for Indic Scripts*. Springer.
3. Krishnan, P.; Sankaran, N.; Singh, A.K. ; and Jawahar, C.V. (2014). Towards a robust OCR system for Indic scripts. *11th IAPR International Workshop on Document Analysis Systems, DAS*, 141-145.
4. Achlioptas, D.; McSherry, F.; and Scholkopf, B. (2002). Sampling techniques for Kernel methods. *Advances in Neural Information Processing Systems*, 335-342.
5. Rahimi, A.; and Recht, B. (2007). Random features for large-scale kernel machines. *Advances in Neural Information Processing Systems*, 1177-1184.
6. Lu, Z.; May, A.; Liu, K.; Garakani, A.B.; Guo, D.; Bellet, A.; Fan, L.; Collins, M.; Kingsbury, B.; Picheny, M.; and Sha, F. (2014). How to scale up kernel methods to be as good as deep neural nets. *ArXiv preprint arXiv:*1411.4000.
7. Rahimi, A.; and Recht, B. (2009). Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. *Advances in Neural Information Processing Systems*, 1(1), 1313-1320.
8. Neeba, N.V.; and Jawahar, C.V. (2009). Empirical evaluation of character classification schemes. *Seventh International Conference on Advances in Pattern Recognition, IEEE Computer Society*, 310-313.

9. Manjusha, K.; AnandKumar, M.; and Soman, K. P. (2015). Experimental analysis on character recognition using singular value decomposition and random projection. *International Journal of Engineering and Technology*, 7(4), 1246-1255.
10. Doermann, D.S.; and Tombre, K. (2014). *Handbook of document image processing and recognition*, Springer.
11. Chow, C.K. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1), 41-46.
12. Cordelia, L.P.; Stefano, C.D.; Tortorella, F.; and Vento, M. (1995). A method for improving classification reliability of multilayer perceptrons. *IEEE Transactions on Neural Networks*, 6(5), 1140-1147.
13. De Stefano, C.; Fontanella, F.; Marcelli, A.; Parziale, A.; and Freca, A.S.D. (2014). Rejecting both segmentation and classification errors in handwritten form processing. *Proceedings of International Conference on Frontiers in Handwriting Recognition, ICFHR*, 569-574.
14. De Stefano, C.; Sansone, C; and Vento, M. (2000). To reject or not to reject: that is the question - an answer in case of neural classifiers. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 30(1), 84-94.
15. Nadeem, M.; Zucker, J.; and Hanczar, B. (2010). Accuracy - rejection curves (arcs) for comparing classification methods with a reject option. *Machine Learning in Systems Biology*, (8), 65-81.
16. Rifkin, R.; Yeo, G.; and Poggio, T. (2003). Regularized least squares classification. *Nato Science Series Sub Series III Computer and Systems Sciences*, 190, 131-154.
17. Tacchetti, A.; Mallapragada, P.K.; Rosasco, L.; and Santoro, M. (2013). Gurls: A least squares library for supervised learning. *Journal of Machine Learning Research*, 14, 3201-3205.
18. Kumar, S.S.; Manjusha, K.; and Soman, K.P. (2014). Novel SVD based character recognition approach for malayalam language script. *Recent Advances in Intelligent Informatics*, 435-442.
19. Dalal, N.; and Triggs, B. (2005). Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1, 886-893.