International Journal of Scientific & Engineering Research, Volume 7, Issue 4, April-2016 475
ISSN 2229-5518

Online handwriting recognition systems

K.A.K.N.D. Dharmapala

Abstract— Dramatic increase of computational devices has increased the use of input methods that are not natural to the human. There are many occasions where handwriting is preferred over making entries with a keyboard. On-line handwriting recognition is a trending and highly anticipated area of research where it comprises of natural textual inputs through handwriting rather than using a keyboard. These systems take inputs through the digitizer from the user and then process those data and convert them to characters which can be recognized by a computer. This paper focuses on the different systems that are present, their differences and issues that are related to them. It is concentrated on the general process of a handwriting recognition system and the processes that are incorporated in order to make the recognition more fast and accurate.

Index Terms— On-line handwriting recognition, real-time character recognition

1 Introduction

The technological advancements at present have led to the discovery of new methods of human and computer interaction. In a typical scenario, textual inputs to a computer system are usually made through a keyboard where the user has to press each and every key. Online Handwriting recognition systems can be used to avoid this and make the textual inputs a whole lot easier. This would enhance the user experience by converting hand written characters into computer recognized text.

Online handwriting recognition involves the conversion of text as the user writes on a special digitizer or a writing surface and the user inputs are recognized and converted into letter codes which can be used in computer systems. In modern smartphones and laptop computers a touch sensitive screen is present which enables to use online handwriting recognition systems rather than depending on the keypad [1]. This enables the widespread use of online handwriting recognition systems enhancing the interaction between computer systems and the human.

2 Background

2.1 Online vs offline handwriting

Online handwriting recognition suggests that the system recognizes the writing or the inputs through the digitizer in real time. The translation of user inputs to the characters which can be understood by the computer system is performed dynamically. Depending on the technology and the performance of the system that is used, the speed at which the written characters are recognized by the system varies. With current advancements of microprocessor technologies and algorithms these performance requirements can be easily achieved.

In contrast, off-line handwriting is performed after the writing is done where an optical scanner converts the writing to an image which can then be processed by the computer system. This image is then processed and converted into characters that can be recognized by the computer system.

On-line and off-line handwriting are different in many aspects. In on-line handwriting recognition, the machine data are being captured as the user makes the input. These inputs include lots of data attributes compared with off-line recognition. Because the data capture is done in real time attributes such as pen pressure, direction of the pen movement, etc. can be captured. This temporal information of on-line system complicates the character recognition which is not present in a static image used for an offline system.

2.2 Digitizer Technology

In order to perform on-line handwriting recognition, it requires special digitizer hardware to pick up the handwriting of the user. A digitizer or a pointing device can perform 6 kinds of tasks and those are select, position, orient, path, quantify and text input [2]. There are a number of tablet digitizer technologies. But the most popular two technologies are electromagnetic/electrostatic tablets and pressure sensitive tablets.

Electromagnetic/electrostatic tablets have a two-dimensional grid of conductors spaced very closely apart. When the grids of conductors are electromagnetically exited due to the capacitivity of the fingertip or the stylus, the induced voltage enables the device to determine the precise location of interaction. In addition to that pressure sensitive tablets use two layers of conductive and resistive material which are physically separated. Once a pressure is applied, the conductive layer picks the voltage from the resistive later and the position of interaction is calculated against the voltage level. In addition to that there are other digitizer technologies which triangulate reflected laser beams from an optical sensing light pen.

To maintain the accuracy in on-line handwriting recognition, the digitizer needs to have an acceptable level of requirements. Usually a resolution of 200 points per inch and a sampling rate of 100 samples per second [3] provide an acceptable level of data for the recognition system.

3 On-line handwriting recognition system

The process of on-line handwriting recognition can be broken down into the following steps. The data gathered from the digitizer hardware is processed and categorized to generate computer recognizable characters.

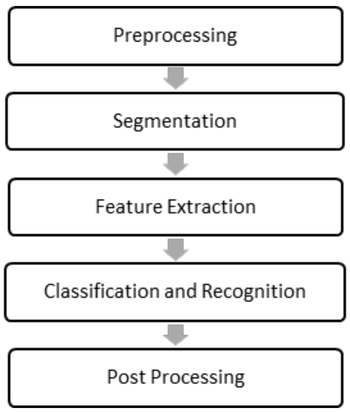

Figure 1: Flow diagram of an on-line handwriting recognition system.

