A Review on Recognition of Online Handwriting in Different Scripts

Nilakhi Saikia[1], S.R. Nirmala[2]

[1](Gauhati University Institute of Science and Technology, India)
[2](Gauhati University Institute of Science and Technology, India)
Corresponding Author: S.R. Nirmala

Abstract: Online handwriting recognition or character recognition is the process in which a handwritten message is recognized by processing the handwritten data. It is the process of converting handwritten characters to machine format. In handwriting, the strokes are composed of two coordinate trace in between pen down and pen up labels. Wide range of features is extracted to perform the recognition. Many research works have been done for English, Japanese, Chinese and Korean languages. During the past decade a vast amount of research has also been done on some Indian scripts, viz., Malayalam, Telugu, Devanagari, Gurumukhi, Hindi, Bangla, Assamese, etc. The work presented in this paper deals with the various processes taken up by the researchers in recognizing of the handwriting of various scripts. In this paper, a detailed study of various methods and classifiers used by the researchers to recognize the scripts are made. The comparison of accuracy obtained in different methods is also presented.

I. Introduction

Computing technology has been spreading and advancing at exponential speeds in the recent years. Tablets and smart phones have become day to day necessities among the masses. Effortless and efficient methods are looked for getting better input output modes in these devices. With the advancement in human-computer interface, more advanced ways of interacting are developed. Along with keypad, speech and visual modes, handwriting has also become a preferable mode for entering in-formation to these devices. Speech and Handwriting are effortless mode of communication. Hand-writing may be viewed as an alternative to the existing keyboard based approach, if an efficient recognition system can be developed. Online handwriting recognition or character recognition is the process in which written message is recognized by processing the handwritten data. There are many applications of handwriting recognition viz., interpreting intelligible handwriting, input from paper, touch screen device or smart phones; vehicle number plate tracking, sign board reading; signature verification/writer recognition; replacement of keyboard. Handwriting recognition can be divided into two types; Offline handwriting recognition and online handwriting recognition. In offline handwriting recognition, the text is written in a piece of paper and scanned. The scanned data is then processed for recognition. The data is in the form of bitmap image. In online handwriting recognition system, some trained databases are used as input texts. The pen based computer screen senses the movement of the pen tip. A basic unit has to be selected to recognize the handwritten text, e.g. word, character, strokes or any suitable component. When this basic unit is recognized, then the text is recognized using a dictionary. There are various modeling techniques used for recognition of the patterns, such as, Hidden Markov Model (HMM), Support Vector Machines (SVM), Elastic Matching, Artificial Neural Network, etc.

II. Background

The different phases of a Handwriting Recognition system (shown in Figure. 1) comprises of preprocessing in the first stage. Secondly, extraction of features from the preprocessed data. Next is designing the classifier using various modeling approaches. Finally, a decision is made whether to accept or reject a label.

Preprocessing: In online handwriting recognition system the character or the pattern is captured as (x, y) coordinates. The preprocessing steps include, size normalization, smoothing, removing duplicate points, interpolation and resampling [4]. The main objective behind preprocessing is to remove or reduce any imperfections caused by the acquisition devices, perform smoothing in the irregularities caused by erratic handwriting, minimize the variations in the writing pattern [25].

---

1. Size Normalization: The main purpose of this step is to normalize the variation in pattern sizes. These variations occur due to the different writing styles and other criteria [4]. The pattern is scaled to get the normalized size. The stroke size can be normalized to a unit square [3].

$$scale(x) = \frac{X}{\max(x) - \min(x)} \qquad (1)$$

$$scale(y) = \frac{Y}{\max(y) - \min(y)} \qquad (2)$$

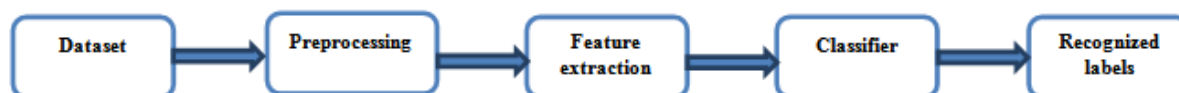$$scale = \min\big(scale(x), scale(y)\big) \qquad (3)$$

Figure 1: Stages of a handwriting recognition system

2. Smoothing: Smoothing is performed to remove the noise or jitter from the handwritten word. Smoothing is done by replacing (x(t),y(t)) by the average value of its neighbor [3]. A moving average window of fixed size can be used for smoothing [1].
3. Removing duplicate points: These are the points that are of no use during recognition and are removed before feature extraction. These points occur due to data redundancy [4].
4. Interpolation: When the system is unable to catch up with the speed of writing as a result of which some points are missed.
5. Resampling: The number of captured points varies due to writing speed and other factors. Different characters in online handwriting can have different number of points along the trajectory [7]. Resampling is done by interpolation of missing points. The resampled data has a series of equidistant points in time and space [4].

Feature Extraction: There are three basic features. The first is the preprocessed (x, y) coordinates of the resampled coordinate sequence, second and third are the first derivative and second derivative of the (x, y) coordinate values. The computation of first derivative is done using equation given [4].

$$x'(j) = \frac{\sum_{i=1}^{2}\big(x(j+i) - x(j-i)\big)}{2\sum_{i=1}^{2} i^2} \quad (4)$$

The second derivative is calculated from the following equation [4].

$$y'(j) = \frac{\sum_{i=1}^{2} y(j+i) - y(j-i))}{2\sum_{i=1}^{2} i^2} \qquad (5)$$

Some other features like linearity, curvature, curliness, bitmap, aspect ratio, chain code, writing direction, slope etc. can also be extracted [4], [25]. Chain code histogram features and point float features are two features that give high performance in the Indian scripts [6]. In the chain code feature, the change in direction while moving from one direction to other is calculated. In point float feature, the input character produces a feature vector which is of length 300.

