JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Kannada Handwritten Character Recognition Using KNN, SVM And CNN

[1]Chinmayee Bhat H

[1]Student

[1]Department of Computer Science and Engineering

[1]JSSSTU, Mysore, Karnataka, India

Abstract: Handwriting character recognition (HCR) refers to a computer's ability to receive and interpret handwritten input. Handwritten Character Recognition is one of the most active and difficult research areas in Pattern Recognition. Pattern recognition is a process that utilises raw data to perform a task based on the category of the pattern. HCR is a well-known pattern recognition application. Due to a lack of research, handwriting recognition, particularly for Indian languages, is still in its infancy. The focus is to recognize the kannada handwritten character using machine learning algorithms such as KNN, SVM and deep learning algorithm(CNN). The data collection, data pre processing, feature extraction played a vital role in recognizing the character. KNN, SVM and CNN algorithm accuracy is obtained and compared. CNN gave the highest accuracy and it has been fed to GUI and thus character has been recognized.

Keywords: Feature Extraction, Handwritten Character Recognition(HCR) , KNN, SVM, CNN

1.Introduction

Handwritten character Recognition is the capacity of the computer to recognize the character which is acquired from an image source and interpret the characters. Handwritten recognition is one of the most important areas of the pattern recognition as it provides solutions such as document classification, email processing, automatic data entry, bank check reading, and customer entry from reading. Due to the rapid development of technology, handled devices are very popular. Today, PCs and mobile phones have proven to be more important in our daily lives, are in increasing demand and are also affordable. This created a system called handwritten recognition that replaced keyboards and input devices. It is possible to recognize handwritten character in two ways: offline and online. The offline recognition system writes pre-created characters on paper, scans them and converts them into 2D images, inputs them to computer and recognizes them. Online character recognition system, on the other hand, uses tablets to enter characters in real time PC or tablets. Much research has been performed to the character recognition field of many other languages such as English, Chinese, Arabic but less work is done for Kannada.

Kannada has rich morphological structures unlike the languages like English. Character can be written in many ways. Recognition of kannada characters has been one of the active and challenging areas of research. Large number of letters present in the kannada language makes it as a open problem for the researchers. Kannada Handwritten characters cannot be identified easily due to the differences in handwriting styles, quality of the scanner etc. The focus is to recognize the kannada handwritten character using machine learning algorithms such as KNN, SVM and deep learning algorithm(CNN).

1.1 Dataset Used for Analysis

In this project firstly have to collect the scanned Kannada Handwritten Character dataset from Kaggle website which contains large collection of image data. Dataset consists of 49 classes where each class contains 25 images so the total dataset is 1,225 images. Dataset consists of vowels(swaras), consonants (vyanjanas) and gunitaksharas.

2. System Design

The figure 1 depicts the various phases of the system's development. The interaction between the different application components is depicted, along with where they fit into the development hierarchy because every module in the chosen issue functions separately.

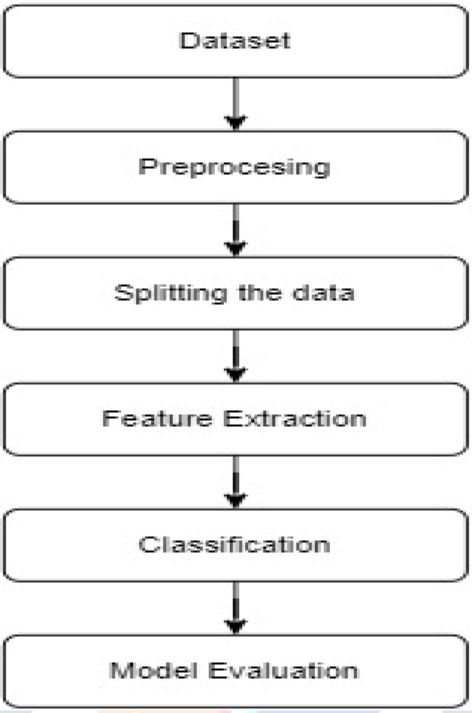


Fig 1 kannada handwritten character recognition system design

The working of the system starts with collection of datasets. Have collected dataset from Kaggle which consists of numerous dataset. Once the data has been collected the immediate next step is to pre-process the data followed by resizing image, removal of noise. Next step is to break down the data into training and testing part. The data has been divided into 80:20 ratio which means for training 80% of data is utilized and for testing 20% of data is used. Next method is to extract the features. Feature extraction technique used is Histogram oriented Gradients (HOG).