3.1 Preprocessing

The preprocessing of handwriting is a series of operations performed on handwriting data prior to applying recognition algorithms. This stage of system is determinant about cleaning and smoothing the strokes. There are a set of tasks which are performed on handwriting data at the preprocessing stage. The purpose of this stage is to address the problems of data reduction, elimination of imperfections and normalization and produce a set of data that is related to an image which is suitable for segmentation.

3.2 Segmentation

In this stage the data set is isolated into various writing units such as characters or words prior to recognition. In the early days the sequence of characters is decomposed into sub sets by means of an explicit signal from the user. The x-coordinate information and their projection on x axis were used to determine the writing units. Other early attempts used temporal information to determine the writing unit. The timing difference between the end of a stroke and the beginning of the next stroke [4] determines if it is a writing unit or not. Another approach is to write characters in predefined boxes where the segmentation is mostly done by the writer.

But in some languages writing units are not spaced and cursively written. Recent spatial segmentation techniques checks for two-dimensional separation of the writing unit. Use of spatial, temporal and other information enhances the accuracy of segmenting the input into writing units.

3.3 Feature Extraction

At this stage, features of the segmented writing units that are essential to classify them at the recognition stage are extracted. These features are based on either dynamic properties, static properties or both. The features can be broadly categorized into binary and non-binary features. For an example in binary features, the presence of a dot or the absence of the dot can determine the difference between lowercase 'i' and 'l'. Some of the features of a character set can be non-binary. A fixed number of non-binary features are considered in handwriting recognition systems that divide the character space into decision regions based on their features.

Another method of feature extraction is curve matching [5] which is almost equivalent to pattern matching where the unknown strokes are matched against a character from the knowledge base. The best match from the knowledge is assigned to the unknown strokes.

3.4 Classification & Recognition

The features extracted in the previous stage are used in this stage to make the decision by classifying and recognizing the character. This stage can be implemented in different ways. In languages where cursive writing is employed, there might be a requirement of recognizing the characters before the segmentation stage. Once this stage is provided with the data set from the feature extraction stage, those data can be fed to a back propagation neural network or some other knowledge based decision making system to perform the recognition and classification.

3.5 Post Processing

Post processing is performed on the recognized output. Computer recognizable characters which are chosen in the decision making process are then transformed into output. The accuracy of the post processor can be enhanced with language information. When the inputs to this stage is to generate different result than what user expects, some degree of language information such as number of letter in that particular word, spellings can fix the errors up to a certain degree.

4 Preprocessing

On-line handwriting recognition systems use the preprocessing technique as a method to simplify the task of shape recognition algorithm which increases the performance and accuracy of the recognition. Preprocessing is done in three major steps.

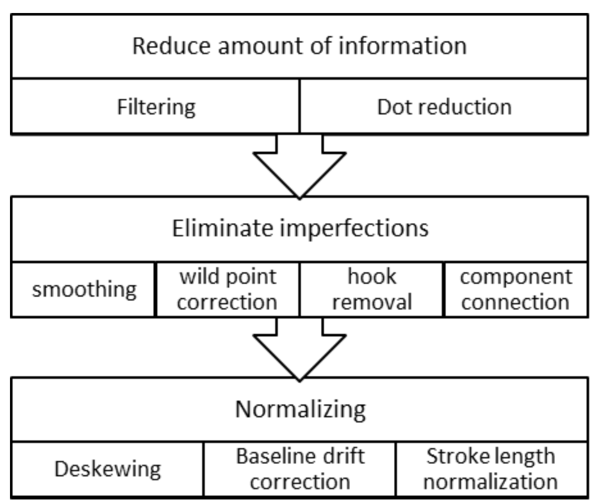

Figure 2: Diagram of techniques used in preprocessing

The first step is to reduce the amount of information using dot reduction and filtering techniques. Then imperfections are eliminated by smoothing, wild point correction, hook removal and component connection. Normalization of hand writing is done as the last step of preprocessing which includes deskewing, baseline drift correction, size and component length normalization.

4.1 Reduction of information

Digitizer devices send data about the pen tip position and other data at a fixed sampling frequency. These data are reduced by eliminating duplicate points and redundant data. This reduces the amount of data which needs to be processed in the later steps of recognition process. Filtering and dot reduction techniques are used in this stage. Both of these operations can be performed simultaneously.

Filtering which is also known as thinning is used in “eliminating consecutive points spaced by an interval under a certain threshold” [6]. In events where there is a curvature in the character, it avoids eliminating significant points which may end up in confusions in distinguishing between characters like ‘U’ and ‘V’. In some scenarios interpolation can be applied to get equally spaced points.

