Handwriting Recognition System for South Indian Languages – A Technical Review

Keerthi Prasad G
Research Scholar VTU
Dept. of ISE, GMIT
Davanagere, INDIA
keerthiprasad@gmit.ac.in

Vinay Hegde
Associate Professor
Dept. of CSE, RVCE
Bangalore, INDIA
vinayvhegde@rvce.edu.in

Asha K
Research Scholar VTU
Dept. of ISE, GMIT
Davanagere, INDIA
ashak@gmit.ac.in

Krishnappa H K
Associate Professor
Dept. of CSE, RVCE
Bangalore, INDIA
krishnappahk@rvce.edu.in

***Abstract*— Handwriting recognition is the most popular area of research which provides major contribution to the trending technology - mobile computing. This paper provides technical details for the implementation of Handwriting recognition system. The techniques suitable for offline and online handwriting recognition system are also discussed. Some of the pre-processing and classification algorithms like normalization; re-sampling, Principal Component Analysis (PCA) and Dynamic Time Wrapping (DTW) are presented in this paper.**

***Keywords*— Online handwriting recognition, Offline handwriting recognition, PCA, DTW, Pattern recognition, Handwriting recognition**

I. INTRODUCTION

Recent trend in computer world is mobile computing and handwritten document processing. In this regard much research has been done for international languages like English, but least work is done for South Indian languages because of hardware limitation to accept large character set of Indian scripts. Hence, handwriting recognition system for South Indian languages is in demand. Majority of the people in South India speak one of the four major Dravidian languages: Telugu, Tamil, Kannada and Malayalam.

Handwriting recognition is a popular area of research under pattern recognition and image processing. There are two categories of handwriting recognition namely offline and online.

Offline

Offline handwriting recognition accepts the input characters to be recognized in the form of image.

Online

Online handwriting recognition accepts the string of (x, y) coordinate pairs in real time from an electronic pen touching a pressure sensitive digital device.

Offline handwriting recognition is useful in digital handwritten document processing and online handwriting recognition is useful in natural way of interacting with computing device.

Further handwriting recognition is divided into writer dependent and writer independent. A writer independent system recognizes wide range of writing styles, whereas a writer dependent system is trained to recognize the writing style of specific user. There is a need for the implementation of both online and offline writer independent handwriting recognition system for South Indian languages.

II. HANDWRITING RECOGNITION SYSTEM

Generalized steps involved in handwriting recognition system are data collection, pre-processing, feature extraction and classification as shown in Figure 1.

The output obtained from one phase is fed as input to the next phase.

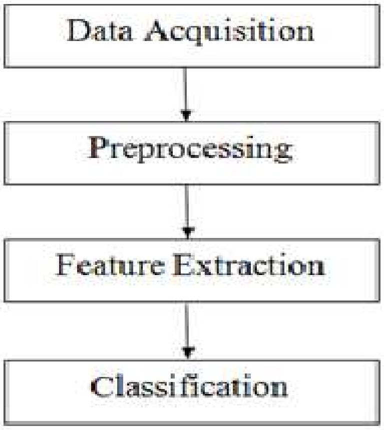


Figure 1.Phases in Handwriting recognition

Data acquisition

This is the first step in handwriting recognition system. In offline system data is in the form of image consisting of handwritten information whereas in online system data is collected in real time by capturing the coordinate points of pen movement. A generic data collection algorithm for online handwriting system is shown below.

```
Data Collection (XR, YR, mPath)
// XR – List of X co-ordinate points
// YR – List of Y co-ordinate points
// mPath – Path containing XY co-ordinates to draw
Start
Step 1
    If (event is in the input area)
        Go to step2
    Else
        Ignore data points
Step 2
    If (event is pointer pressed) //beginning of a stroke
    Add x to XR //x is x-coordinate of the point pressed
    Add y to YR //y is y-coordinate of the point pressed
```

978-1-5386-2044-1/17/$31.00 ©2017 IEEE

```
        Initialize the mPath
Else if (event is pointer dragged)
        Append x & y to XR & YR
        Calculate mPath
Else //the event is pointer released (end of a stroke)
        Call the draw method to draw the mPath points
        Reset mPath
End
```

Pre processing

The accuracy of any handwriting recognition system can be improved by applying proper pre-processing techniques. There are many pre-processing techniques like Binarization, noise elimination, slant correction, size normalization, smoothing or thinning, re-sampling etc. These pre-processing techniques are system dependent i.e. some of the techniques are not applicable for online or offline handwriting recognition system. Basically pre-processing techniques are used to present the clear input to the recognition system by removing noise or distortion present in the input.