Recognition: The input to online handwriting recognition system is the movement of digital pen on the (x, y) coordinates ofa trace [3]. A stroke is the sequence of coordinates obtained from one pen down to pen up [1]. By studying the pen down to pen up information of all the characters present in the database and based on the similarity they can be grouped into unique classes. A character can be written either in a single stroke or a combination of multiple strokes. After the analysis of the database, the maximum number of strokes required to write a character can be set [12]. Various classifiers like Hidden Markov Model or HMM classifier [2], SVM [12], ARTMAP approach [1], ANN, etc. can be used as modeling technique to recognize the stroke. When an unknown character input comes up, the stroke classifier identifies a sequence of strokes which is then checked in reference set to recognize the character. Some researchers have made comparison of various modeling techniques. The Hidden Markov Model or HMM classifier is a finite state machine which has a set of hidden states [5]. Each of the sample character contains various strokes within it which are subdivided into sub strokes. The feature vector of all these sub strokes is then converted into states. These states then form the state sequences. From the state sequences, the initial state distribution and the state transition probabilities can be estimated. HMM can solve the problem segmentation in pattern or character recognition [2]. NN classifier is based on Dynamic Time Wrapping (DWT). It calculates the nearest prototype [6]. It can match two curves of

unequal length which makes it an important classifier in online handwriting recognition. Another classifier called the MLP or the Multilayer Perception which is trained by the BP or back propagated architecture is a neural network architecture used in handwriting recognition [6]. Another neural network architecture based on Adaptive Resonance theory (ART) is the fuzzy ARTMAP [1]. It is capable of fast, stable, online, or supervised, incremental learning, classification and many more. The simplified fuzzy ARTMAP can be developed by removing the redundancies in the architecture. SFAM is faster than FAM (Fuzzy ARTMAP). It is a very useful technique in recognition of online handwriting of Indian script. Another classifier most widely used for recognition is SVM or Support Vector Machines. SVMs (Support Vector Machines) are techniques used for data classification. The main objective behind the SVMs is to produce a model based on the training data, which estimates the target values of the test data. PolynomialKernel and Gaussian radial basis functions are the common kernel functions in SVM.

III. Literature Review

The previous related works on handwriting recognition systems are discussed below:

In [1], a recognition system for online Malayalam handwriting was proposed using Simplified Fuzzy Artmap (SFAM) Approach. The features considered were, preprocessed (x, y) coordinates; start quadrant, end quadrant; horizontal and vertical point density; loop; cusp; stroke length; aspect ratio; start, end and start-end direction. The features were extracted and were passed to a SFAM artificial neural network (ANN) classifier for training and classification. Fuzzy ARTMAP is a neural network architecture which is based on Adaptive Resonance Theory (ART). The redundancies in ARTMAP networks were removed successfully to develop the simplified fuzzy ARTMAP (SFAM). The system was trained with 29 user's data and tested with 25 user's data. The recognition accuracy achieved was 98.26%.

In [2], a recognition system using Hidden Markov Model with sub stroke level features of cursive Bangla word was proposed. Each complex preprocessed stroke was divided into simpler shaped stroke.
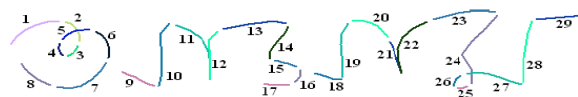

Figure 2: Sub-strokes obtained for a sample word [2]

From each of the sub-stroke, 8 scalar feature values were computed and extracted which were then trained and tested using HMM modeling. The use of HMM modeling solves the problem of segmentation in the system. The writing models used were holistic, combined characters, pseudo characters and context dependent. A database of 14,073 Bangla handwriting samples of 50 city names were written by 163 writers. Out of these 87 writers provided 7557 samples for training and remaining 76 writers provided 6516 samples for testing. The accuracy rate achieved was more than 93%.

In [3], the recognition of online handwriting for Assamese language is done using both HMM and SVM modeling and their performance is compared. 181 different Assamese strokes are trained to generate Recognition models using HTK (HMM Toolkit) and LIBSVM (SVM Toolkit). The developed stroke classifier gives average recognition accuracy rate of about 94 % in case of HMM and 96 % in case of SVM. The akshara level average performance rate is 84.67 % in case of HMM and 86.23 % in case of SVM. The SVM based system gives better performance than HMM based system by 2 % in stroke accuracy and then 1.56 % in akshara case.