Different algorithms such as KNN, SVM and CNN has been used. The model is trained by training the data. The algorithm which gives the highest accuracy that has been considered and it is loaded to the graphical user interface (GUI) where the user is able to feed the input and corresponding ouput will be displayed.

Dataset: The first and foremost step of ML is to collect the required data. Dataset have been collected from Kaggle.. There are 49 classes in the dataset and each class contains 25 images.

Preprocessing : The next procedure is to pre process the data. The method of converting raw data into a useful, understandable way identified as data preprocessing. It is necessary because algorithms gain knowledge from the data and the results of learning for issue resolution largely rely on appropriate data required to cope with the complexity, which are referred to as features.

Splitting the data: Data is divided into two or more subsets. With a two-part split, one part is used to evaluate or test the data and other one to train the model. Have splitted data into 80:20.

Feature Extraction: Next is to extract the features. It enhances the accuracy of learned models by extracting the features from the input data. Have used Histogram of oriented Gradients to extract the features The HOG descriptor focuses on an object's structure or shape.

Classification : Following feature extraction, the data is classified using machine learning algorithms such as Support Vector Machine (SVM), K-nearest neighbour (KNN) and deep learning models such as Convolutional Neural Network.

Evaluating the model : Analyzing a machine learning model's performance and identifying its advantages and disadvantages is the process of model evaluation.It is done using various evaluation metrics. . To know whether the model is working well with new data evaluation metrics such as accuracy and confusion matrix for KNN, SVM and CNN is implemented. The one which gives the highest accuracy is considered and loaded to the GUI where the user can give the input and gets corresponding output.

2. Implementation

In this project firstly have to collect the scanned Kannada Handwritten Character dataset from Kaggle website which contains large collection of image data. Dataset consists of 49 classes where each class contains 25 images so the total dataset is 1,225 images. Dataset consists of vowels(swaras), consonants (vyanjana) and gunitakshara.

3.1 Preprocessing

Data preprocessing or data cleansing is crucial step. It includes missing value treatments, removing the unwanted or noisy data. Image pre-processing refers to operations on images that are performed at the most fundamental level of abstraction. Its goal is to enhance image data by removing unwanted altering or emphasising certain image elements that are crucial for subsequent processing and analysis tasks. The preprocessing steps followed are resizing and removing noise in an image.

Image size may vary so it is necessary to resize the image to establish a base size for all images. To remove the noise from an image Gaussian blur is used. It allows to use the low pass filter in Skimage.

The kernel size is determined by the blurring effect desired. In general, the smaller the kernel, the less visible is the blur.

3.2 Feature Extraction

3.2.1 Histogram Oriented Gradients (HOG)

Histogram of oriented gradients is one of the feature extraction methods used to extract character features by adding up the presence of gradient magnitude and orientation in localised image fragments. The character image is split up into various cells during HOG feature extraction. The gradient orientation and magnitude are computed by each cell. In each cell, a gradient histogram is created. One important feature of HOG is its resistance to changes in geometry and lighting. This technique is used in a variety of applications such as human detection and character recognition. The HOG feature extraction method has several advantages such as recognizing edge or gradient structure as well as local shape characteristics. Translations and rotations have little variation in HOG.

