Epigraphic Document Analysis using Supervised Classifiers

Manjunath B
Dept. of MCA, Maharaja Institute of Technology Mysore
Mandya, India

Sharathkumar Y H
Dept. of ISE, Maharaja Institute of Technology Mysore
Mandya, India

Abstract

The expert epigraphists decipher the text of ancient epigraphic scripts and translate them into regional languages. Also, it is observed that expert epigraphists who are capable of deciphering the inscriptions manually are few nowadays and they could become extinct in future. Modern readers find difficulty in reading the documents of ancient times. Hence, the automation of deciphering of the inscription is the need of the hour and is imperative. From the literature review it is evident that no substantiating work is done for deciphering epigraphical scripts. The aim of the work here is to develop an automated system for classification and recognition of an ancient Kannada epigraphic text, whose period has been identified. Different methods are proposed to decipher text of ancient times with a combination of varying feature extraction techniques and classifiers. The methods are: Zernike Features with SVM Classifier; Central and Zernike Moment features with RF Classifier; Zone-based and Gabor features with ANN; Fourier Features with SVM, k-NN, ANN and Naive Bayes Classifier; SURF features with SVM, ANN and k-NN classifiers. These techniques have been experimented with the test images of different periods and the results obtained are satisfactory. The performance characteristics of the approaches are also discussed.

Keywords: SVM, KNN, ANN, Feature Extraction ;

1. Introduction

Kannada is one of the famous and oldest historical south- ern Indian language, which has its own scripting style derived by brahmic family. It is founded in before 230BC in the form of inscriptions. Inscriptions are the basic and primary form of any language, which are used to gain the historical knowledge, which includes an astronomy, traditions, medication, administration, politics, religion, art, educational and economic conditions and so on. We all know that, the inscriptions are completely different than the existing scripting language, it is highly tedioustask to read and recognize the inscriptions [1]. Hence,it is very much necessary to recognize the inscriptions to gain the historical knowledge. The people, those who can easily recognize the inscriptions are known as epigraphers.But, in this modern era, it is highly impossible to find out the epigraphers for recognition of inscriptions. Even, if wefind out the epigraphers, it is very time consuming task to carry out manual recognition. This is a major drawback of manual recognition system of inscriptions. With the advancement of science and technology, it is very much necessary to develop an automated inscription recognition system to overcome the drawback of manual inscriptions recognition system. This automated inscriptions recognition system, which reads the inscriptions, extract the relevant features, based on the extracted features performs the classification, and finally recognition the inscriptions. The inscriptions has set of ancient characters. As a preliminary task, in this proposed work, an effort has been placed to recognize the ancient characters of BC and AD periods by using deep learning classification and recognition techniques. Some of the challenges in automatic reading of epigraphs are Design an automated epigraphical document recognition system with reasonable recognition accuracy, regardless of the quality of the input document and character font style variation. Segmentation of handwritten text of ancient epigraphs is challenging because of its structural complication, touching lines as well as characters, non-uniform spacing, and the presence of compound characters. The classification of the epigraphical script according to their period is very vital in determining the character set to be applied for supervisory reading. Greatest challenge is the lack of standard/ benchmark data corpus to aid the recognition of ancient Indian scripts.

2. Related work

Combining Zernike Moments with Regional features for Classification of Handwritten ancient Tamil scripts using Extreme Learning Machine is presented in [2]. The Extreme Learning Machine is trained by Zernike moments and Regional features. The performance of Extreme Learning Machine is compared with Probabilistic Neural Networks and inferred that Extreme Learning Machine gives highest accuracy rate of 95%. In [3] the evolution of Brahmi script into Sinhala script on the basis of ancient Sri Lankan documents inscribed on stone surface is discussed. With the aid of modern techniques of computer image processing, precise alphabet fonts of early Brahmi scripts has been produced from photographic data of ancient Sri Lankan inscriptions. It has been shown that the produced fonts are available for establishing a method of automatic reading of ancient inscriptions by computers. An approach for transcribing historical documents in [4] divides a text-line image into frames and a graph is constructed using the framed image. Dijkstra algorithm is applied later to find the line transcription. A character recognition accuracy of 79.3% is found in its experiments. RF Classifier has been used on the Persian language [5] to classify handwritten Persian characters. Loci features are used in the paper. A classification rate of up to 87% has been achieved. A description of the paleographic analysis of Jawi manuscripts is given in [6]. It also gives a comparison of the features, algorithms and results for paleography techniques in different languages. Some ways of computerizing paleography are described in [7]. It uses a sparse document coding for the representation of characters. An accuracy of 93.3% has been claimed in that. It also compares against three other methods which had been done before that.

Characterization of the Arabic and Latin ancient document images is explained in [8]. Regions of images having the same size are extracted from the heterogeneous base and fractal dimension method is used to discriminate between ancient Arabic and Latin scripts. It achieves 95.87% accuracy on the discrimination between Arabic and Latin ancient document collections. A method for the dating of the Greek inscription's content in [9] uses "platonic" realization of alphabet symbols for the specific inscription and various geometric characteristics for the features, and classifies the period according to some statistical criteria. An efficient technique for multi-script identification at connected component level using the convolutional neural network is described in [10]. Suitable script identification features are automatically extracted and learned as convolution kernels from the raw input. It is tested on a dataset of ancient Greek-Latin mix script document images and an accuracy of 96.37% is achieved on a test dataset at the connected component level and improved to 98.40% by using a class majority in the left-right neighboring area [11] Proposes a texture-based approach for text recognition in ancient documents. It copes with the challenges such as degradation, staining, fluctuating text lines, superimposition of text etc. The approach is applied to three different manuscripts, namely to Glagolitic manuscripts of the 11th century, a Latin and a composite Latin-German manuscript, both originating from the 14th century. A method of recognition of ligatures [12] in cursive scripts like Pashto recognizes ligatures having variations like orientation, font style, and scaling. The use of Scale Invariant Feature Transform (SIFT) descriptors is proposed in this to evaluate its effectiveness for representing Pashto ligatures. 1000 unique ligatures with 4 different sizes are tested and average recognition rate of 74% is obtained. [13] presents an approach for the detection of elements like initials, headlines, and text regions, focused on ancient manuscripts. SIFT descriptors are used to detect the regions of interest, and the scale of the interest points is used for localization. It gives a detection rate of 57% for initials and headlines, and 74% for regular text. Work on automated scribe identification on a Middle-English manuscript dataset belonging to the 14th-15th century has been presented in [14]. Identification of the patterns in the image and extracting its features are the finest task as it directly affects the classification process. The authors of the paper Statistical Analysis of the Indus Script Using n-grams discuss the advantage of using statistical feature extraction methodologies in feature extraction process [15]. As per the analysis statistical features provide 75% accuracy in the results. An effective system for the classification of ancient handwritten documents according to the writing style has been reported in [16]. It employs a set of features that are extracted from the contours of the handwritten images. These features are based on the direction and curvature histograms that are extracted at a global level from local contour observations. Two writings are then compared by computing the distance between their respective histograms. An identification rate of 94% is obtained in this. RF Classifier's performance for Handwritten Digit recognition has been accounted in [17]. The Ancient document recognition process consists of two stages: training with collected character image examples and classification of new character images [18]. The proposed OCR builds fuzzy membership functions from oriented features extracted using Gabor filter banks. Results on a significant test led to a character recognition success rate of 88%. The problem of recognizing early Christian Greek manuscripts written in lower case letters [19] is given. Based on the existence of closed cavity regions in the majority of characters and character ligatures in these scripts, a novel, segmentationfree, fast and efficient technique that assists the recognition procedure by tracing and recognizing the most frequently appearing characters or character ligatures is proposed. This method gives highly accurate results and offers great assistance to old Greek handwritten manuscript OCR. The work in [20] on classification and age identification

of different characters by a hybrid model is implemented in two phases. The first phase of the work incorporates an Artificial Neural Network for identifying the base character. The second phase consists of a Probabilistic Neural Network model designed for the identification of age pertaining to the base character. A system to identify and classify Telugu characters extracted from the palm leaves, using Decision Tree approach is brought to light in [21]. The decision tree is developed using the SEE5 algorithm, which is an improvement from the predecessor ID3 and C4.5 algorithm in [22]. The identification accuracy obtained is 93.10% using this method. Much work is reported in the literature on recognition of modern Indian and non-Indian scripts. As seen, literature also reveals substantial work on preprocessing of ancient scripts which include tasks such as noise removal, thinning, binarization and segmentation. It is noticed that work on automated reading of ancient Indian script particularly ancient Kannada script is minimal. Hence, in this research work an attempt is made in automatic recognition of ancient Kannada epigraphical scripts. The Zone based Normalized Positional Distance Metric algo- rithm proposed for recognition of Stone Inscription Char- acters in Ref. [23]. Mean Standard deviation and Sum of Absolute difference Algorithm (MSDDA) has been reported in Ref. [24] for recognition of Historical characters in Kannada stone inscriptions. Here, the Hoysala and Ganga periods characters is used as dataset. In Ref. [25] the authors used the different Image processing tech- niques for recognition and analysis of Historical Tamil stone inscriptions. Hidden Markov Models for recognition of Greek Historical degraded texts has been reported in Ref. [26], and comparison has been done between the Hid- den Markov Models and commercial OCR engines with challenging dataset from a novel database for Greek poly- tonic scripts. In Ref. [27] authors presented different fea- ture extraction techniques: direction histogram and bag of histogram for recognition of Thai handwritten charac- ter. In Ref. [28] authors proposed a system for charac- ter recognition, which is after palm manuscripts of Tamil ancient documents, Here, the authors has taken Brahmi, and Vattezhuthu characters as database. In Ref. [29] the authors used OCR, NLP, SVM and Unicode mapping techniques for recognition and classification Tamil ancientcharacters between 9th and 12th century. The Markov Model has been reported in Ref. [30] for recognition of Kannada handwritten character.