Binarization

This pre-processing technique is applicable to offline handwritten as the input is in the form of image. This step is necessary in offline handwritten recognition because the captured /scanned image may be having data/information in different colours. The idea behind binarization is to separate the foreground and background information by setting the background to white. In case of online handwritten this step is not required as sequence of coordinate points of the moving pen/hand is captured in real time in digital form.

Noise removal

This step is compulsory in case of offline handwriting recognition as the accuracy of the system much depend on the clarity of the input image. Most of the time additional noise may be present in the input image either because of the quality of capturing device or aging of image or external dust. This additional noise is called impulse noise or salt and pepper noise, by applying filtering techniques like median filter or morphological filter or Dvadasham (Dodeca) Edge Filter (DEF) [1]. Basically in median filtering, the value of each pixel is replaced by the median of pixel values in the specified dimension of neighbourhood. Online handwriting recognition system does not require this step to be performed as the chance of having external noise in the input is quite minimal.

Slant correction

Slant correction is a process of detecting and correcting the skew by estimating the angle at which an input image is rotated during the data acquisition in offline recognition system. Skew detection and correction is applicable to both offline and online recognition system, it can be applied correct the skew of input image in offline recognition and correct the skew of a line of sentence or word in both offline and online recognition system. Some of the techniques for slant correction are Projection profile method, run length based technique, Hough transform, extrema method, generalized chain code estimator etc [2].

Size normalization

This technique is applied to bring uneven sized character present in the input to a predefined size to make it even with respect to all the characters present in the input. Size normalization can be done by comparing the input stroke border frame with assumed fixed size frame as shown in the below algorithm.

Size Normalization (XR, YR, N)

```
// XR – List of X co-ordinate points
// YR – List of Y co-ordinate points
// N – No. of points
Start
Step 1
        Compute maximum x from the XR series
Step 2
        Compute maximum y from the YR series
Step 3
        Initialize i=1
Step 4
        Repeat the following steps while i less than N
        XR[i] = XR[i] / maxX * Scale factor
        YR[i] = YR[i] / maxY * Scale factor
End
```

Smoothing or thinning

Thinning is applied to have characters with single pixel thickness by removing the flickers which may exist in handwritten data because of handwritten style and the hardware used. Thinning can be done by modifying each pixel value with mean value of k-neighbours and the angle subtended at position from each end.

Re-sampling

Re-sampling is performed on the input character data to keep all the pixels representing the character at equal distance as shown in the below algorithm.

Re-sampling (XR, YR, RNP)

```
// XR – List of X co-ordinate points
// YR – List of Y co-ordinate points
// RNP – Required No. of points
Start
Step 1
        Initialize the following variables
        X1 = XR [0]
        X2 = XR [1]
        Y1 = YR [0]
        Y2 = YR [1]
        Dist = Distance between 2 points
        TD = Total Distance
        DL = Incremental length is TD / RNP-1
        RESX [0] = X1 //Resulting X points
        RESY [0] = Y1 //Resulting Y points
Step 2
        Repeat the following for k=1 to RNP
        Case 1: If (Dist == DL)
                RESX [k] = X2
                RESY [k] = Y2
                X1 = X2 and Y1 = Y2
                X2 = XR [k] and Y2 = YR [k]
                Compute Dist
        Case 2: If (Dist < DL)
                Compute Dist until Dist >= DL
                If (Dist == DL)
                Go to case 1
                Else
                Go to case 3
        Case 3: If (Dist > DL)
                Find out a new point between 2 points
End
```

Feature extraction

After normalizing the data, by applying pre-processing techniques, the most challenging part in recognition phase is segmentation. Segmentation is not required for isolated handwritten character recognition but it is very important in case of document or word recognition. Handwritten document recognition system involves line segmentation which separates each line of text followed by word segmentation which separates each word in a line [3] as shown in Figure 2, Figure 3 and Figure 4.

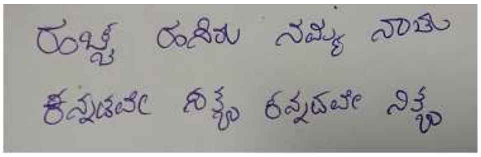


Figure 2.Before Segmentation

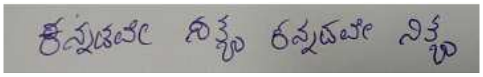


Figure 3.Line Segmentation

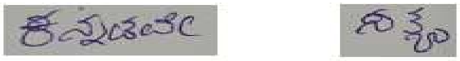


Figure 4.Word Segmentation

It is very important to identify and extract unique and correct features from character set of a particular language to maximize the recognition rate with the least amount of elements. Features used to represent a character are language dependent hence the method that gives better result for a particular script cannot be applied for other scripts.