Steps of HOG

- Make the original image 128*64 pixels in size.
- Next is to calculate the gradient. The gradient is generated by combining the image's magnitude and angle.
- After identifying the gradient of each pixel, the gradient matrices are separated into to form a block of cells 8*8. For each block, a 9-point histogram is computed. A 9-point histogram produces a histogram with 9 bins and a 20 degree angle range.
- Once all blocks histograms have been computed, A new block (2*2) is formed by combining four blocks from the nine-point histogram matrix. This clubbing is done overlappingly with an 8-pixel strideConcatenate all nine point histograms for each constituent cell to create a 36 feature vector for all four cells in a block .
- Normalize each block by L2 norm.

Normalization is used to minimize the impact of contrast differences in between pictures of the same thing.A 36-point feature vector is gathered from each block. The horizontal direction has 7 blocks and also vertical direction has 15 blocks. As a result, the length of the HOG feature has increased.

3.3 Classification

3.3.1 K-nearest neighbor (KNN)

K-nearest neighbor is a supervised machine learning algorithm which is used for classification purpose. KNN which is also known as lazy learning uses instance based on learning. It classifies new data point based on similarity. Based on the value of K, the object is assigned to the class ofits nearest neighbor.

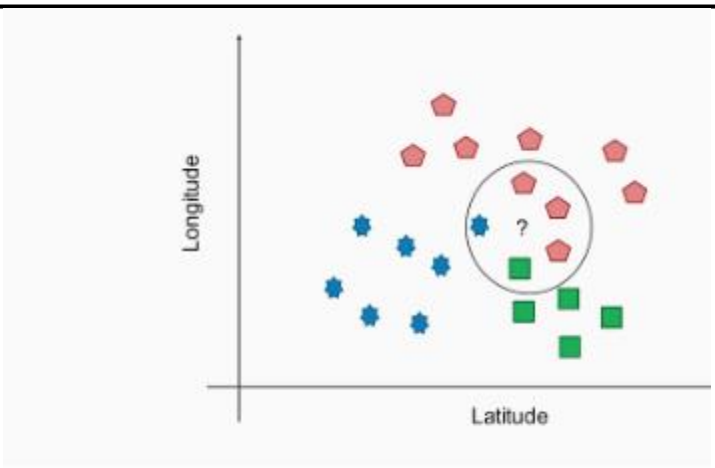


Fig 2 KNN diagram

As shown in the above figure2 suppose if there are three characters then if new data is inserted then based on the data it will predict to which category that data/point belongs. Suppose if the K value is 5 then it will select the 5 points closest in the range. For different k values the accuracy obtained are mentioned in below table.

Table 1 KNN accuracy for different k values

K value	Accuracy
1	50
2	61
3	66
4	70
5	72
6	66

From table 1 it is observed the optimal k value considered is 5. The accuracy obtained is about 72%.

3.3.2 Support Vector Machine (SVM)

Support vector machines (SVMs) are a family of supervised learning models, along with related learning methods, used for classification analysis to examine data and identify distinct patterns.A support vector machine generates a hyperplane or set of hyperplanes in a larger or effectively unlimited space that can be used for classification, regression, or other tasks. Given that the greater the margin, the lower the generalisation error of the classifier, it stands to reason that the hyperplane with the highest distance from the closest training data point of any class will achieve successful separation. This distance is known as the functional margin. SVM receives a set of feature vectors as input, scales, selects, and then the hyperplane separates the data and classifies them to different classes. SVM diagram is shown in the given below Fig 2.