3. Proposed Model for Epigraphic Character Recognition

The proposed model for the Epigraphical Character Recognition is shown in Figure 1 and involves the following components:

- **Preprocessing:** The input epigraphic image is preprocessed to remove noise.
- **Segmentation:** The noise-free epigraphs are segmented to obtain sampled characters.
- **Feature Extraction:** Essential features are extracted from the sampled characters and saved in a file during the training phase. During testing, the same set of features is extracted for test characters.
- **Database:** The database here represents the file used to save the extracted features. The feature vectors are used for training the classifier or for the recognition of characters during testing.
- **Classifier:** The Classifier is trained using the features stored in the file during training phase. It is also possible to save the trained Classifier for later use.The trained Classifier is used to classify the test characters during testing phase. The classified ancient characters are mapped to modern form and displayed.

Methodology for Recognition

Algorithm: **RECOGNITION (Epigraph_Image)Input:**
Epigraphical document image
Output: Classified and Recognized characters in modern form

Method: [Training Phase]

Step 1: Preprocess and Binarize the training epigraph images
Step 2: Segment the characters
Step 3: Extract the features of each of the characters
Step 4: Train the classifier using these features

[Testing Phase]
Step 5: Preprocess and Binarize the test epigraph image
Step 6: Segment the characters
Step 7: Extract the features of each of the test characters
Step 8: Classify each test character using the trained classifier
Step 9: Map the classified characters to modern form.

3.1 Methods for Recognition of Epigraphic Records

In the current work, the methods used to decipher text of ancient times with combination of varying feature extraction techniques and classifiers are: Zernike Features with SVM Classifier; Central and Zernike Moment features with RF Classifier; Zone-based and Gabor features with ANN classifier; Fourier Features with SVM, k-NN, ANN and Naive Bayes Classifier; SURF features with SVM, ANN and k-NN classifiers and lastly First-order and Second-order Statistical features with Fuzzy Classifier for recognition of ancient epigraphic documents. A survey of different shape feature extraction techniques is reported in [31]. The different image classification methods, and techniques for improving classification performance is reported in [32,33].

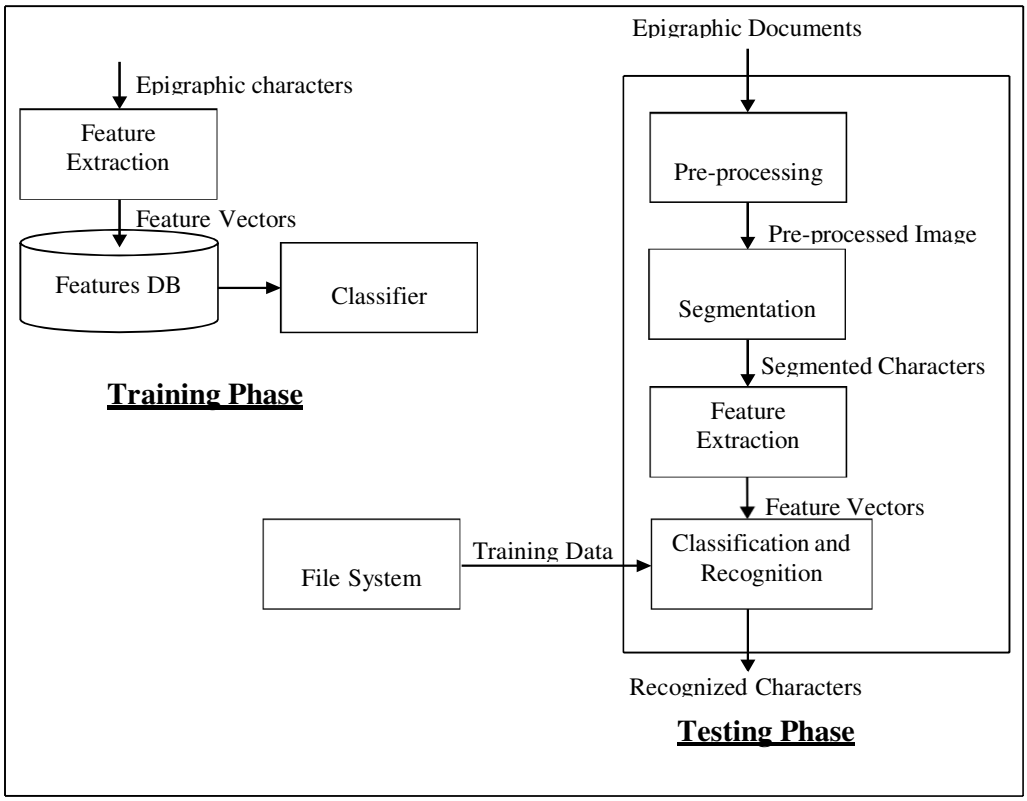


Figure 1: Proposed Model for Epigraphic Character Recognition with Training and Testing phases

3.2 Classification

Support Vector Machine (SVM) Classifier

Support Vector Machines (SVM) is a set of related supervised learning methods used for classification and regression. Viewing input data as two sets of vectors in an n-dimensional space, an SVM will construct a separating hyper- plane in that space, one which maximizes the margin between the two data sets.To calculate the margin, two parallel hyper-planes are constructed, one on each side of the separating hyper-plane, which are "pushed up against" the two datasets. Intuitively, a good separation is achieved by the hyper-plane that has the largest distance to the neighboring data points of both classes, since in general the larger the margin the lower the generalization error of the classifier.

Artificial Neural Network (ANN) Classifier

ANN usually called "neural network" (NN), is a mathematical model or computational model based on biological neural networks. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase. In more practical terms neural networks are non-linear statistical data modeling tools. They can be used tomodel complex relationships between inputs and outputs or to find patterns indata.

k-Nearest Neighbor(k-NN) Classifier

The k-Nearest Neighbor (k-NN) is a method for classifying objects based on closest training examples in the feature space. k-NN is a type of instance-based learning- or lazy learning where the function is only

approximated locally and all computation is deferred until classification. It can also be used for regression.

Naive Bayes Classifier

A Naive Bayes classifier is a term in Bayesian statistics dealing with a simple probabilistic classifier based on applying Bayes' theorem with strong (naive)independence assumptions. Depending on the precise nature of the probability model, Naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for Naive Bayes models uses the method of maximum likelihood; in other words, one can work with the Naive Bayes model without using any Bayesian methods.

Zernike Features with SVM Classifier

The work considers off line recognition of Epigraphical Kannada charactersfrom three ancient eras Ashoka, Badami Chalukya and Mysore Wodeyar. The preprocessed and segmented epigraphic characters from input epigraphicdocument image are fed to the feature extraction phase. The feature extraction method used is Zernike moments (discussed in Chapter 4). The advantages of using Zernike moments are that they are invariant to rotation, robust to noise and minor variations in shape and contain minimum information redundancy [78]. The features extracted for epigraphic characters are fed to the SVM classifier. The SVM classifier classifies the character of ancient age using support vectors and next the character is mapped to present Kannada form.

3.3 Methodology

The steps towards the classification and recognition of epigraphic charactersare given here.

***Algorithm*: Recognition (Epigraphic_Base_Characters)**
Input: Segmented base characters of ancient epigraphs **Output:**
Classified and Recognized characters
Step 1: Perform the following steps during training for segmented characters:

a: Compute Zernike features for base characters of training data

b: Compute the average feature value and store in the data base for later use.

c: SVM classifier is used to produce a model (based on the training data)

Step 2: Compute the Zernike features for each of the base characters during testing.

Step 3: SVM using support vectors from the training database, compares with the test character features and predicts the target values of the test data.

Step 4: Finally the classified characters are recognized, and mapped to the present Kannada form.