Feature extraction plays vital role in handwriting recognition system for South Indian languages like Kannada, Tamil, Telugu and Malayalam. As the character set of these South Indian languages contains wide variety of structural features like loops, crossings, headline, straight line, dots etc. and statistical features like zoning, projection and profiles.

The classification step is discussed as a separate section.

III. CLASSIFICATION

Classification is the last and the most important step which carry out some form of comparison between a given unknown handwriting pattern to reference handwriting patterns to assign one of the references to the unknown one. In this section some of the work done in the recognition of South Indian languages is discussed.

Keerthi Prasad *et al*., [4] proposed online Kannada handwritten character recognition for mobile devices. The proposed system was implemented on mobile devices for Kannada vowels and consonants using Principal component analysis (PCA) and dynamic time wrapping (DTW) techniques and they obtained an accuracy of 88% for PCA and 64% for DTW approach.

The below depicted algorithm is used to obtain a more compact encoding of the data by applying PCA thereby reducing the dimension of the data from n to p, where p < n.

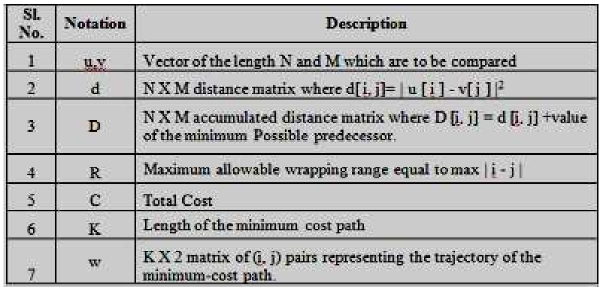
PCA (Data)

```
// Data – M * N matrix of input data samples (M dimensions, N trials)
// Signals – M * N matrix of projected data
// PC - each column is a Principal component
// V – M * 1 matrix of variances
Start
Step 1
        Subtract off the mean for each dimension
Step 2
        Calculate the covariance matrix
        Covariance = 1 / (N-1) * Data * Data’;
Step 3
        Find the Eigen vectors and Eigen values
Step 4
        Extract diagonal of matrix as vector
Step 5
        Sort the variances in decreasing order
Step 6
        Project the original data set
        Signals = PC’ * Data;
End
```

DTW Algorithm

A dynamic programming approach is used to find this minimum-distance wrap path. DTW technique is used to finds the optimal alignment between two time series if one time series may be wrapped non-linearly by stretching or shrinking it along its time axis. This wrapping between two time series can then be used to find corresponding regions between the two time series or to determine the similarity between the two time series. Table 1 gives the details of the notation that are being used with respect to DTW algorithm.

Table 1.Notations used in DTW



Sl. No.	Notation	Description
1	u,v	Vector of the length N and M which are to be compared
2	d	N X M distance matrix where d[i, j]= | u [i] - v[j] |^2
3	D	N X M accumulated distance matrix where D [i, j] = d [i, j] +value of the minimum Possible predecessor.
4	R	Maximum allowable wrapping range equal to max | i - j |
5	C	Total Cost
6	K	Length of the minimum cost path
7	w	K X 2 matrix of (i, j) pairs representing the trajectory of the minimum-cost path.

Dynamic Time Wrapping (u, v)

This algorithm uses two vector named u and v with length N and M.

```
Start
Step 1
        Create the distance matrix
        d [i, j] = (u [i] - v [j])2
Step 2
        Initialize the first element of the DTW Matrix
                D [1, 1] = d [1, 1]
Step 3
        Compute the first row of D
        D [i, 1] = d [i, 1] + D [i-1, 1] for i=2 to N
Step 4
        Compute the first column of D
        D [1,j] = d [ 1, j] + D [1, j-1] for j = 2 to M
Step 5
        Compute the rest of the D
        For i= 2 to N, j=2 to M
D [i, j] = d [i, j] + min {D [i-1, j], D [i-1, j-1], D [i, j-1]}
        End for
Step 6
        Compute Total Cost C = D [N, M]
End.
```

Algorithm to Compute the Minimum-cost Path

The algorithm depicted below is used to find the minimum cost path between a test pattern and stored reference pattern.