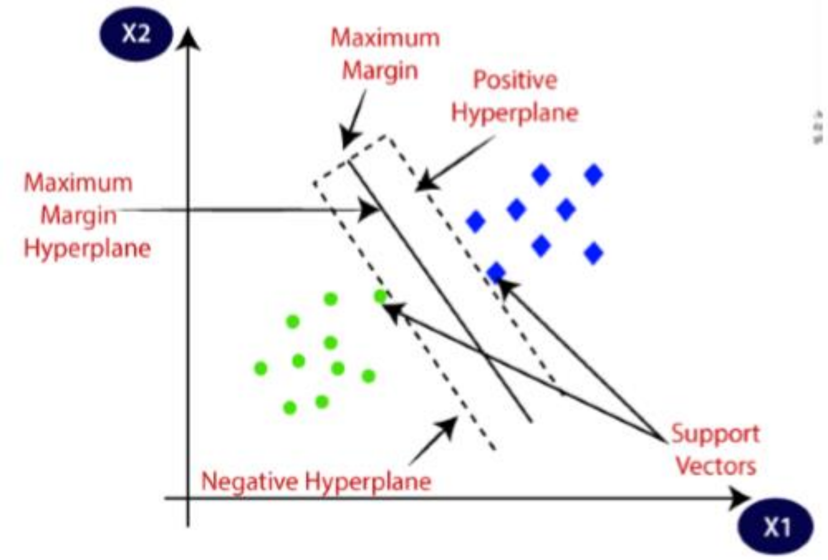


Fig 3 General SVM diagram

3.3.2 Convolutional neural network (CNN)

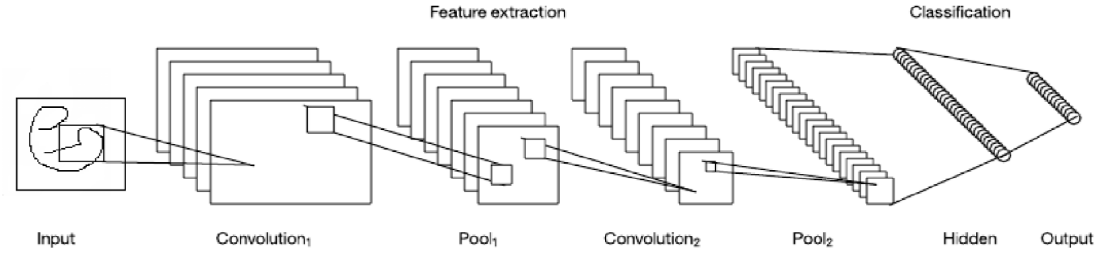


Figure 3 CNN architecture

Figure 3 shows the overall CNN architecture, which is divided into two parts: feature extraction and classification. During feature extraction, every network layer gets the previous layer's output as input and passes the current output to the subsequent layer. The CNN architecture consists of three layers: convolution, max-pooling, and classification. There are two varieties of layers in the network's low and middle levels: convolutional layers and max-pooling layers. Feature mapping is the process of grouping the output nodes of the convolution and maxpooling layers in a 2d space. Each layer's plane is usually formed by combining a plane or planes from preceding layer. The plane's node is attached to a minor portion of the related planes from the preceding layer. Each node of the convolution layer gathers features from source image by applying a convolution operation to the input nodes. By averaging or propagating on the input nodes, the max-pooling layer abstracts features. The propagated features of the lower level layers serve as the basis for the higher level characteristics. Based on the dimensions of the max-pooling and convolutional masks, the dimension of the features decreases as they propagate to the top layer. To obtain improved classification accuracy, nonetheless, the quantity of feature mapping is typically enlarged for mapping extremely appropriate characteristics of the input images. The classification layer of the fully connected network is fed with the outputs of CNN's most recent feature maps.

CONVOLUTION LAYER: In this layer, learnable kernels are convolved with the feature maps from the preceding layer. The output feature maps are created by passing the kernel's outputs via non-linear or linear activation functions like the softmax, hyperbolic tangent, sigmoid, identity functions and rectified linear.

- Sub Sampling layer:

The subsampling layer applies downsampling to the input maps. The input and output mappings are unchanged in this layer. For instance, if there are N input maps, then there will be N output maps. The size of the output maps is going to be reduced due to the downsampling operation, depending on the size of the downsampling mask.

Classification Layer

Based on the convolutional layer's retrieved features, this fully connected layer calculates the value for each object type. The feature map in this study is thought to be 3 x 3 in size, and the activation function is softmax..

- **Steps for CNN implementation**

Input------------------- Shape (n,80,80,1). Here n is number of input images and the number of channels is 1 due to the binary nature of the images.

Layer 1---------------- Convolutional layer with 52 filters and filter size 5*5.Max pooling layer with filter size 2*2.