Dot reduction employs using special filters to condense dots to a single point. These dots are recognized by their angular variations and there position relative to one another. Dot reduction needs to be performed delicately in order to maintain the distinction between characters like ‘:’ and ‘;’.

4.2 Eliminate imperfections

Imperfections can be widely observed which occurs due to issues in digitizer hardware, erratic hand motion and inaccuracies of pen down/up indications. Smoothing, wild point correction, hook removal and component connection are the techniques which are most commonly used techniques to eliminate imperfections.

Smoothing reduces the imperfections occurred due to erratic hand motion or digitizer hardware issues. Smoothing is averaging the position of point Pi(xi, yi) with respect to its neighbors. The coefficient and the number of neighbors determine the filter type, its’ order and the frequency band. This computation can be performed at a very high pace and is performed at data acquisition step. Smoothing reduces the angular variations and too much of smoothening can result in the same result as in filtering by reducing the distinction between letters like ‘V’ and ‘U’.

Wild point reduction is detecting and correcting occasional spurious points occurred due to hardware issues of the digitizer devices. Recent improvements of hardware devices have drastically reduced the chances of this kind of imperfections. High velocity variations and thresholds that exceed hand motions can be used to detect wild points and correct them.

Dehooking is performed to remove hooks that are at the ends and beginnings of written elements which occur due to erratic hand motion and poor pen-digitizer (pen-paper) contact during pen-up and pen-down moments. These hooks are detected by their smallness and high angular variation and then the points are eliminated.

Stroke connection is categorized into two types of imperfections. “The first problem, described by Mandler and Ward and Kuklinski involves straight lines at both ends of a component, usually retracing themselves in a significant way” [7]. This can be detected by high velocity variations and small angular variations of the points within the component. “The second problem, described by Brown and Ganapathy, is usually the result of inaccuracies in pen-down detection” [7]. This can be identified by the angular continuity and the shortness between successive points. Commonly this imperfection can be observed in occasions where character size is small compared to the strokes that build up the character.

4.3 Preprocessing

This stage reduces the variations of handwriting and tries to generate more standard set of inputs for recognition algorithms. This stage involves techniques for baseline drift correction, deskewing and normalization of size and component length.

Baseline drift correction is aimed to bring the writings to a horizontal plain or a baseline. This step is crucial for handwriting recognition where the user inputs may not be horizontal to the level. This affects the overall efficiency and accuracy in later progressions of preprocessing stage such as deskewing and size normalization as well as in the stages of segmentation and shape recognition itself.

Stroke length normalization adjusts the size of the characters and strokes to a standard size resulting in easier segmentation and recognition processes. There are several approaches to the normalization process. One of the approaches is to normalize isolated characters with comparison to each other. Another approach is to normalize the length of strokes based on a specified number of points which are determined by filtering/interpolation.

Deskewing algorithms are used to measure and correct the slant of characters. There are several approaches to detect the slant of characters prior to correcting them. “R.M. Bozinovic and S.N. Shriari measured the local slant in regions that present low pixel density projected on the Y-axis. The global slant is the average of local slants. M.K. Brown and S.Ganapathy measured the local slant of central regions of the word. D.J. Burr used a very different technique, based on kinematics. A relation between the y and x velocities was used to measure the global slant of writing” [7].

5 Shape recognition

Shapes recognition is related to recognizing the shapes of the base writing unit. Shape recognition of characters can be discussed under the two broad topics of recognition of characters and cursive scripts. There are many methods that can be used to extract features of writing units.

5.1 Feature Analysis

Given that a set of features can represent a character; this method relies on static and dynamic properties of the character. These feature recognition, integrated with a decision tree can be used to recognize unknown characters.

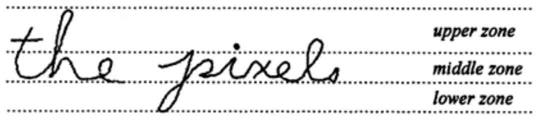

Figure 3: Illustration of lower, middle and upper regions of characters Source: [13]

For an example, letters “a”, “c”, “e” are completely in the middle zone while letters such as “b”, “d” are extended to upper zone as well. Characters like “i”, “j” have the binary property of having a dot. [13] Apart from that non-binary features can also be divided into decision space by a feature space of Fourier coefficients.

This method does not produce alternate character choices and requires highly accurate preprocessed data.

5.2 Time Sequence of Zones, Directions or Extremes

In this method, the character is represented by a sequence of zones that is specified by dividing the rectangular area which surrounds the character. Then the character is superimposed on to the rectangle and the zones are determined with respect to pen tip movement.