In [4], handwriting is recognized using Npen++ recognition engine. The main steps involved during preprocessing were; computing baselines, normalizing size, normalizing rotations, interpolating missing points, smoothing, normalizing slant, resampling and removing the delayed strokes. The basic features, the preprocessed coordinates, the first and the second derivative were extracted for recognition. Some other features were also considered viz. vertical position, writing direction, curvature, pen up/pen down, hat feature, aspect ratio, curliness, linearity, and context map. Multistate time delay neural networks (MS-TDNN) are the core recognition engine of NPen++ which integrates recognition and segmentation into one single network. The extracted features were trained and tested using this recognition system and yielded good recognition rates. For 5000 words, the accuracy rate was found to be 96%, for 20,000 words, the accuracy rate was found to be 93.40% and for 50,000 words, the accuracy rate was 91.20 %. Some of the features extracted are given below:

Normalized x-y traces: Three basic features were considered. The first set was the preprocessed (x, y) coordinates of the resampled coordinate sequence, second and third set consist of the first derivative and second derivative of the (x, y) coordinate values. The computation of first derivative was done using equation given below.

$$x'(j) = \frac{\sum_{i=1}^{2}(x(j+i) - x(j-i))}{2\sum_{i=1}^{2} i^2} \quad (6)$$

The second derivative is calculated from the following equation:

$$y'(j) = \frac{\sum_{i=1}^{2}(y(j+i) - y(j-i))}{2\sum_{i=1}^{2} i^2} \quad (7)$$

Aspect Ratio: The aspect ratio of the trajectory in the vicinity of a point (x(t), y(t)) defined as ,

$$a(t) = \frac{\Delta y(t) - \Delta x(t)}{\Delta y(t) + \Delta x(t)} \quad (8)$$

Aspect Ratio characterized the height-to-width ratio of the bounding box which contained the preceding and succeeding points of (x(t), y(t)).

Writing Direction : The writing direction of a point (x(t), y(t)) was defined by sine and cosine functions as,

$$\cos \alpha(t) = \frac{\Delta x(t)}{\Delta s(t)} \quad (9)$$

$$sin\alpha(t) = \frac{\Delta y(t)}{\Delta s(t)} \quad (10)$$

where,

$$\Delta s(t) = \sqrt{\Delta x^2(t) + \Delta y^2(t)} \quad (11)$$

$$\Delta x(t) = x(t-1) - x(t+1) \quad (12)$$

$$\Delta y(t) = y(t-1) - y(t+1) \quad (13)$$

Curvature: The representation of the curvature at a point (x(t), y(t))was done by using the cosine and sine of the angle defined by some points. Curvature would be 1/r, of a circle partially fitting the curve, with radius r. The curvature at a point (x(t), y(t)) was computed as,

$$\cos \beta(t) = cos\alpha(t-1) * \cos(t+1) + sin\alpha(t-1) * sin\alpha(t+1)$$
(14)

$$\sin \beta(t) = cos\alpha(t-1) * sin\alpha(t+1) + sin\alpha(t-1) * sin\alpha(t+1)$$
(15)

Curliness: Curliness (c(t)) was the feature that computed the deviation from a straight line in the vicinity of (x(t), y(t)). It was represented mathematically as the ratio of the length of the trajectory to the maximum side of the bounding box.

$$c(t) = \frac{l(t)}{\max(\Delta x(t), \Delta s(t))} (16)$$

Here, l (t) denoted the length of the trajectory i.e., the sum of lengths of all line segments.

Linearity: Linearity was the average square distance (d) between every point in the vicinity of (x(t), y(t)) and the straight line joining the first and last point.

$$l(t) = \frac{1}{N} * \sum d^2 \quad (17)$$

In [5], isolated Devanagari character is recognized for iPhone using HMM model. Devanagari script has no capital letters and written in left to right manner. The shapes of the Devanagari character are highly cursive. The similarity of some characters gives rise to the difficulty in the recognition of the Devanagari character. In this paper, the Devanagari alphabet which is divided into 13 vowels and 36 consonants, 42 stroke classes were created. Sub strokes were extracted from the strokes and then six scalar features were extracted

from those sub strokes. The recognition of each sub stroke was done using Hidden Markov classifier or HMM. The HMM is a set finite state machine that have a set of hidden states, output alphabet, transition probabilities, output probabilities and initial state probabilities. Each of the sample character contains within it various strokes which are subdivided into sub strokes. The feature vector of all these sub strokes is then converted into states. These states form the state sequences. From the state sequences, the initial state distribution and the state transition probabilities can be estimated. An iPhone application was developed to manually study the strokes thus enabling to create the stroke classes. A HMM was created for each stroke class, which is to be added in the final application.

In [6], online handwriting recognition of various Indian scripts was done. The online handwritten character databases of four Indian languages viz. Bangla, Devanagari, Tamil and Telugu were taken and recognized. The features extracted were point-float and direction code histogram features. The extracted features were recognized using three classifiers viz. Nearest Neighbor (NN), Multilayer Perception (MLP), and Hidden Markov Model (HMM). The efficiency of all the three classifiers were seen and result of each classifier was compared. A total of 23891 samples of Devanagari databases were provided by 109 writers, for Bangla writers, 25948 samples. There were 45217 samples written by 228 writers for Telugu and 77609 samples by 228 persons for Tamil. The recognition rate is found better in NN classifier for both the features than the other two classifiers. For point float feature and chain code feature the accuracy rate of NN classifier is shown in Table no 1 and Table no 2 respectively.

Table no 1: Recognition rates for point float features

Databases	Top 1	Top 2
Bangla	89.76	95.86
Devanagari	85.32	93.33
Telugu	91.16	98.04
Tamil	89.43	96.58

Table no 2: Recognition rates for chain code features

Databases	Top 1	Top 2
Bangla	91.01	97.78
Devanagari	86.01	95.25
Telugu	92.65	98.97
Tamil	92.71	98.52

In [7], research was carried out for a dataset of online handwritten Assamese characters. Assamese is a major script in the northeastern part of India. No online handwritten Assamese characters dataset is available in any standard online repository of datasets. The main objective behind this project was to build a dataset of online handwritten Assamese characters and extracting the different features recognize the labels using modeling techniques. In Assamese there are 10 numeric characters and 52 basic alphabetic characters that consist of 11 vowels and 41 consonants. An Assamese character has a large number of conjunct consonants (Juktakkhor) of about 164-201. There are certain vowels, consonants, and conjunct consonants that have headlines called matra in Assamese. The data was captured as text and written on a digitizer with an electronic pen and the pen-up/pen-down switching was sensed. Support Vector Machine classifier was used for the modeling. SVMs (Support Vector Machines) are techniques used for data classification.