Layer 2 --------------- Convolutional layer of 64 filters and size of filter is 5*5. Max pooling layer with filter size 5*5.

Flatten() ---------------- It transforms the CNN's convolutional output into a one-dimensional feature vector that can be used by the fully connected layer.

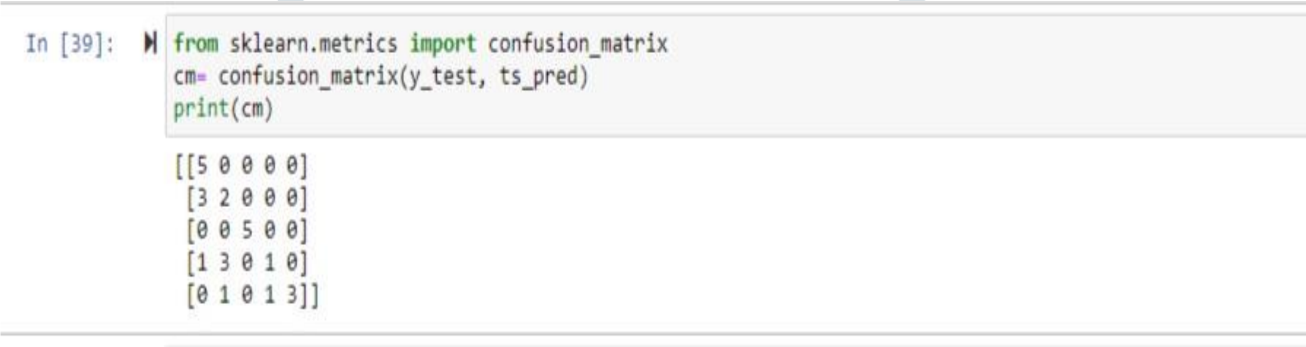
Layer 3 ----------------- Fully connected layer (Dense) has 256 neurons, which has dropout regularization and also rate of 0.5.

Here softmax classification is utilized for the output layer (Fully connected Layer).

4. Results and Screenshots

4.1 Confusion Matrix

Confusion matrix is a NxN matrix used to evaluate classification model performance, where N is the target class number. In the matrix the actual target value is compared to those that were predicted by the learning. This offers a complete picture of how the classification model is functioning and the different kinds of classification errors that can happen.

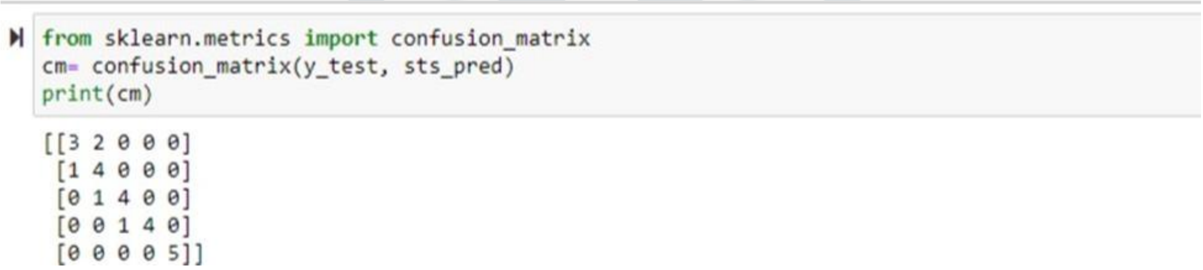


```
In [39]:  from sklearn.metrics import confusion_matrix
          cm= confusion_matrix(y_test, ts_pred)
          print(cm)

          [[5 0 0 0 0]
           [3 2 0 0 0]
           [0 0 5 0 0]
           [1 3 0 1 0]
           [0 1 0 1 3]]
```

Fig 4 Confusion matrix for KNN

The above shown figure 4 is the confusion matrix for KNN. All the diagonal elements are the correct predictions. Upper diagonal values are false positive and lower diagonal elements are false negative.



```
from sklearn.metrics import confusion_matrix
cm= confusion_matrix(y_test, sts_pred)
print(cm)

[[3 2 0 0 0]
 [1 4 0 0 0]
 [0 1 4 0 0]
 [0 0 1 4 0]
 [0 0 0 0 5]]
```

Fig 5 Confusion matrix for SVM

The above shown snapshot 5 is the confusion matrix for SVM. All the diagonal elements represents are the correct Predictions. Upper diagonal values represents the false positive. Lower diagonal values represents the false negative.