3.4 Experimental Results

This approach of Zernike features with SVM classifier demonstrates the recognition of base characters of ancient Kannada Script pertaining to three periods ordynasties – Ashoka, Badami Chalukya, and Mysore Wodeyars. The system is trained with all basic symbols of Ashoka, Badami Chalukya, and Mysore Wodeyars period. The recognizer has been tested on more than 250 samples of ancient Kannada epigraphic characters belonging to three different periods. The character recognition system successfully recognizes the base characters from three different periods and maps it to modern Kannada character. Figure 2 shows the recognition of test letter ‘ka’ from Ashoka period. The input ancient character is classified and recognized, and the character mapped to present Kannada form is displayed. Figures 3 and 4 show the recognition of letter ‘ja’ from Badami Chalukya era and letter ‘ou’ from Mysore Wodeyar era respectively.

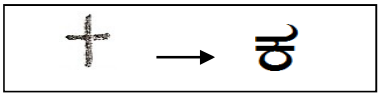


Figure 2: Recognition of letter ‘ka’ of Ashoka era using SVM Model

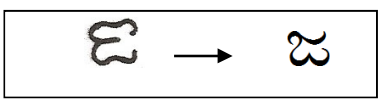


Figure 3: Recognition of character ‘ja’ from Badami Chalukya period using SVM

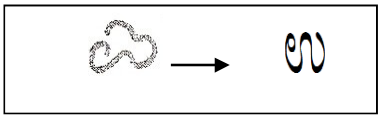


Figure 4: Recognition of character ‘ou’ from Mysore Wodeyar period using SVM

The proposed model recognizes base characters from three different eras Ashoka, Badami Chalukya and Mysore Wodeyar. Characters from these eras are also mapped to present Kannada Character set. Thus, recognition of such ancient characters gives the knowledge of how characters have evolved over generations and transformed to modern form. The show the evolution of few sample characters. The recognizer has been tested on more than 250 samples of ancient Kannada epigraphic characters belonging to three different periods. The model achieves an average 90% recognition accuracy. The recognition accuracy rate of Ashoka era is 93%, Badami Chalukya is 90% and that of Mysore Wodeyar era is 88%.

4. Central and Zernike Moment Features with RF Classifier

The RF Classifier was designed for dating ancient epigraphs as discussed in Chapter 4. This section covers the details of classification and recognition of Kannadaepigraphical characters using the earlier designed RF classifier. Normalized Central Moments and Zernike Moments are extracted from the segmented characters and used as the feature vectors for classification. RandomForest is used as the classifier, which is an ensemble of classification trees, and each tree votes for a class and the output class is the majority of the votes [31, 34]. Thus, all the characters in the image are classified. Finally the classified ancient characters are mapped to characters of modern form.

Methodology

***Algorithm*: RECOGNITION (Epigraph_Image)Input:**
Segmented epigraphic characters.
Output: Classified and Recognized characters.

Method:

Step 1: [Feature Extraction]: The Normalized Central Moments and Normalized Zernike Moments are computed, and the computed feature vectors are written to a file.

Step 2: [Random Forest Classification]

- **a. [Load Text]:** Get the feature vectors from the text file and save it in two arrays, one consisting of the classes and the other consisting of feature vectors of the corresponding classes.
- **b. [Fit Forest]:** Train the trees in the RF which can be used to classify the ancient Kannada characters.
- **c. [Fit Tree]:** A random subset of the training data from the step Fit Forest is taken as input and a single Classification Tree for the given subset of data is made.
- **d. [Get Gini Impurity]:** Determine the impurity index of a subset of classes and corresponding data for the node so that it can find the best split and thebest threshold value for that feature.
- **e. [Classification]:** Predict the class of the test characters considering the data consisting of feature vectors, using the trained RF Classifier.

Step 3: [Recognition]: Map the classified ancient characters into modern form.

4.1 Experimental Results and Performance Analysis

The experimental results and analysis of the designed RF for classifying ancient Kannada Epigraphical characters are discussed here. The system is tested on base characters belonging to Ashoka, Satavahana and Kadamba dynasties. For each dynasty, 105 samples with 35 base characters are considered. Two-thirds of the data isused for training and the remaining one-third is taken for testing the classifier.

Performance Characteristics of RF Classifier

- **Evaluation Metrics**

The metrics used to evaluate the proposed model are:

- **Classification rate:** This metric given by Equation 1 is used to determine the accuracy of the Classifier, which is defined as the number of correct classifications out of the total number of samples considered.

$$\text{Classification rate} = \frac{\text{Number of correctly classified characters}}{\text{Total number of characters in the data}} \quad (1)$$

Training time: This metric measures the time taken to train the Classifier.

- **Classification time:** The classification time is the time taken to predict theclass labels for the given set of inputs.

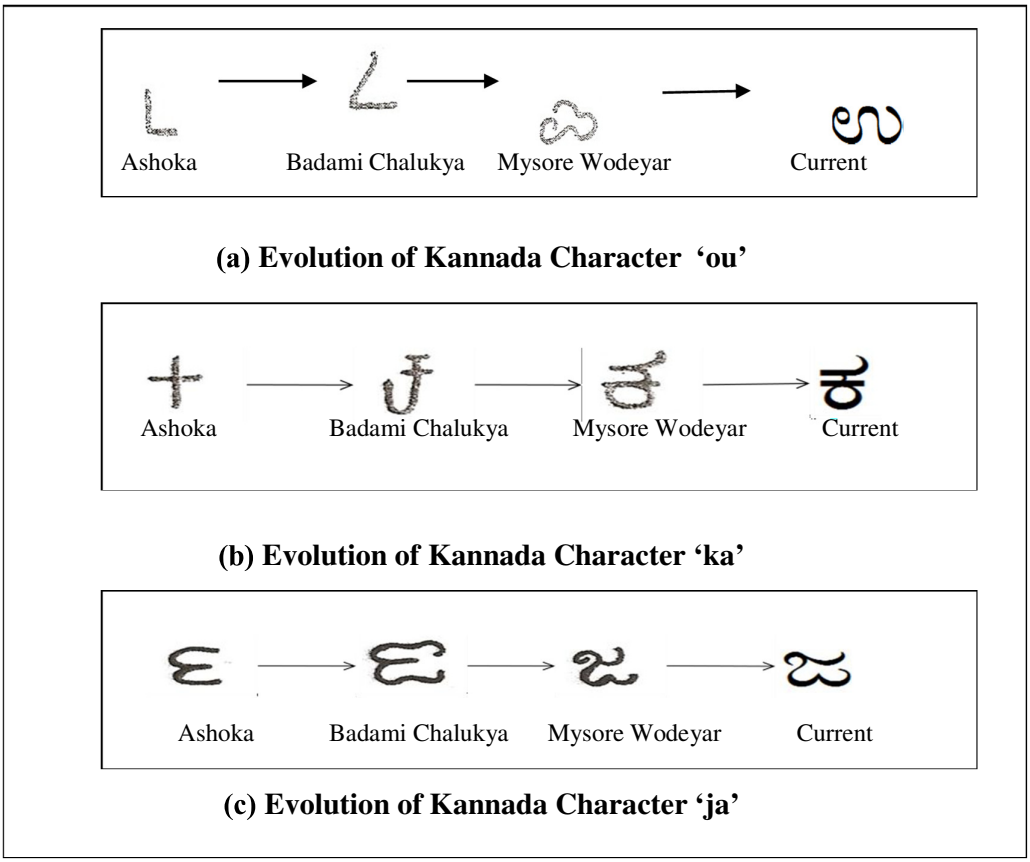


Figure 5: Evolution of Sample Ancient Kannada Characters

- Classification Rate of RF on the Characters from Satavahana Period

The accuracy of RF in classifying characters from trained data set of Satavahanaperiod for the threshold value 10 and a varying number of trees are tabulated in Table 1. The plot in Figure 6 shows the results of the same on trained data.

Table 1: Classification Rates (%) of RF Model for Trained Data

	Number of Trees			
Thresholds		**10**	**20**	**30**
	10	47.69	81.54	90.77
	20	50.77	69.23	84.62
	30	49.23	73.85	87.69

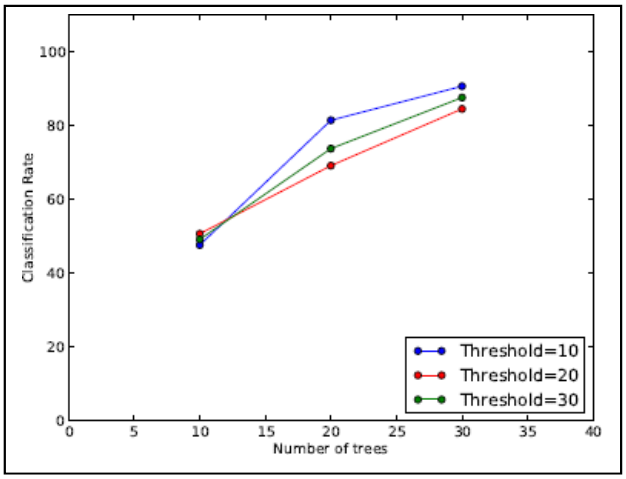


Figure 6: Classification Rates of RF with different Parameters for Trained data

The Classification rate for test characters is tabulated in Table 2 and shownin Figure 7.