The main objective behind the SVMs is to produce a model based on the training data, which estimates the target values of the test data. Polynomial kernel and Gaussian radial basis functions are the common kernel functions in SVM. Experiments were performed using these functions for the classification of both numerals and alphabets. The training and testing of the classifier was done using k-fold cross validation procedure. The original sample was partitioned into k equal size sub-samples. Out of the k subsamples, one subsample was retained as validation data for testing and remaining k-1 samples were used for training. The cross validation process was repeated k times after that and each of the subsamples were used exactly once as the validation data. The k results were combined together producing a single estimation.In this work a 10 cross validation was done. 2,340 characters were used as samples and the overall recognition rates achieved for the online handwritten Assamese basic alphabetic characters were 71.54% (using linear kernel), 75.39% (using polynomial kernel), and 78.38% (using RBF kernel). The overall recognition rate of 99.11% was achieved for the online handwritten Assamese numerals obtained by using the polynomial kernel with the kernel parameter settings C = 1, E = 4 and a 10 fold cross validation process. Here C is the complexity parameter and E is the exponent of the

polynomial kernel. Among the three SVM kernels used in the experiment for Assamese alphabetic characters, the RBF kernel gave the best recognition rate of 81.15%.

Figure 3: Architecture for the classification of characters[7]

Paper [8] describes the On-line Malayalam Handwritten Character Recognition using Wavelet transform and SFAM. Malayalam is the official language of Kerala. The Malayalam script consists of independent vowels, dependent vowel signs, consonant letters, consonant signs, consonant conjuncts and Chillu. The data acquired goes through various preprocessing steps before feature extraction. The preprocessing operations were duplicate point removal, smoothing, size normalization and equidistant resampling. The character is sampled into 26 sample points due to equidistant resampling. The preprocessed coordinates are then used for feature extraction which is a two-step process. At first, a total of six time domain features were extracted from the normalized (x, y) coordinates. The (x, y) coordinate features, angular features, direction and curvature were extracted. In the second step, the wavelet transform of the features were calculated to form the resultant compressed feature vector. Wavelet transform

uses simplified wavelets to represent signal in terms of approximation and detailed coefficients. Simple wavelets such as haar are sufficient to represent the simple features like the angular features with a satisfactory rate of accuracy. The main goal of the wavelet transform is to compress the individual extracted features so as to filter out only the relevant information in them thus compressing the size. In this paper db1 and haar wavelets are used. The recognition is done using Simplified Fuzzy ARTMAP(SFAM). 2530 samples of 63 symbols collected from 40 different users were used for training. 1279 character samples collected from 20 different users consisting of 20 symbols were used for testing. A maximum accuracy of 97.81% was achieved when haar wavelet was used. The recognition time is about 0.5sec/symbol. A maximum accuracy of 97.73% was achieved when db1 wavelet was used.

In [9], real-time on-line unconstrained handwriting was recognized using statistical methods. A general recognition system for large vocabulary, writer independent, unconstrained handwritten text was explained. The data were collected as set of (x, y) coordinates which were sampled at rates between 70 Hz and 100 Hz. The equispaced points were normalized to remove the jitters and inconsistencies. To limit the computation they used a single state model. But there cropped some issues like, the model did not provide duration modeling and there was no distinction seen between the beginning and end of the character. These issues were removed in the detailed match model in which each character was modeled by a series of states, each having an output distribution associated with it. The 20,000 above word lexicon was stored in a structure which merged some of the common prefixes and suffixes. Only those paths which correspond to valid words were expanded. A stack was associated with each frame that contained all possible partial paths ending at the frame. These stacks were then sorted by probability. Among all the elements, the top element of the final stack corresponded to the recognized string. Models were built for each significant variation of each character to capture the writing styles across different writers. An automatic procedure was used to identify the variations called lexemes. Approximately 150 lexemes were generated. Each HMM consisted of 6 states which resulted in 900 distinct states. Recognition performed using HMM modeling technique. 100,000 characters of data were collected from 100 writers. The error rate achieved is 18.9%.

In [10], A K-NN based On-Line Handwritten Character recognition system was described. Malayalam text was entered using pen-like devices. Malayalam is the official language of Kerala. The Malayalam script consists of independent vowels, dependent vowel signs, consonant letters, consonant signs, consonant conjuncts and Chillu.The raw data for this recognition experiment was collected using the device Wacom Graphire 4 CTE-640. A database of 40 writers consisting of 64 basic characters were collected and trained. The test was done based on writer dependent and writer independent strokes. The system consisted of preprocessing, feature extraction, training and recognition stages. Preprocessing was done to reduce unwanted noise, normalization and segmentation of the signal into meaningful units. Preprocessing consists of the steps viz. Dot detection, Dehooking, Smoothing, Thinning, Loop detection, Normalization and Equidistant Resampling. The preprocessed coordinate is further used for feature extraction. The features extracted were: Normalized x-y coordinates, Pen-up/pen-down, Aspect, and Curvature. The Nearest Neighbor Classifier (K-NN) is a type of classifier where the function is approximated locally and all computation are deferred till classification. In

pattern recognition, the k-nearest neighbors algorithm (k-NN) is the method to classify the objects based on closest training. The k-nearest neighbor algorithm is the simplest machine learning algorithms. The distance between each test and the entire training samples is calculated to determine its nearest neighbor list. The whole system was implemented in JAVA, on a 32-bit AMD Athlon 2.0 with 512MB RAM. 2560 total samples were collected from 40 people for the system. The system gave an accuracy of 98.125%.