4.2 Comparitive Analysis

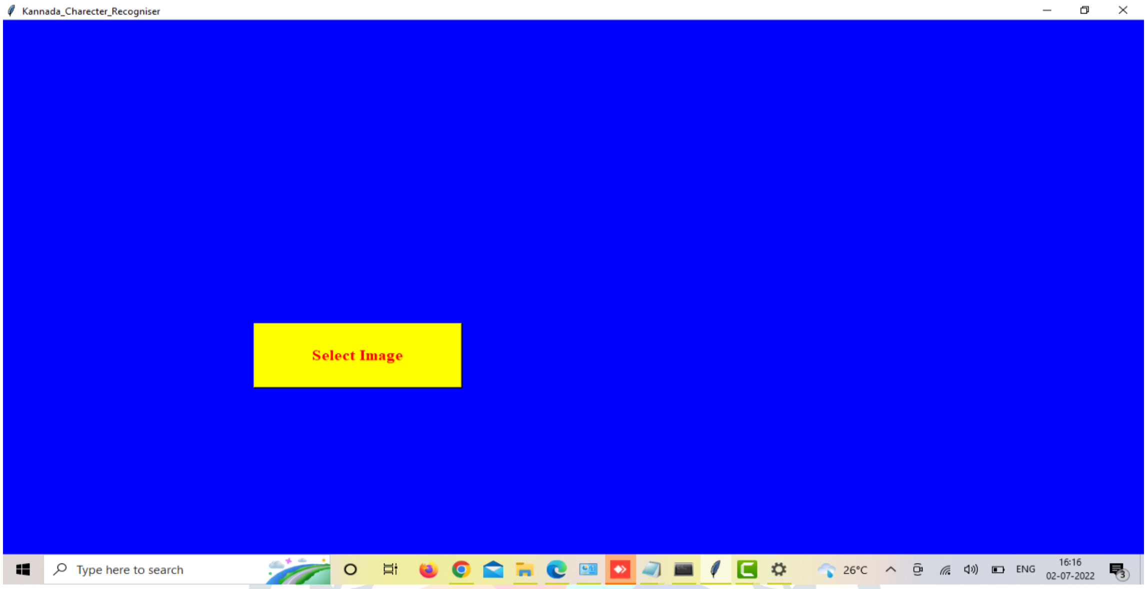
The table 7.1 given below consists of accuracy of KNN, SVM and CNN. It is observed that CNN model gave the highest accuracy. KNN obtained 72%, SVM yielded 88% and CNN accuracy is about 96%. Hence CNN is considered as the best model.