Table 2: Classification Rates (%) of RF Model for Test Data

	Number of Trees		
Thresholds	**10**	**20**	**30**
10	43.53	52.35	67.06
20	44.53	52.35	70.00
30	58.25	53.35	61.18

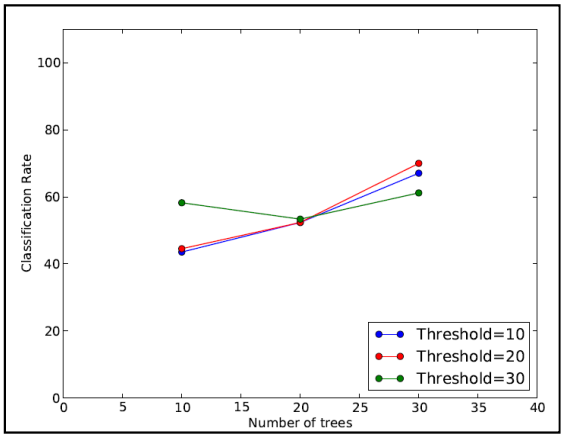


Figure 7: Classification Rates of RF with different Parameters for Test data

- Training and Testing time of RF on the characters from Satavahana period

The time (seconds) for training characters from Satavahana period are tabulated in Table 3 and plotted in Figure 8 respectively. As the number of trees inthe forest increases, the time taken for training also increases proportionately.

Table 3: Training Time (seconds) of Random Forest Model

		Number of Trees		
		10	**20**	**30**
Thresholds	**10**	3.53	7.42	11.62
	20	6.62	11.99	19.07
	30	7.78	15.84	22.64

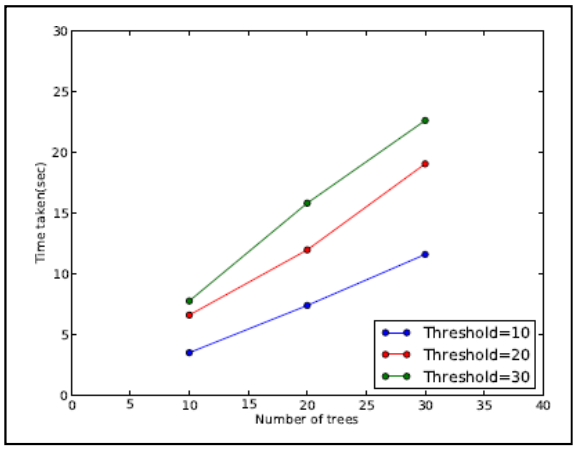


Figure 8: Training Time for RF with different Parameters

The classification times (in seconds) for new characters are tabulated in Table 4 and the plot for the same for different parameters is shown in Figure 9

Table 5.4: Classification Time Taken In Seconds for RF Model

		Number of Trees		
		10	**20**	**30**
Thresholds	**10**	0.0183	0.0361	0.5436
	20	0.0187	0.0367	0.0553
	30	0.0181	0.0365	0.5343

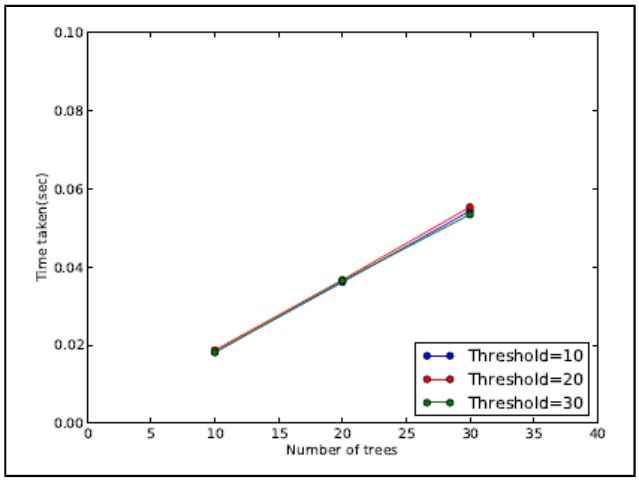


Figure 9: Classification Time for RF with different parameters

The training time taken by the RF Classifier using samples with 35 base characters from Satavahana period were 3.5, 7.4 and 11.6 seconds for RF with 10, 20 and 30 classification trees, respectively, for a threshold of 10. When the number of thresholds was increased to 20, the times taken to train were 6.6, 12 and 19 seconds for RF with 10, 20 and 30 classification trees, respectively. When the number of thresholds was 30, the training times were 7.8, 15.8 and 22.6 seconds for RF with 10, 20 and 30 classification trees, respectively. But the classification rate changed only in the range of 4%-10%. Hence, fixing the number of thresholds at 10 would be a good tradeoff between training time and classification rate.

The following inferences are drawn from the performance analysis:

- The accuracy in classification of the trained data is at least 1.2 times greater than the classification rate of new characters for any classifier.
- There is a linear increase of classification rate as the number of trees in the forest is increased, but no significant changes when the number of thresholdsis increased.
- The training time is directly proportional to the number of classification trees and the number of thresholds.
- The classification time is directly proportional to the number of classification trees. It is not dependant on the number of thresholds since it is used only when growing the trees.
- The training time of the RF classifier is about 200 times more than the classification time. This is because most of the time is spent for the calculationof Gini index during training. Classification involves only a comparison ateach node till it reaches the leaf.

5. Zone-based and Gabor features with ANN Classifier

In this section recognition of epigraphic characters using Zone-based andGabor features with Neural network classifier is discussed.

Methodology

This work includes steps: Pre-Processing, Segmentation, Feature Extraction,Recognition and Post-Processing.

***Algorithm*: RECOGNITION (Epigraph_Image)Input:**
Scanned Epigraphic document.
Output: Classified and Recognized characters.

Method:
Step 1: Input scanned ancient Kannada epigraph to the recognizer.
Step 2: [Preprocess]: Preprocess and **S**egment to extract individual characters.
Step 3: [Feature Extraction]: Extract Zone-based and Gabor features for segmentedcharacters and store in the feature vector.

Step 4: [Training]: Train Artificial Neural Network with feature vectors of thesampled train characters.

Step 5: [Classification]: Classify the segmented characters from test images usingthe trained ANN classifier

Step 6: [Mapping]: The classified ancient characters are mapped to modern form.

5.1 **Related Theory and Mathematical Background**

- **Zone-based feature extraction**

In character recognition, zoning [31] is used to extract topological informationfrom patterns. The segmented image is divided into 'n' zones and from each zone statistical features like number of horizontal/vertical/diagonal lines, length of horizontal/vertical/diagonal lines, the total number of intersection points are extracted.

- **Gabor Feature extraction**

Gabor filters act very similarly to mammalian visual cortical cells so they extract features from different orientation and different scales [35]. The filters have been shown to possess optimal localization properties in both spatial and frequency domain and thus are well suited for texture segmentation problems. Gabor filters havebeen used in many applications, such as texture segmentation, target detection, document analysis and edge detection. In the spatial domain, a 2D Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave. The impulse response of Gabor filter is defined by a sinusoidal wave (a plane wave for 2D Gabor filters) multiplied by a Gaussian function. Because of the multiplication-convolution property (Convolution theorem), the Fourier transform ofa Gabor filter's impulse response is the convolution of the Fourier transform of the harmonic function and the Fourier transform of the Gaussian function. The filter has areal and an imaginary component representing orthogonal directions. The two components may be formed into a complex number or used individually.

Complex

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \exp\left(i\left(2\pi \frac{x'}{\lambda} + \psi\right)\right) \quad (2)$$

Real

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (3)$$

Imaginary

$$(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \sin\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (4)$$

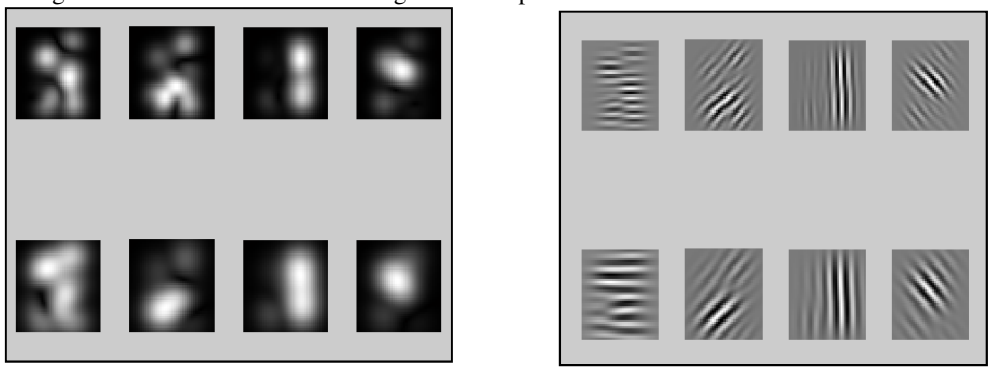
Where

$$x' = x \cos\theta + y \sin\theta$$

and

$$y' = -x \sin\theta + y \cos\theta$$

In these equations, □ represents the wavelength of the sinusoidal factor, □ represents the orientation of the normal to the parallel stripes of a Gabor function, ψ is the phase offset, □ is the standard deviation of the Gaussian envelope and □ is the spatial aspect ratio- and specifies the ellipticity of the support of the Gabor function. Figure 10 shows Gabor Filtered images of a sampled character at two scales and four orientations.