In [11], a Neural Network Approach to Online Devanagari Handwritten Character Recognition system was implemented. The paper proposed a classifier to classify online Devanagari characters into one of 46 characters in the alphabet set. The feature extracted is just the Discrete Cosine Transform of the temporal sequence of the character points. The Discrete Cosine Transform was used for creating the feature vector for each sample. There is the utilization of DCT as feature extractor to (x, y) coordinate data. This data was generated along with Neural Network based classifier from the samples. DCT is not only a good feature extraction method but also is a better transform technique than most other transforms. Training was done with Artificial Neural Networks. One output neuron was set aside for each character, thus 46 neurons were used in the output layer. For the hidden layer configuration, the general concept that the performance of the system increases monotonically with increase in the number of neurons was adopted. When 40 hidden layer neurons were considered, the accuracy rate was over 90%. Then again 50 neurons gave an accuracy rate of more than 94.5%. But this monotonic pattern was broken when 60 hidden layer neurons were considered. After the experiments were performed, it is observed that the best recognition rate of 97.28% occurred when 3 neural networks were fused averaging 50 neurons per hidden layer. Thus with 2760 characters being tested, recognition rates of up to 97.2% were achieved.

In [12], comparison of Assamese Character Recognizer is done using Stroke Level and Character Level Engines. Assamese is the major language of Assam. There are total 11 vowels, 41 consonants and 10 numerals in the Assamese language. An Assamese character set consists of one or more strokes. There are two approaches used in this paper for character recognition using HMMs. In the first approach, HMMs were trained on stroke level data and characters were recognized by comparing stroke sequences generated by HMM. The untrained character was recognized by adding their stroke combination in the reference set. In the second approach, HMMs were directly built on character level data. Variations in the stroke order and sequences to write a character, led to the difficulty in updating the reference set. In that case the character based method was used as it was more efficient. The input to online handwriting recognition system was the pen up/pen down movement of the digital pen, i.e., the (x, y) coordinate. The sequence of coordinates thus obtained from one pen down to pen up is termed as stroke. The pen down to pen up information of all the characters present in the database were studied and based on the similarity, they were then grouped into 181 unique classes. A character could be written either in a single stroke or a combination of multiple strokes. After analyzing the Assamese character database, the maximum number of strokes that is required to write a character is set to eight. Based on this stroke combinations and analyzing degree of confusion between similar stokes, eight reference sets were generated. An HMM stroke classifier was developed for recognition by constructing one HMM for each class of stroke. When there is an unknown character input, the stroke classifier identified a sequence of strokes. This sequence of character was then checked in reference set to recognize the character. Here the different operations involved in preprocessing of (x, y) coordinates was described. The preprocessing steps were, size normalization, smoothing, duplicate points removal and resampling. Normalization was performed at stroke level thus eliminating the size variations in the characters of different writers. The stroke size was normalized to unit square. Noise captured due to noise and jitters while collecting the data was removed by smoothing. Multiple points with same (x, y) coordinates were removed by duplicate points removal. Again to compensate for the variation that occurs in the coordinates due to different speed of writing of different users, resampling was done where coordinates points were sampled equidistantly. In Assamese script only a few strokes can be written completely above the baseline or center line of a character. Some strokes in Assamese characters may lie above the baseline (Figure. 4). Figure. 5 shows some character examples with strokes above baseline. After preprocessing, features were extracted for recognition. Three set of features are considered. The first set was the preprocessed (x, y) coordinates of the resampled coordinate, second and third set were the first derivative and second derivative of the (x, y) coordinate, respectively. HMM technique is a widely used modeling technique because of the time sequential nature as well as for its capability of modeling shape variability in probabilistic terms.

A continuous density HMM was used for modeling each stroke. A set of 181 HMM models were built, one for each of the 181 strokes. Stroke level classifier was used to classify the characters in training set and the reference set was prepared. The stroke sequences for each character were added to the reference set. 141 total classes of Assamese characters were present in the database. HMM based modeling was done in character level to recognize a character, where each character was modeled using the HMM classifier. After the recognizer and the reference sets were combined, the stroke based character recognition system gave an average accuracy of 43.76% and the average accuracy of the character recognizer using character HMMs was obtained as 82.96%.
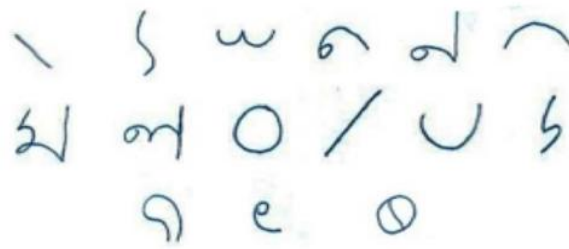

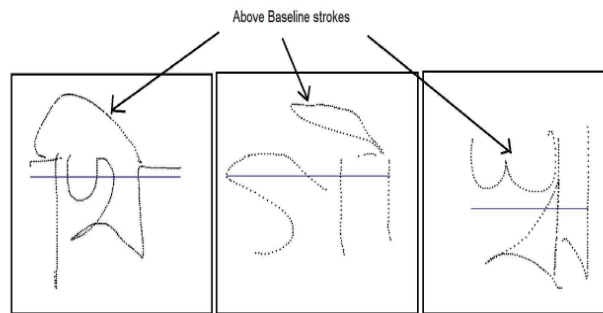**Figure 4**: Strokes that may lie above the baseline [12]