7.3 Comparitive Analysis

Table 2 Comparison of accuracy values of different Models

Models	Accuracy
KNN	72%
SVM	88%
CNN	96%

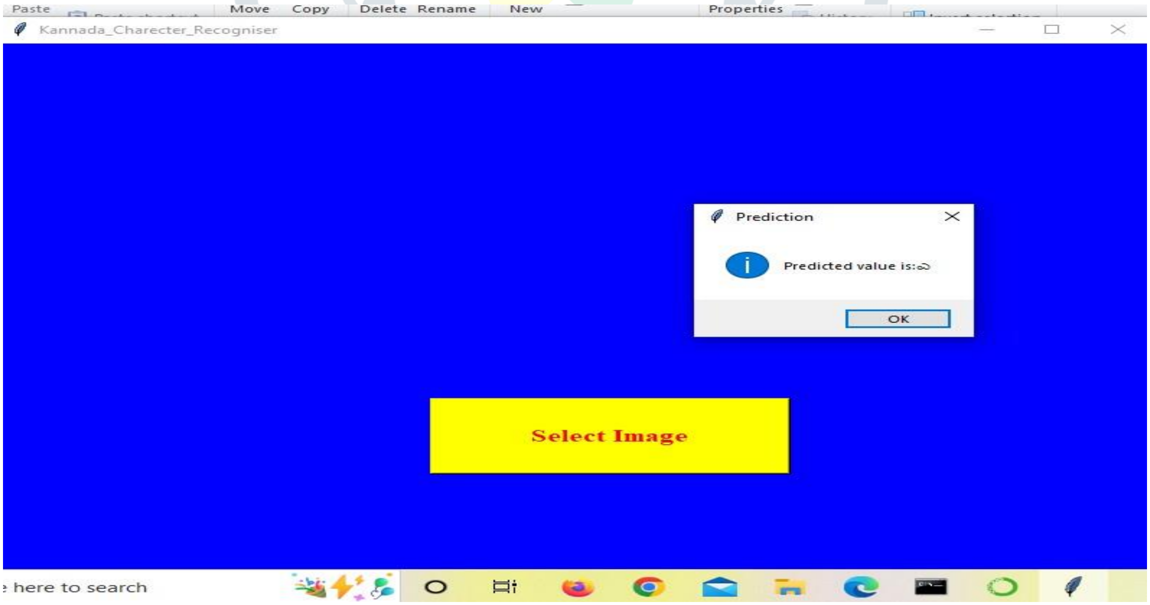
The table 2 given above consists of accuracy of KNN, SVM and CNN. It is observed that CNN model gave the highest accuracy. KNN obtained 72%, SVM yielded 88% and CNN accuracy is about 96%. Hence CNN is considered as the best model.



s

Fig 6 Screenshot of web page

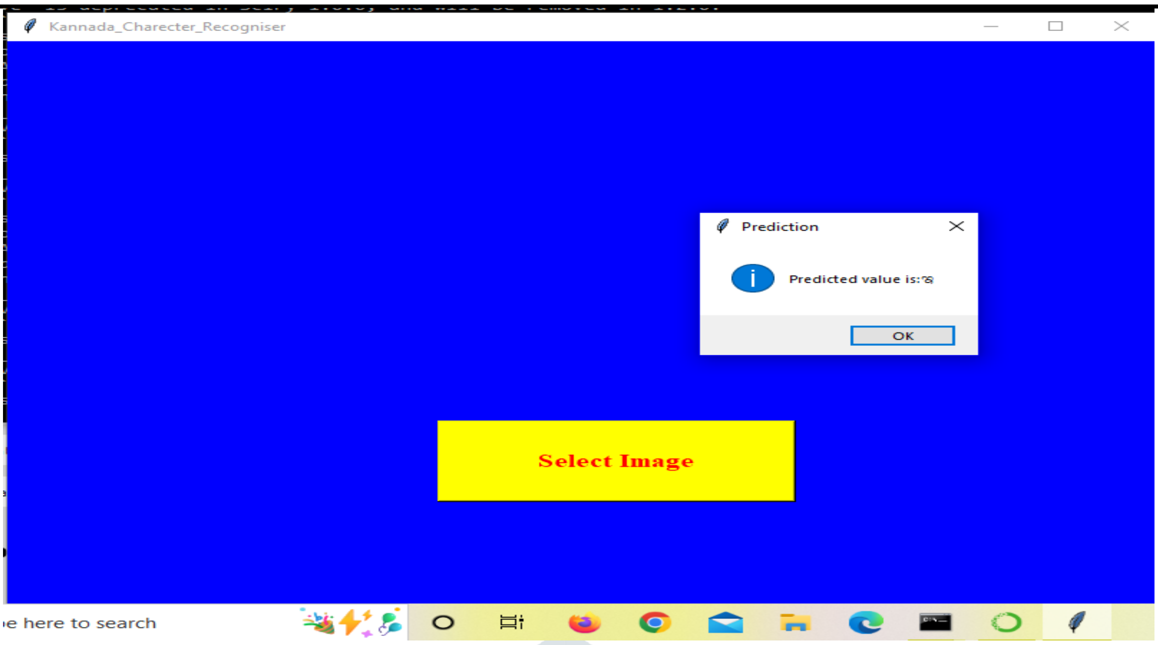
The web Interface implemented for this project is shown in Fig 7.1. Here the user is able to select the image.