(a) Magnitudes of Gabor Filter (b)Real parts of Gabor Filter

Figure 10: Gabor Filtered Images of a Sampled Character

5.2 Detailed Description and Algorithms

Feature Extraction

The purpose of this phase is to extract the features of the epigraphicalcharacter images.

- Gabor features

This function uses Gabor filter methods to extract features from the characterimage.

***Algorithm*:** Gabor_Feature_Extract (Segmented Character)
Input: Segmented Characters **Output:**
Feature Vector of characters**Method**

Step 1: Compute the orientation
Step 2: Compute the gabor filter bank
Step 3: Convolve it using the conv2 function.
Step 4: Down sample the image by factors of the size of image
Step 5: Store the resultant value in a feature vector

- Zonal features

This function extracts Zone-based features from the character image.

Algorithm: Zone_Based_Features (Segmented Character)

Input: Segmented Characters

Output: Feature Vector of characters

Method

Step 1: Input image is divided into 9 zones of equal size$zone_{ij}$ = image(1:zone_height,1:zone_width);

Step 2: From each zone following features are extracted.

The number of horizontal lines, total length of horizontal lines, number of right diagonal lines, total length of right diagonal lines, number of vertical lines, total length of vertical lines, number of left diagonal lines, total length of left diagonal lines and number of intersection points

Step 3: Extracted features are stored in the new feature vector.

- Mapping

Final mapping of each classified character to the modern Kannada character is done. Class label returned by classifier is used to match with the modern character database and that character is displayed on the screen.

5.3 Experimental Results and Analysis

Figure 11 illustrates the results of Classification and Recognition of inputepigraph of Ashoka period.

- **Performance Analysis**

- Training: Ashokan Brahmi script

The training database of Ashokan Brahmi script contains 8 vowels, 33 consonants which give rise to 264 different compound characters. So there are 272 different characters hence 272 class labels. Four instances of each character are used, so this gives raise to (264+8=272) x 4 = 1088 characters which forms input for training and target vectors indicates the class to which each of the input character belongs.

- Testing: Ashokan Brahmi script

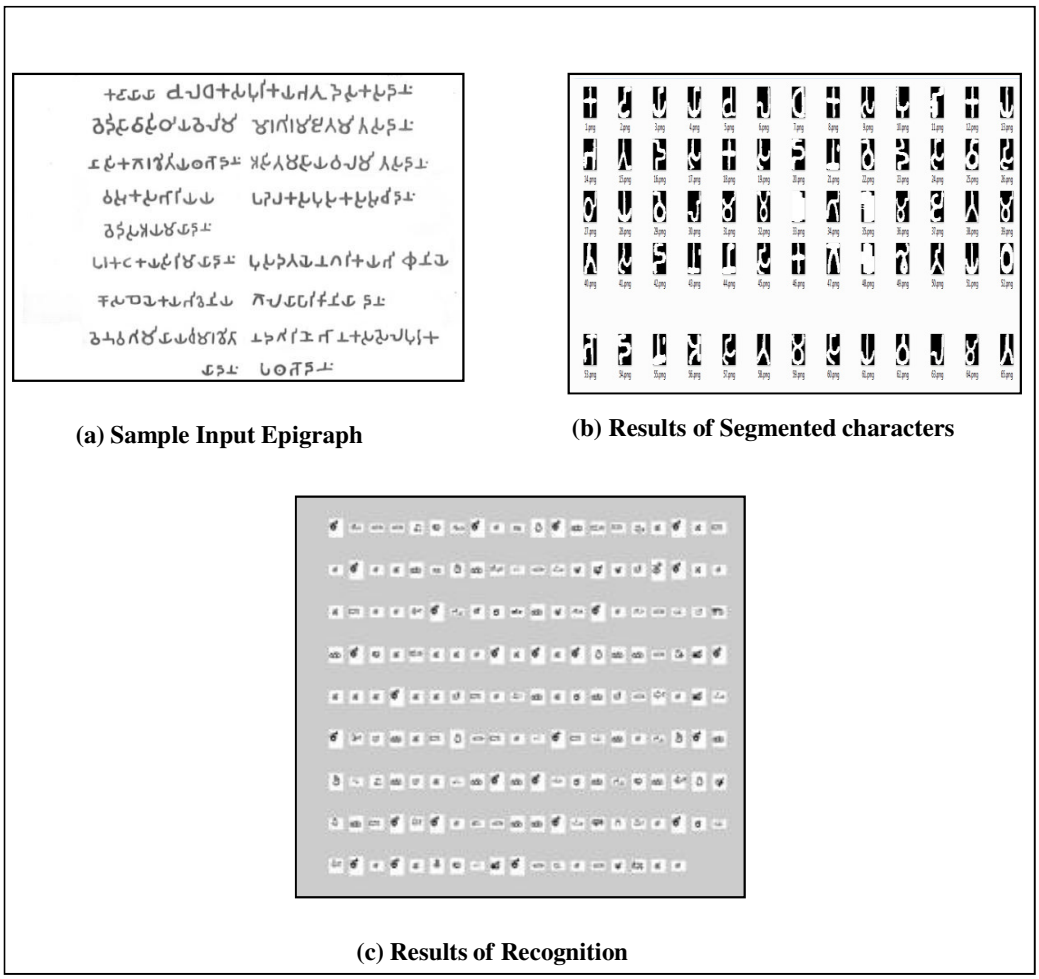
The model is tested with 100 epigraphic images of Ashokan dynasty and obtains an average recognition accuracy of 80.2%.

- Training: Hoysala script

The training database of Hoysala script contains 11 vowels and 36 consonants which give rise to 396 different compound characters. So there are total 407 different characters hence 407 class labels. Four instances of each character were used, so this gives raise to (396+11=407) x 4 = 1628 characters which form input for training and target vectors indicates the class to which each of the input character belongs.

- Testing: Hoysala script

The model is tested with 50 epigraphic images of Hoysala dynasty and obtainsan average recognition accuracy of 75.6%.



(a) Sample Input Epigraph

(b) Results of Segmented characters

(c) Results of Recognition

Figure 11: Classification and Recognition of Epigraph using Gabor Features

6. Fourier Features with SVM, k-NN, ANN and Naive Bayes Classifier

Methodology

The approach takes an image of an epigraph pertaining to ancient Kannada script as its input. The image is preprocessed to remove noise. The Preprocessed image is segmented using Canny edge detection which extracts the edges of ancient script characters. Close character contours are detected from the edges, based on that characters are segmented and stored in the database. General Fourier features are extracted from the segmented characters and based on this the Size and Scale Invariant Fourier features are extracted and used as the feature vectors for classification. Next for classification four classifiers Support Vector Machine (SVM), Artificial Neural Network (ANN), K- Nearest Neighbor (k-NN), Naive Bayes (NB) classifiers [79] are used. These classifiers are trained with different instances of characters during the training phase and while testing categorizes the ancient characters in the test image. Finally, from the predicted class label ancient character is mapped to the modern Kannada character.

6.1 Related Theory and Background

Fourier features a_n, b_n, c_n, and d_n are extracted from close character contours. From these general Fourier features, scale and rotation invariant features are extracted[78].