Figure 5: Character examples with strokes completely above baseline [12]

In [13], Arabic online handwriting recognition was done using Hidden Markov Model or HMM. The preprocessing was performed by removing the duplicate points, interpolation, smoothing, resampling and de-hooking. In this paper, a new preprocessing technique was introduced for handling the delayed strokes by rearranging them. After the rearrangement of the delayed strokes, features computed and extracted. The researchers extracted some of the best set of features, viz. chain code, curvature, aspect ratio, curliness, baselines and zones, loop detection, Hat feature and some extended features like, path tangent angle, path velocity magnitude, log-curvature radius and total acceleration magnitude. The modeling technique used was HMM modeling. Left to right HMM model was used. In accordance to how complex the model shape is, the different number of states per model was used. 23251 words were trained and 6671 words were tested. The accuracy using the same trained models were found to be 68.76% and then adapting the trained model using CMLLR technique the accuracy was found to be 87.47%.

In [14], a Hybrid Model for Recognition of Online Handwriting in Indian Scripts was discussed. A generic design for development of online handwriting recognizer for two Indian scripts, Malayalam and Telugu, was presented and its effectiveness was demonstrated. The basic unit of writing in Indian scripts is an akshara, combination of multiple consonants(C), followed by a vowel (V) i.e. CnV. Thus a word is formed using a sequence of aksharas and each akshara is formed from a set of basic characters and then each character is formed by a sequence of strokes.

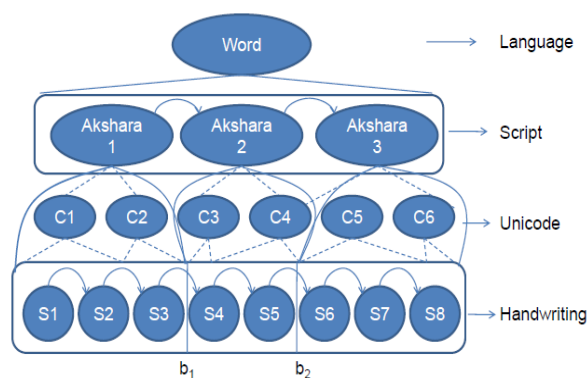**Figure 6**: Character 1 written using different number of strokes [12]

Figure 7: Handwriting formation from a word [14]

In the paper, explicit recognition of basic characters was avoided. Even though the language defines an akshara as a set of basic characters, the script was mapped from aksharas, directly to the strokes in the handwriting. The strokes were considered as a sequence emanated from the underlying sequence of aksharas in the word. This model led to a formulation where the stroke sequence was generated by a Markov process. The features used were: raw x and y co-ordinates of the resampled points, moments of the stroke up to fourth order, overall direction and curvature of the stroke, length of the stroke, aspect ratio, area of the stroke, number and direction of points in different sub-windows (55), projection histograms in X and Y directions, and Fourier coefficients of x and y sequences. Extensive feature selection was performed and various classifiers were employed for this purpose, and found that an SVM classifier using a Decision Directed Acyclic Graph (DDAG) formation for combining individual pair-wise classifiers to be the most effective classifier. A total of 7348 samples from 90 Malayalam strokes were used for training the system. The Telugu dataset comprised of 57,669 samples for training. The overall accuracy of Malayalam recognizer, when 60, 492 words were tested was found to be 78.07%. The data contained large variability in writing style and other errors. The isolated character recognition achieved an accuracy of 93.10% on a dataset of over 11, 000 characters collected from 125 writers. The accuracies on a similar-sized Telugu dataset were 75.70% on the word level.

In [15], the researchers describe the recognition of Online Assamese Stroke and Akshara Recognizer using HMM modeling. Strokes are composed of two coordinate traces in between pen down and pen up labels. A number of strokes combine to form the Assamese aksharas. The maximum number of strokes taken to form the combination is eight. Eight language rule models were made based on these combinations to test a set of strokes if they form a valid akshara or not. The features extracted were; preprocessed (x, y) coordinates and first and second derivatives of (x, y) coordinates. The window size considered was two. The stroke classifier was built using HMM model. The database was annotated was annotated at stroke level. Six dimensional features were extracted from the preprocessed coordinates. A stroke classifier with 181 HMM models were developed. During testing of the akshara, the strokes were at first preprocesserd and then the features were extracted. The extracted features were tested against the stroke classifier.
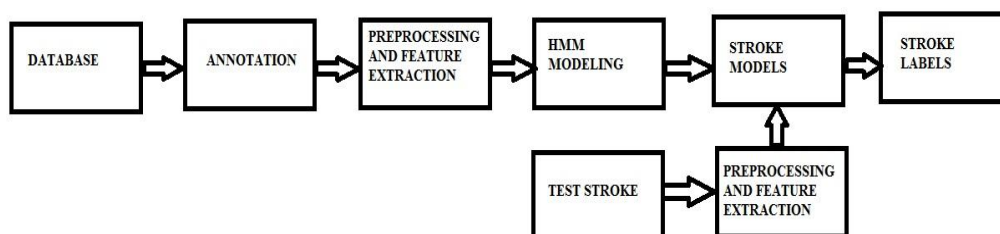

Figure 8: Stroke level classifier [15]