Predicted value is:ಲ

Fig 7 Screenshot of recognised Character

Figure 7 depicts the screenshot of recognized character. When the user gives the input as ಲ the system recognizes it as ಲ.



Kannada_Charecter_Recogniser

Prediction

Predicted value is:ಙ

Figure 8 Screenshot of recognised Character

Figure 8 depicts the screenshot of recognized character. When user gives the input ಙ the system recognizes it as ಙ.

Conclusion and Future Work:

A simple and efficient Kannada handwritten character recognition system has been developed. Data collection, preprocessing, Feature Extraction and classification played a vital role in recognizing the character. In this study machine learning techniques such as KNN, SVM and deep learning technique CNN are implemented inorder to recognize the characters. KNN, SVM and CNN accuracy are compared and the one which gives the highest accuracy is considered and it is loaded to the web. It has been observed that CNN gave the highest accuracy that is 96%. GUI is designed in such a way that anybody can operate it. User can sit at their conveince and use the system.When Kannada Handwritten character is given as input it recognizes it successfully. Future enhancement is to recognize the handwritten kannada words as well as to translate text to speech.

REFERENCES

[1] Ganpat singh Rajput, Rajeshwari Horakeri, “Zone based Handwritten Kannada Character Recognition Using Crack code and SVM”, IEEE International conference on machine learning 2013.

[2] Sridharmurthy S K, Sudarshana Reddy, “PCA based feature vector for handwritten Kannada characters recognition”, 2015.

[3] Sulaiman Khan, Hazrat Ali, Dr. Zahid Ullah, “KNN and ANN-based Recognition of Handwritten Pashto Letters using Zoning Features “, 2018.

[4] M. C. Padma, Saleem Pasha, “Recognition of handwritten kannada characters using hybrid features”, IEEE International Conference on Image Processing, 2015.

[5] Leena R Ragha, M Sasikumar, “Adapting moments for Handwritten Kannada Kagunita Recognition”, Second International Conference on Machine Learning and Computing, 2010.

[6] Betselot Yewulu Reta, Dhara Rana, Gayatri Viral Bhalerao, “Amharic Handwritten Character Recognition Using Combined Features and Support Vector Machine”, 2018.

[7] M. C. Padma, Saleem Pasha, “Feature extraction of handwritten Kannada characters using curvelets and principal component analysis”,IEEE International Conference on Image Processing, 2015.

[8] Roshan Fernandes, Anisha P Rodrigues, “Kannada Handwritten Script Recognition using Machine Learning Techniques”, IEEE International Conference on Distributed Computing,VLSI, Electrical Circuits and Robotics, 2019.

[9] Kavya N, Madhushree N, Maheshwari S, “Handwriting Recognition Using SVM”, International Journal of Engineering Research & Technology, 2019.

[10] D. Eshwar Reddy, K. V. Pranathi Naidu, M. Kartheek Srinivas, Abdul Raheem, Ms Sneha Sureddy, “Handwritten Character Recognition using SVM”, International journal of Advance science and technology, 2020.

[11] Anupama Angadi, Vali Kumaro Vatsavayi, “ A Deep Learning Approach to Recognize Handwritten

Telugu Character Using Convolution Neural Networks" ,2017.

[12] Sanugula Durga Prasad, "Telugu handwritten character recognition using adaptive and static zoning methods", IEEE students technology symposium, 2016.

[13] www.youtube.com

[14] www.wikipedia .com

[15] www.google.com

[16] https://www.kaggle.com/general/221455