6.2 General Fourier features

Fourier features can be extracted from close character contours. a_n, b_n, c_n, andd_n are the extracted features and given by Equations 5 to 8 respectively.

$$a_n = \frac{T}{2n^2\pi^2}\sum_{i=1}^{m}\frac{\Delta x_i}{\Delta t_i}[\cos\phi_i - \cos\phi_{i-1}] \quad (5)$$

$$b_n = \frac{T}{2n^2\pi^2}\sum_{i=1}^{m}\frac{\Delta x_i}{\Delta t_i}[\sin\phi_i - \sin\phi_{i-1}] \quad (6)$$

$$c_n = \frac{T}{2n^2\pi^2}\sum_{i=1}^{m}\frac{\Delta y_i}{\Delta t_i}[\cos\phi_i - \cos\phi_{i-1}] \quad (7)$$

$$d_n = \frac{T}{2n^2\pi^2}\sum_{i=1}^{m}\frac{\Delta y_i}{\Delta t_i}[\sin\phi_i - \sin\phi_{i-1}] \quad (8)$$

Where

$\phi_i = {2n\pi t_i}/{T}$, $\Delta x_i = x_i - x_{i-1}$, $\Delta y_i = y_i - y_{i-1}$, $\Delta t_i = \sqrt{\Delta x^2 + \Delta y^2}$

$T = t_m = \sum_{j=1}^{m}\Delta t_j$, $t_i = \sum_{j=1}^{i}\Delta t_j$, and m is the number of pixels along the boundary

Rotation invariant Fourier features

To obtain the features that are independent of the particular starting point, it isrequired to calculate the phase shift from the first major axis as in Equation 9.

$$\phi_1 = \frac{1}{2}\tan^{-1}\frac{2(a_1b_1 + c_1d)}{\sqrt{a_1^2 - b_1^2 + c_1^2 - d_1^2}} \quad (9)$$

Then, the coefficients can be rotated to achieve a zero phase shift as given by Equation 10.

$$\begin{bmatrix} a_n^* b_n^* \\ c_n^* d_n^* \end{bmatrix} = \begin{bmatrix} a_n b_n \\ c_n d_n \end{bmatrix} \begin{bmatrix} \cos n\phi_1 - \sin n\phi_1 \\ \sin n\phi_1 \cos n\phi_1 \end{bmatrix} \quad (10)$$

Now to obtain rotation invariant description, the rotation of the semi-major axis can be found by Equation 11:

$$\psi_1 = \tan^{-1} \frac{c_1^*}{a_1^*} \quad (11)$$

Now using Equation 12 features can be obtained.

$$\begin{bmatrix} a_n^{**} b_n^{**} \\ c_n^{**} d_n^{**} \end{bmatrix} = \begin{bmatrix} \cos\psi_1 & \sin\psi_1 \\ -\sin\psi_1 \cos\psi_1 \end{bmatrix} \begin{bmatrix} a_n^* b_n^* \\ c_n^* d_n^* \end{bmatrix} \quad (12)$$

Scale invariant Fourier features

To obtain Scale invariant features the coefficients can be divided by the magnitude, E, of the semi-major axis, given by Equation 13:

$$E = \sqrt{a_1^{*2} + c_1^{*2}} = a_1^{**} \quad (13)$$

6.3 Detailed Description and Algorithm

The steps involved in classification and recognition of epigraphic characters usingFourier features with SVM, ANN, k-NN, NB classifiers are given in this section.

Algorithm: RECOGNISE (Epigraph)**Input:** Ancient Kannada Epigraph **Output:** Modern Kannada characters

Method:

Step 1: Read the epigraph image.
Step 2: Preprocess epigraph
Step 3: Segment epigraph and store segmented characters.
Step 4: Extract Fourier features for segmented characters.
Step 5: Train Classifiers SVM, ANN, k-NN and NB using these features.
Step 6: Extract features of segmented test character.
Step 7: Classify the segmented character of the ancient period.
Step 8: Classified ancient character is mapped to the modern Kannada form.
Step 9: Return modern Kannada character.

- Fourier Feature Extraction

General Fourier features are extracted from close character contours. a_n, b_n, c_n,and d_n are the extracted features.

***Algorithm*:** Fourier_Features (Character Contour, x, y)

Input: Segmented character image
Output: General Fourier features a_n, b_n, c_n, d_n.

Method:
Step 1: Initialize Parameters **Step 2:**
Compute Fourier features**Step 3:** Store
Fourier features

End Method

- **Classification**
  - SVM Classifier

This model classifies the segmented character and predicts its class label. Itconsists of the following steps:

Step 1: Set up the training data **Step 2**:
Set up the training classes**Step 3:** Set up
SVM's parameters
Kernel Type = LINEAR, SVM Type = C_SVC, Termination Criteria

Step 4: Train the SVM

Step 5: Classification of characters using SVM

- ANN Classifier

This model classifies the segmented character and predicts its class label. Itconsists of the following steps:

Step 1: Set up the training data **Step 2:**
Set up the training class **Step 3:** Set up
ANN's parameters
Parameters of the MLP training algorithm are set.

***term_crit*:** Termination criteria of the training algorithm.

***train_method*:**Indicates the training method of the MLP - back-propagation.

Step 4: Train the ANN
ANN model is built with MLP network and Activation_function SIGMOID

Step 5: Classification of characters using trained ANN

- k-NN Classifier

The k-nearest neighbors (k-NN) is a method for classifying objects based on closest training examples in the feature space. Here it classifies the segmented character and predicts its class label. It consists of the following steps:

Step 1: Set up the training data
Step 2: Set up the training classes
Step 3: Set up k-NN's parameters
Finds the neighbors

samples – Input samples stored by rows.
k – Number of nearest neighbors used.

results –results of prediction (classification) for each input sample.

Neighbor Responses –Optional output values for corresponding neighbors.

Step 4: Train the k-NN
Step 5: Classification of characters by k-NN

- Naive Bayes Classifier

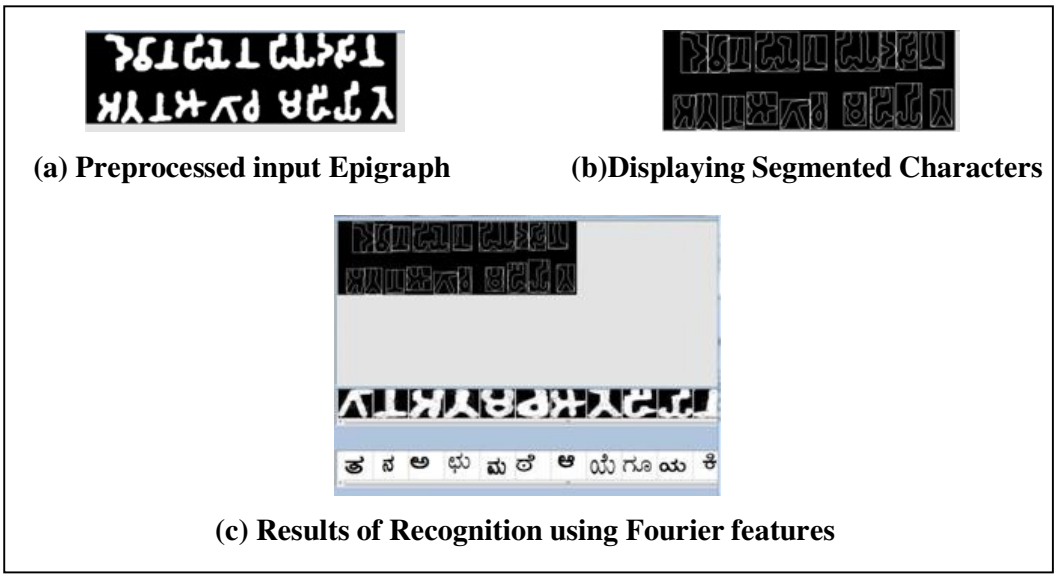
A naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature. This classifies the segmented character and predicts its class label. It consists of the following steps:

Step 1: Set up the training data
Step 2: Set up the training classes
Step 3: Train the Naive Bayes
Step 4: Classification of characters by Naive Bayes

Experimental Results and Analysis

The system designed recognizes and converts Ashokan Brahmi script and Hoysala script into modern Kannada form. The system is trained with characters of Ashoka period and characters of Hoysala period with 4 different instances of each.The trained OCR system is tested on 50 epigraphs of ancient times.

Figure 12 shows the results of Preprocessing, Segmentation and recognition.



(a) Preprocessed input Epigraph **(b)Displaying Segmented Characters**

(c) Results of Recognition using Fourier features

Figure 12: Results of Recognition using Fourier features

The performance characteristics of the Classifiers is obtained and observed that with Ashokan Brahmi script the system performed well with recognition accuracyof 83.60%, 76.80%, 49.20%, 64.80% . For Hoysala script, recognition accuracy of 80.50%, 71.50%, 48.50%, 62.50% with `SVM, ANN, k-NN and NB classifiers respectively is obtained.

7. SURF Features with SVM, ANN and k-NN classifiers

Methodology

The aim of this approach is to use multiple classifiers that recognize ancient Kannada characters and maps them to modern Kannada. The approach accepts ancientKannada epigraph from Ashoka period as input. The input image is binarized by Adaptive thresholding and noise is removed by applying a combination of three filtersnamely Median, Bilateral and Gaussian Filters. Furthermore, ancient text is made prominent by applying erode and dilate morphological operations. The resultant image is passed to the Segmentation stage where Bounding Box and Contour Detection Algorithm are used to segment individual characters including the vattaksharas (compound characters). These segmented characters are then passed to the Feature extraction stage that makes use of SURF technique to create the feature vectors. Classification is the final stage shown in Figure 5.14, which is carried out in two phases namely Training and Testing. In order to train the Classifiers, the feature vectors are passed to the classifiers namely SVM, k-NN and ANN in the training Phase. In the testing phase, the feature vectors are passed to a trained classifier to recognize the ancient character. A combination of three classifiers namely, SVM, k- NN and ANN are used to achieve better accuracy. The recognized character is subsequently mapped to its modern equivalent. In this way, a document in ancient Kannada can be translated to modern Kannada.