For recognition of the aksharas, the recognized stroke labels along with the language models were used. If all the strokes correctly recognized then only it isconsiderd as recognized akshara. The testing at akshara level was performed by integrating a GUI with the Binaries of HTK toolkit classifier. The stroke level performance was 94.14% and akshara level performance was 84.2%.
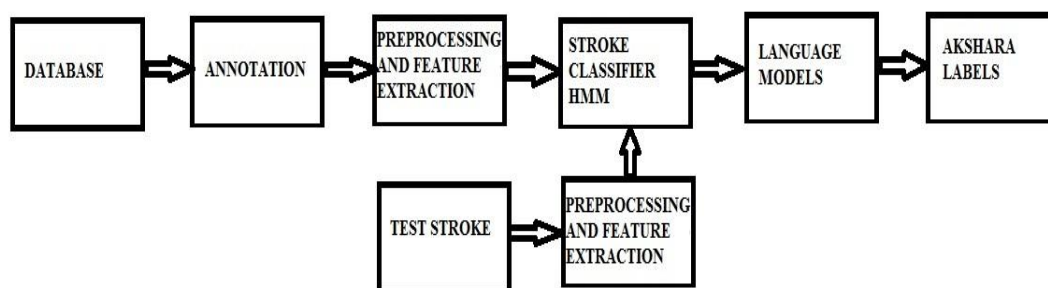
Figure 9: Akshara level classifier [15]

IV. Performance measures of existing methodologies

The summary of some papers for Indian scripts is tabulated in Table 3. The various databases used by the authors, the methods they used, the numbers of samples used for training and testing and the accuracy rate are tabulated. As it is seen various methods were carried out for recognition of online handwriting in Indian script and the performance is quite good in most of the methodologies used.

Table no 3: Performance comparison of the existing methodologies

Author Name	Method used	Year of publishing	Database	No. of samples	Accuracy obtained
Indhu T.R., Bhadran V.K.	SFAM	2012	Malayalam character, database	29 training users, 25 testing users	98.26%
Gernot A.Fink, Szilard Vajda, Ujjwal Bhattacharya, SwapanK.Parui, BidyutB.Chaudhuri	HMM	2010	Bangla character database	14073	93%
Deepjyoti Das, Rituparna Devi, SRM Prasanna, Subhankar Ghosh, Krishna Naik	HMM,SVM	2014	Assamese character database	203 isolated strokes, 147 aksharas	For strokes-HMM (94%), SVM (96%) For aksharas-HMM (84.67%), SVM (86.23%)
S.Jaeger, S. Manke, J. Reichert, A.Waibel	NPen++ recognizer	2000	CMU,UKA,MIT	5000,20000,50000	5000 words-96%, 20000 words-93.4% , 50000 words-91.2%
UdayanBaruah, Shyamanta Hazarika	SVM	2014	Assamese character database	8235	numeral-99.11% alphabets-81.15%
K.P. Primekumar, S. M. Idiculla	SFAM	2011	Malayalam character database	2530 training, 1279 testing users	Using HAAR wavelet-97.81% using db1-97.73%
M. Sreeraj, S. M. Idicula	k-NN	2010	Malayalam character database	2560	98.125%
H. Choudhury, S. Mandal, S. Devnath, S. R. M. Prasanna, S.Sundaram	HMM	2015	Assamese character database	200 training, 200 testing users	Stroke level-43.76%, Character level-83.96%
A. Arora, A. M. Namboodiri	SVM	2010	Malayalam and Telugu character database	Malayalam-60492 Telugu-57669	Malayalam word level-78.07% and isolated-93.10% Telugu word level-75.70%

V. Conclusion

Various researches has been carried out for Handwriting Character recognition from the past decades. In this paper, we are trying to identify an efficient online handwriting character recognition system for Indian script. This paper provides a detailed review about the various Handwriting Recognition systems and also discusses some of the recenttrends on this system. A comprehensive study for recognition of online handwritten words in Indian languages was done and the methods implemented were analyzed. Some researchers implemented various methods together to find out the best technique for online handwriting recognition. But there is still very limited work is done for Indian scripts compared to other languages. In recent works, new methods based on DNN (Deep Neural Network) are explored apart from the existing methodologies and the results are compared.

References

[1] T. R. Indhu and V. K. Bhadran, "Malayalam online handwriting recognition system: A simplified fuzzy ARTMAP approach", 2012 Annual IEEE India Conference (INDICON), Kochi, 2012 pp. 613-618.

[2] G. A. Fink, S. Vajda, U. Bhattacharya, S. K. Parui and B. B. Chaudhuri, "Online Bangla Word Recognition Using Sub-Stroke Level Features and Hidden Markov Models",2010 12th International Conference on Frontiers in Handwriting Recognition, Kolkata, 2010, pp. 393-398 .

[3] Deepjoy Das, RituparnaDevi,SRM Prasanna, Subhankar Ghosh, Krishna Naik. , "Performance Comparison of Online Handwriting Recognition System for Assamese Language Based on HMM and SVM Modelling", Computer Science and Information Technology,2014

[4] S.Jaeger, S. Manke, J. Reichert,A.Waibel , "Online handwriting recognition: the NPen++ recognizer", Interactive Systems Laboratories, University of Karlsruhe, Computer Science Department, Interactive Systems Laboratories, Carnegie Mellon University,School of Computer Science, Pittsburgh, USA, 2000

[5] Abhimanyu Kumar and SamitBhattacharya , "Online Devanagari isolated character recognition for the iPhone using Hidden Markov Models", 2010 IEEE Students Technology Symposium (TechSym), Kharagpur, 2010, pp. 300-304.