Minimum Cost Path (D)

```
Start
Step 1
        Initialize the following variables
        K=1
        i = Length of the Test pattern i.e. N
        j = Length of the Reference pattern i.e. M
Step 2
        Perform the following steps while either i or j is greater than 1
        1. If i equals 1 then
                Decrement j by 1
        2. Else
        3. If j equals 1
                Decrement i by 1
        4. Else
        i. Assign x with Minimum of D [i-1, j], D [i-1,j-1] and D[ i, j-1]
        ii. If x corresponds to D [i-1, j] then
                Decrement i by 1
        iii. Else if x corresponds to D [i, j-1]
                Decrement j by 1
        iv. Else
                Decrement both i and j by 1
        5. The w array will contain the indexes which give the minimum-cost
         path when traversed through the DTW Matrix D
End
```

S.A Angadi and Sharanabasavaraj H.Angadi [5] proposed a method that uses structural features and Support Vector Machine (SVM) classifier for recognition of handwritten Kannada characters. They obtained recognition accuracy of 89.84% and 85.14% for handwritten Kannada vowels and consonants.

Anitha Mary M.O *et al.*, [6] proposed a method for Offline Malayalam character recognition using multiple classifier combination technique. Their system uses Chain code histogram and Fourier descriptors features extracted from preprocessed images and fed into feed forward neural networks. They achieved an accuracy of 92.84% and 96.24% respectively for the writer independent and writer dependant scheme.

P.V Manoj *et al.*, [7] proposed a system Handwritten character recognition for English and Telugu Scripts using Multi Layer Perceptions (MLP). Hand written characters were extracted by measuring X, Y and Z co-ordinates of pixels of each and every character reproducibly. The intricacy of the characters in the language is reduced by dividing all of the characters into 6 sub-groups using printed characters and their correlation coefficient values.

Anitha Mary M.O *et al.*, [8] proposed a method which uses the combination of Chain Code Histogram and Differential Chain Code Histogram based features for recognition of isolated basic Malayalam characters. They obtained an accuracy of 92.75% using neural network classifier. Authors [9] have also proposed a novel method for the isolated Malayalam character recognition based on the combination of global and local features. Global features include moment invariants and projection features and gradient features of the characters are considered as local features. Proposed method achieves an accuracy of 96.16% recognition using a two layer feed forward neural network as a classifier.

IV. Conclusion

Handwriting recognition system for languages like English is commercially available and much work has to be done for South Indian languages like Kannada, Tamil, Telugu and Malayalam. This paper has more technical details which is very useful for implementation of both offline and online handwriting recognition system for Indian regional languages. We hope this technical review encourages the research in the field of Indian scripts.

References

[1] Naveen R Chanukotimath, Feroz Khan, Keerthi Prasad G, Imran Khan, Deepak D J, Nasreen Taj M B, "Dvadasham (Dodeca) Edge Filter for Impulse Noise, Gaussian Noise, Quantum Noise Reduction in Images", Compusoft, An International Journal of advanced computer technology, July 2013, Volume-II.

[2] Nazia Makkar and Sukhjit Singh, "A Brief tour to various skew detection and correction techniques", IJSETT, 2012, pp 54-58.

[3] A. Sushma and Veena G.S "Kannada handwritten word conversion to electronic textual format using HMM model" *International conference* on *CSISS* 2016, pp 330-335.

[4] Keerthi Prasad, Imran Khan, Naveen R Chanukotimath and Firoz Khan. "On-line Handwritten Character Recognition System for Kannada using Principal Component Analysis approach", In *Proc. WICT-12*, Trivendram, India.

[5] S.A Angadi and Sharanabasavaraj H.Angadi. "Structural Features for Recognition of Hand Written Kannada Character based on SVM". International Journal of Computer Science, Engineering and Information Technology, Vol. 5, No. 2, April 2015.

[6] Anitha Mary M.O. Chacko, Dhanya P.M, "Combining Classifiers for Offline Malayalam Character Recognition", Emerging *ICT* for bridging the future, Vol. 2, Springer International Publishing, Switzerland 2015.

[7] P.V Manoj, A.K Sahoo, Samudra Gupta Maurya and Rohith Kumar, "Handwritten Character Recognition for English and Telugu Scripts Using Multi Layer Perceptions (MLP)", IJSET, Vol. 3, pp. 730-733, June 2014.

[8] Anitha Mary M.O. Chacko, Dhanya P.M, "Combining Classifiers for Offline Malayalam Character Recognition", Emerging *ICT* for bridging the future, Vol. 2, Springer International Publishing, Switzerland 2015.

[9] Anitha Mary M.O. Chacko, Dhanya P.M. "A differential chain code histogram based approach for offline Malayalam character recognition", International conference on communication and computing, pp. 134-139, 2014.