7.1 Feature Extraction

SURF or Speeded up Robust Features is a scale- and rotation-invariant interestpoint detector and descriptor [82]. SURF is a detector and high-performance descriptor points of interest in an image where the image is transformed into coordinates, using a technique called multi-resolution. It approximates or even outperforms previously proposed schemes with respect to repeatability, distinctiveness, and robustness, yet can be computed and compared much faster. The mathematical representation is given by Equation 14,

$$H(p,\sigma) = \begin{pmatrix} L_{xx}(p,\sigma) & L_{xy}(p,\sigma) \\ L_{yx}(p,\sigma) & L_{yy}(p,\sigma) \end{pmatrix}$$

where $Lxx(p, \sigma)$ is the convolution of second order derivative $\partial x \partial x 2g(\sigma)$ with the image in the point x, y similarly with $Lxy(p, \sigma)$ and $Lyy(p, \sigma)$.

The SURF algorithm is based on the SIFT predecessor. This is achieved by

- Relying on integral images for image convolutions
- Building on the strengths of the leading existing detectors and descriptors (using a Hessian matrix-based measure for the detector, and a distribution-based descriptor)
- Simplifying these methods to the essential

This leads to a combination of novel detection, description, and matching steps. The detector is based on the Hessian matrix, but uses a very basic approximation, just as DoG is a very basic Laplacian-based detector [82]. It relies on integral images to reduce the computation time and therefore it is called the 'Fast-Hessian' detector. The descriptor, on the other hand, describes a distribution of Haar-wavelet responses within the interest point neighbour hood. The integral images are exploited for speed. Moreover, only 64 dimensions are used, reducing the time for feature computation and matching, and increasing simultaneously the robustness. A new indexing step based on the sign of the Laplacian is presented, which increases not only the matching speed- but also the robustness of the descriptor.

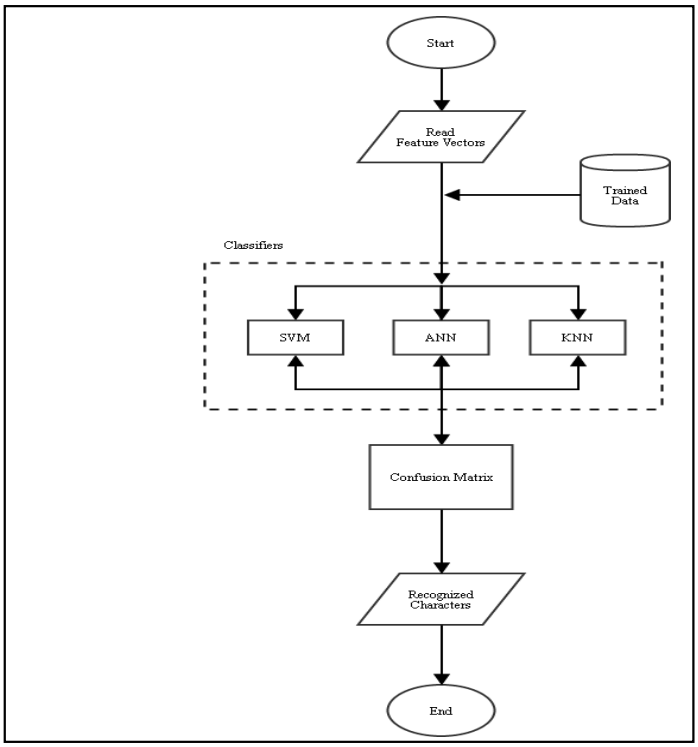


Figure 13: Model for Classification and Recognition using Multiple Classifier

Detailed Description and Algorithm

➢ Feature Extraction

In this stage, for each character segmented image, a hessian threshold iscalculated that leads to detecting key points of the character to form Feature Vectors. Algorithm for SURF Feature Extraction technique is as follows:

Algorithm: SURF_ Feature_Extract (Epigraphic character)
Input: Individual Character Segment

Output: Feature Vectors
Functionality: Extracts the key-points from the image and creates the feature vectors.

Step 1: Input the necessary segmented character

Step 2: Set the Hessian Threshold to 450.

Step 3: Choose the SURF descriptor size to be 64 dimensions.

Step 4: Detect the SURF features.
Step 5: Construct the feature descriptor for the detected features.

Classification and Recognition
The SURF features are extracted for the test characters and classified using theclassifiers- SVM, ANN and k-NN. For the efficient recognition of characters, the output of these multiple classifiers is passed to the Confusion Matrix. The Confusion Matrix evaluates the result of the classifiers and provides the final character label.The classified character is next mapped to modern form.

7.2 Experimental Results and Analysis

This OCR recognizes the ancient scriptures of Ashoka period using SVM, ANN and k-NN. Figure 5.15 represents the input epigraphic image. The Figure 5.16 shows the results of Preprocessing and Segmentation of the input epigraph. The output of recognition is shown in Figure 5.17. The system performs well with better recognition rate when multiple classifiers namely SVM, ANN and k-NN are used along with the confusion matrix in order to resolve the errors which arise during the character recognition. Thus, a combination of classifiers in recognizing characters gives a higher accuracy than using individual classifiers. The classifiers SVM, k-NN and ANN when tested on randomly chosen 90 characters achieve a recognition accuracy of 85%, 85% and 80%, but when the classifiers are combined the recognition accuracy increases to 95% thereby experimentally demonstrating that a combination of classifiers achieves 5-10% higher accuracy. However this may increase in time complexity due to the presence of more than one classifier. Careneeds to be taken to decrease the time complexity and improve the recognition accuracy. Hence, the confusion matrix is introduced to create this win-win situation. Another approach is to execute the classification stage in parallel so as to reduce the time complexity.

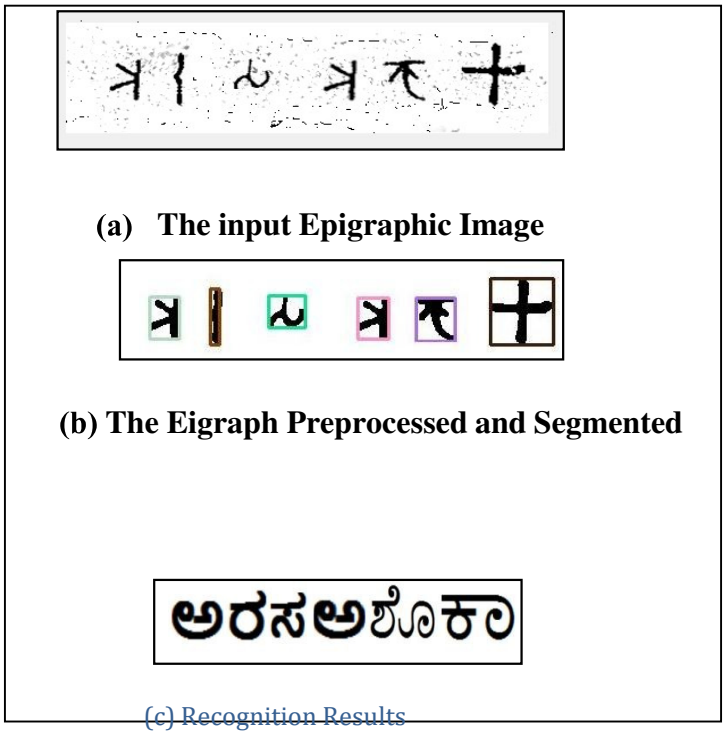


Figure 14: The Results of Recognition using SURF features

8. Summary

This chapter discussed the methods used for recognition of epigraphical characters from different periods. Six methods for feature extraction and recognition of epigraphical characters with a combination of different classifiers are discussed. Based on structural and statistical features with different classifiers, methods are explored for recognition of epigraphical text. The methods discussed are:- ZernikeFeatures with SVM Classifier; Central and Zernike Moment features with RF Classifier; Zone-based and Gabor features with ANN; Fourier Features with SVM, k-NN, ANN and Naive Bayes Classifier; SURF features with SVM, ANN and k-NN classifier; finally Fuzzy

Classifier using First-Order and Second-order Statistical features for the for recognition of epigraphic documents. These techniques have been experimented with the test images of different periods and the results obtained are satisfactory. The performance characteristics of the approaches are also discussed.