[6] T. Mondal, U. Bhattacharya, S. K. Parui, K. Das and D. Mandalapu, "On-line Handwriting Recognition of Indian Scripts - The First Benchmark", 2010 12th International Conference on Frontiers in Handwriting Recognition, Kolkata,2010, pp. 200-205.

[7] UdayanBaruah, Shyamanta Hazarika, "A Dataset of Online Handwritten Assamese Characters", . Journal of Information Processing Systems,2014

[8] K. P. Primekumar and S. M. Idiculla, "On-line Malayalam handwritten character recognition using wavelet transform and SFAM",3rd International Conference on Electronics Computer Technology, Kanyakumari,2011 pp. 49-53, 2011.

[9] K. S. Nathan, H. S. M. Beigi, JayashreeSubrahmonia, G. J. Clary and H. Maruyama, "Real-time on-line unconstrained handwriting recognition using statistical methods",," 1995 International Conference on Acoustics, Speech, and Signal Processing, Detroit, MI,1995, pp. 2619- 2622 vol.4.

[10] M. Sreeraj and S. M. Idicula, "k-NN Based On-Line Handwritten Character Recognition System", 2010 First International Conference on Integrated Intelligent Computing, Bangalore, pp. 171-176, 2010.

[11] S. Kubatur, M. Sid-Ahmed and M. Ahmadi, "A neural network approach to online Devanagari handwritten character recognition", 2012 International Conference on High Performance Computing and Simulation (HPCS), Madrid, pp. 209-214, 2012.

[12] H. Choudhury, S. Mandal, S. Devnath, S. R. M. Prasanna and S. Sundaram, "Comparision of Assamese Character Recognizer using Stroke Level Engines", Dept. of Electronics and Electrical Engg,IIT Guwahati,Guwahati,2015

[13] I. Abdelaziz, S.Abdou, Al-BarhamtoshyHassanin, "Large Vocabulary Arabic Online Handwriting Recognition System", Pattern Anal Applic, 2016

[14] A. Arora and A. M. Namboodiri, "A Hybrid Model for Recognition of Online Handwriting in Indian Scripts", 12th International Conference on Frontiers in Handwriting Recognition, Kolkata, pp. 433-438, 2010

[15] Deepjoy Das, RituparnaDevi,SRM Prasanna, Subhankar Ghosh, Krishna Naik. , "Online Stroke and Akshara Recognition GUI in Assamese language using HMM", International Journal of Scientific and Research Publications, Volume 4, Issue 1, January 2014

[16] K. Shashikiran, K. S. Prasad, R. Kunwar and A. G. Ramakrishnan, "Comparison of HMM and SDTW for Tamil handwritten character recognition", International Conference on Signal Processing and Communications (SPCOM), Bangalore, pp. 1-4, 2010

[17] S. A. Azeem and H. Ahmed, "Combining online and offline systems for Arabic handwriting recognition" Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, pp. 3725-3728, 2012

[18] V. L. Lajish and S. K. Kopparapu , "Fuzzy Directional Features for unconstrained on-line Devanagari handwriting recognition", 2010 National Conference On Communications (NCC), Chennai, pp. 1-5, 2010.

[19] S. Prum, M. Visani and J. Ogier, "Cursive On-line Handwriting Word Recognition Using a Bi-character Model for Large Lexicon Applications," 2010 12th International Conference on Frontiers in Handwriting Recognition, Kolkata, pp. 194-199, 2010.

[20] I. Khodadad, M. Sid-Ahmed and E. Abdel-Raheem, "Online Arabic/ Persian character recognition using neural network classifier and DCT features," 2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS), Seoul, pp. 1-4, 2011.

[21] S. A. Azeem and H. Ahmed, "Recognition of Segmented Online Arabic Handwritten Characters of the ADAB Database," 2011 10th International Conference on Machine Learning and Applications and Workshops, Honolulu, HI, pp. 204-207, 2011.

[22] I. Hosny, S. Abdou and A. Fahmy, "Using advanced Hidden Markov Models for online Arabic handwriting recognition," The First Asian Conference on Pattern Recognition, Beijing, pp.565-569, 2011.

[23] N. Hajj and M. Awad, "Isolated handwriting recognition via multistage Support Vector Machines," 2012 6th IEEE International Conference Intelligent Systems, Sofia, pp. 152-157, 2012.

[24] M. A. Abuzaraida, A. M. Zeki and A. M. Zeki, "Feature extraction techniques of online handwriting arabic text recognition," 2013 5th International Conference on Information and Communication Technology for the Muslim World (ICT4M), Rabat, pp. 1-7, 2013.

[25] S. D. Chowdhury, U. Bhattacharya and S. K. Parui, "Online Handwriting Recognition Using Levenshtein Distance Metric," 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, pp. 79-83, 2013.

[26] Claus Bahlmann, "Directional features in online handwriting recognition", Pattern Recognition, Volume 39, Issue 1, 2006.

[27] Y. Yamaji, T. Shibata and Y. Tonouchi, "Online Handwritten Stroke Type Determination Using Descriptors Based on Spatially and Temporally Neighboring Strokes," 2014 14th International Conference on Frontiers in Handwriting Recognition, Heraklion, pp. 116-121, 2014.

[28] Santosh K.C., CholwichNattee, "A comprehensive survey on online handwriting recognition technology and its real application to the Nepalese natural handwriting," Kathmandu University Journal of Science, Engineering, and Technology, Kathmandu University, pp.31- 55,2009