This method primarily relies on dynamic data where pen tip position is calculated against time. The superimposed character provides with a set of on and off pixels, whose center of gravity is calculated to rearrange and shift the x and y coordinates of the on-pixels. [12] The radial distances of all the on-pixels are calculated and normalized then normalized by the average radial distance which can then be matched against the database.

5.3 Curve Matching

This method is used to match unknowns against a set of known prototype characters in a database. Then the character which is represented by the best matching prototype is assigned to the unknown character. The x, y coordinates of the character is recorded as a function of the time. The character may be divided into multiple time regions depending upon the usage of curved strokes in the characters. Chinese characters that consists mostly of straight strokes require a little as six time regions while English characters require eight to ten time regions render successful [9].

Fourier coefficients obtained from respective coordinates of the curve can also be used as an alternative to the matching function of time. This method is effective in characters that consist of curved strokes which can be represented by a less number of Fourier coefficients [6].

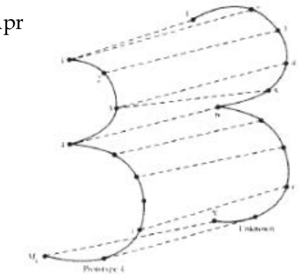

Figure 4: Illustration of elastic matching Source: [6]

In practical scenarios a matching linear alignment cannot be obtained with the points of the curve. Therefore elastic matching technique is applied where the bound lines can have a certain slope to represent the compressions and expansions of the match between the prototype and the unknown.

5.4 Stroke Code

In this method the characters are classified into subparts. This classification of subparts is done based on the slope differences of each of those subparts of the character. Then decision tree of stroke code sequences are used to classify the character under relative constraints on the subpart position.

Figure 5: Sample stroke code
Source: [6]

5.5 Analysis by Synthesis

This technique is also known as recognition-by-generation. Analysis-by-synthesis model uses character segments and a set of rules for connecting them to build up the character [6]. The idea is that the character is built from an inventory of segments. If a given unknown character can be created from the segments from the database, then the character which is mostly associated with the segments of a single character is attributed to the unknown.

5.6 Recognition of Cursive Scripts

In most of the cases where cursive scripts are employed as the common way of daily writing, character segmentation cannot be done independently of the character recognition. There are two approaches to address this issue. One of the approaches is the use of analysis-by-synthesis method to break the word in to subparts and recognize separate characters. The other approach is the whole-word approach to eliminate the segmentation problem entirely which is effective when the

ISSN 2229-5518

number of words to be recognized is small.

In the method of recognizing separate characters, stroke segments have to be identified which can then be used on analysis-by-synthesis methods. The most common approach of stroke segmentation is obtaining the substrokes at the point of minimum velocity. The writing units consist of upstrokes and downstrokes, whose character information in cursive writing is in the downstrokes while the upstrokes act as ligates characters [6]. The word can then be analyzed letter-by-letter.

6 COMPARISON OF CLASSIFICATION METHODOLOGIES

Once a character's features are identified there are many approaches to classify the character attributing a known character to the unknown writing unit. Most commonly used algorithms can be divided into the three categories of knowledge/heuristic based approach, global feature vector matching and structure based methods. All these categories consist many algorithms and methodologies to classify extracted features from handwritten inputs. Each of these methods has their advantages and disadvantages.

Knowledge/Heuristic based approach commonly uses 'Decision Trees' and 'Fuzzy logic' based solutions. When applying fuzzy values, each value is calculated with respect to the universe of discourse of the characteristics of the segmented component. If a certain characteristic has its' values within a certain range then the characteristic is matched against a database which contain information about each character, the number of segments and the characteristics of each of the segments to determine the character [8] .

When applying decision trees to recognize characters, the idea behind is to divide the segmented components recursively based the properties of a character in the most efficient way possible. This allows predictions based on known data by reducing the solution domain in each recursion. These features can be either binary or non-binary. For an example the presence of descenders, the presence of dot in letters like 'j' can be considered as binary features. Use of binary decision trees reduces the solution domain for succeeding analysis of more complex feature [9]. There are some non-binary features which can be incorporated into a binary tree based upon pattern recognition and classification methods as well. This method is not suitable for large sets of characters and it may find it difficul