References

1. K. Srikanta Murthy, G.Hemantha Kumar, P.Shivakumar, “A Novel method based on rectangle fitting for noise removal in an epigraphical script,” Proceedings of 39th Annual convention of Computer Society of India, Mumbai, pp 166-171, 2004.
2. Sridevi, N., and P. Subashini. "Combining Zernike moments with regional features for classification of handwritten ancient Tamil scripts using Extreme learning machine." In Emerging Trends in Computing, Communication and Nanotechnology (ICE-CCN), 2013 International Conference on, pp. 158-162. IEEE, 2013.
3. Bandara, Dammi, Nalin Warnajith, Atsushi Minato, and Satoru Ozawal. "Creation of precise alphabet fonts of early Brahmi script from photographic data of ancient Sri Lankan inscriptions." Can. J. Artif. Intell. Mach. Learn. Pattern Recognit 3, no. 3 (2012): 33-39.
4. Meza-Lovon, Graciela Lecireth. "A graph-based approach for transcribing ancient documents." In Ibero-American Conference on Artificial Intelligence, pp. 210-220. Springer Berlin Heidelberg, 2012.
5. Zahedi, M., and S. Eslami. "Improvement of Random Forest Classifier through Localization of Persian Handwritten OCR." ACEEE Int. J. Inf. Technol 1, no. 2 (2012): 31-36.
6. Azmi, Mohd Sanusi, Khairuddin Omar, Mohammad Faidzul Nasrudin, Azah Kamilah Muda, and Azizi Abdullah. "Digital paleography: Using the digital representation of Jawi manuscripts to support paleographic analysis." In Pattern Analysis and Intelligent Robotics (ICPAIR), 2011 International Conference on, vol. 1, pp. 71-77. IEEE, 2011.
7. Wolf, Lior, Liza Potikha, Nachum Dershowitz, Roni Shweka, and Yaacov Choueka. "Computerized paleography: tools for historical manuscripts." In 2011 18th IEEE International Conference on Image Processing, pp. 3545- 3548. IEEE, 2011.
8. Zaghden, Nizar, Remy Mullot, and Adel M. Alimi. "Characterization of ancient document images composed by Arabic and Latin scripts." In Innovations in Information Technology (IIT), 2011 International Conference on, pp. 124-127. IEEE, 2011.
9. Papaodysseus, Constantin, Panayiotis Rousopoulos, Dimitris Arabadjis, Fivi Panopoulou, and Michalis Panagopoulos. "Handwriting automatic classification: application to ancient Greek inscriptions." In Autonomous and Intelligent Systems (AIS), 2010 International Conference on, pp. 1-6. IEEE, 2010.
10. Rashid, Sheikh Faisal, Faisal Shafait, and Thomas M. Breuel. "Connected component level multiscript identification from ancient document images." In Proceedings of the 9th IAPR Workshop on Document Analysis System, pp. 1-4. 2010.
11. Garz, Angelika, and Robert Sablatnig. "Multi-scale texture-based text recognition in ancient manuscripts." In Virtual Systems and Multimedia (VSMM), 2010 16th International Conference on, pp. 336-339. IEEE, 2010.
12. Ahmad, Riaz, Syed Hassan Amin, and Mohammad AU Khan. "Scale and rotation invariant recognition of cursive Pashto script using SIFT features." In Emerging Technologies (ICET), 2010 6th International Conference on, pp. 299-303. IEEE, 2010.
13. Garz, Angelika, Markus Diem, and Robert Sablatnig. "Detecting text areas and decorative elements in ancient manuscripts." In Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on, pp. 176-181. IEEE, 2010.
14. Gilliam, Tara, Richard C. Wilson, and John A. Clark. "Scribe identification in medieval English manuscripts." In Pattern Recognition (ICPR), 2010 20th International Conference on, pp. 1880-1883. IEEE, 2010.`
15. Yadav, Nisha, Hrishikesh Joglekar, Rajesh PN Rao, Mayank N. Vahia, Ronojoy Adhikari, and Iravatham Mahadevan. "Statistical analysis of the Indus script using n-grams." PLoS One 5, no. 3 (2010): e9506.
16. Siddiqi, Imran, Florence Cloppet, and Nicole Vincent. "Contour based features for the classification of ancient manuscripts." In Conference of the International Graphonomics Society, pp. 226-229. 2009.
17. Bernard, Simon, Sebastien Adam, and Laurent Heutte. "Using random forests for handwritten digit recognition." In Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), vol. 2, pp. 1043-1047. IEEE, 2007.

18. Sousa, J. M. C., Joao Rogerio Caldas Pinto, Claudia S. Ribeiro, and Joao M. Gil. "Ancient document recognition using fuzzy methods." In The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ'05. 2005.
19. Gatos, Basilios, Kostas Ntzios, Ioannis Pratikakis, Sergios Petridis, Thomas Konidaris, and Stavros J. Perantonis. "An efficient segmentation-free approach to assist old Greek handwritten manuscript OCR." Pattern analysis and applications 8, no. 4 (2006): 305-320.
20. Kashyap, K. Harish, and P. A. Koushik. "Hybrid neural network architecture for age identification of ancient Kannada scripts." In Circuits and Systems, 2003. ISCAS'03. Proceedings of the 2003 International Symposium on, vol. 5, pp. V-661. IEEE, 2003.
21. Sastry, Panyam Narahari, Ramakrishnan Krishnan, and Bhagavatula Venkata Sanker Ram. "Classification and identification of Telugu handwritten characters extracted from palm leaves using decision tree approach." J. Applied Engn. Sci 5, no. 3 (2010): 22-32.
22. Andrew, C. "Building decision trees with the ID3 algorithm." Dr. Dobbs Journal (1996).
23. Bhuvaneswari, G. and Subbiah Bharathi, V., **2015**. An Efficient Positional Algorithm for Recognition of Ancient Stone Inscription Characters. *2015 Seventh International Conference on Advanced Computing (ICoAC)*, Chennai, pp.1–5.
24. Rajithkumar, B.K., Mohana, H.S., Uday, J., Bhavana, M.B. and Anusha, L.S., **2015**. Read and Recognition of Old Kannada Stone Inscriptions Characters Using Novel Algorithm. *2015 Inter- national Conference on Control, Instrumentation, Communica- tion and Computational Technologies (ICCICCT)*, pp.284–288, DOI: 10.1109/ICCICCT.2015.7475291.
25. Janani, G., Vishalini, V. and Kumar, P.M., **2016**. Recognition and Analysis of Tamil Inscriptions and Mapping Using Image Process- ing Techniques. *2016 Second International Conference on Science Technology Engineering and Management (ICONSTEM)*, Chennai, pp.181–184.
26. Katsouros, V., Papavassiliou, V., Simistira, F. and BasilisGatos, **2016**. Recognition of Greek Polytonic on Historical Degraded Texts Using HMMs. *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, Santorini, pp.346–351.
27. Chaowicharat, E., Naruedomkul, K. and Cercone, N., **2016**. Direction histogram: Novel discriminative global feature for thai offline handwritten OCR. *Pattern Analysis and Applications, 19*(4). DOI: 10.1007/s10044-016-0536-0.
28. Vellingiriraj, E.K., Balamurugan, M. and Balasubramanie, P., **2016**. Information Extraction and Text Mining of Ancient Vattezhuthu Characters in Historical Documents Using Image Zoning. *2016 International Conference on Asian Language Processing (IALP)*, Tainan, pp.37–40.
29. Manigandan, T., Vidhya, V., Dhanalakshmi, V. and Nirmala, B., **2017**. Tamil Character Recognition from Ancient Epigraphical Inscription Using OCR and NLP. *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, Chennai, pp.1008–1011.
30. Veena,G.S., Kumar, T.N.R. and Sushma, A., **2018**. Handwritten Off- Line Kannada Character/Word Recognition Using Hidden Markov Model. Proceedings of International Conference on Cognition and Recognition, pp.357–369.
31. Yang, Mingqiang, Kidiyo Kpalma, and Joseph Ronsin. "A survey of shape feature extraction techniques." Pattern recognition (2008): 43-90.
32. . Lu, Dengsheng, and Qihao Weng. "A survey of image classification methods and techniques for improving classification performance." International journal of Remote sensing 28, no. 5 (2007): 823-870.
33. Rajput, G. G., and H. B. Anita. "Handwritten Script Identification from a BiScript Document at Line Level using Gabor Filters." Proc. of SCAKD (2011): 94-101
34. Strobl, Carolin, James Malley, and Gerhard Tutz. "An introduction to recursive partitioning: rationale, application, and characteristics of classification and regression trees, bagging, and random forests." Psychological methods 14, no. 4 (2009): 323.
35. Rajput, G. G., and H. B. Anita. "Handwritten Script Identification from a BiScript Document at Line Level using Gabor Filters." Proc. of SCAKD (2011): 94-101