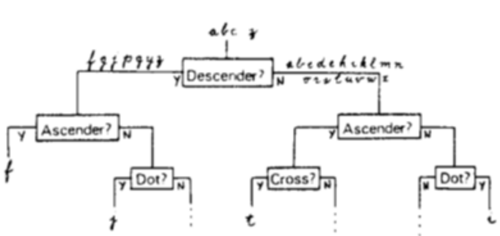

Figure 6: Decision Tree
Source: [6]

Global Feature Vector Matching is another approach of classifying features extracted from handwritten characters. This approach uses many recognition algorithms. Among them are Kohonen Neural Network and Feed Forward (unidirectional) Neural Networks. These methods are somewhat same in their functionality with minor changes of performance, efficiency and accuracy. The functionality of a Kohonen neural network is described here to get an understanding on how the characters are recognized based on their features using a neural network. The Kohonen neural network has input and output layers. Once normalized inputs are given the outputs values are calculated based on the input vector/pattern and neuron connection weights. One of the output neurons are selected as the winning neuron and its' weights are adjusted to react more strongly the next time so that it recognizes that particular pattern better in the future [10]. The resulted wining neuron can be used to recognize the input component.

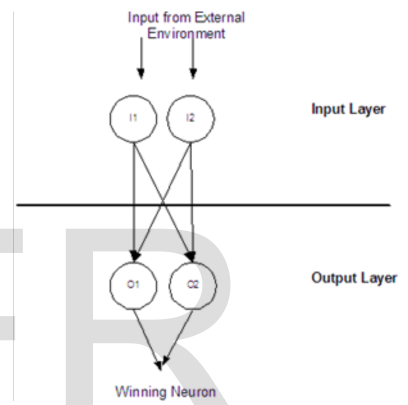

Figure 2: Kohon NN with two inputs
Source: [14]

The Kohon neural network differs from feed forward neural network in certain aspects. The Kohohen neural network does not use an activation function to determine whether to fire the neuron or not. The Kohohen neural network has only one output neuron where the feed forward neural network has multiple outputs from several output neurons. When compared Kohohen neural networks are relatively simple and can be trained rapidly to obtain better accuracy and efficiency. Noisy or cursive input scripts can be difficult to tackle in this approach.

Structure Based Method is another approach to classify and recognize handwritten inputs. One of the most commonly used techniques is using a hybrid of a neural network and hidden markov model. A Markov process is a randomly determined process whose future behavior is only dependent on its' present state. Six-element feature vector is calculated for each data point in typical application of HMM. These features are

- delta x position,
- delta y position,
- writing angle (ratio between delta x and delta y),
- delta writing angle (difference between the writing angle of the data point P_n and P_{n-1}),
- PenUp/PenDown bit (determine wether the data point is artificially generated or written by the writer), s

ISSN 2229-5518

- sgn(x-max(x)) (weather the current x position is greater or less than all the preceding data points)

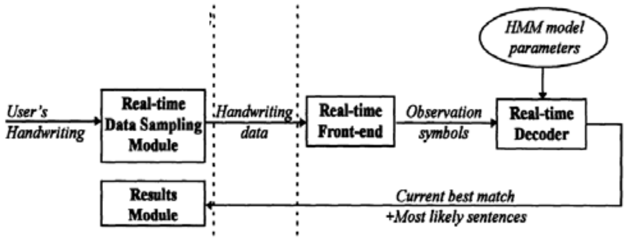

Figure 3 Online Handwriting System using HMM
Source: [13]

These feature vectors may differ slightly from system to system. But in a common scenario local topological features along the pen trajectory are extracted and represented as a sequence of time ordered observations $x_1..x_n$ where $Xi \in V$ for $i = 1 \dots n$. When it comes to recognition it consists of identifying the state sequence $y_1..y_{n+1}$ where $Yi \in S$ for $i = 1 \dots n$, is a concept class of a character feature or a character itself as it is calculated in formula 1. The joint distribution of state sequences and observations will be as follows;

$$p(x_1..x_n, y_1..y_{n+1}) = \prod q(y_i|y_{i-1}, y_{i-1}) \prod e(x_i|y_i)$$

Formula 1: Joint Distribution of Sequences

The most likely state sequence for the sequence of observations $x_1..x_n$ would be;

$$\max \ p(x_1..x_n, y_1..y_{n+1})$$

Formula 2: State Sequence of Observations

The result of this would be the character feature or the character itself with most probability of highest match [11]. This model is heavily used in handwriting recognition systems. This has the advantage of being able to integrate segmentation and recognition where conventional handwriting recognition algorithms may find it difficult to identify noisy or cursive scripts.

7 DISCUSSION

Online handwriting recognition systems recognize the writing or the input through the digitizer in real time. This topic discusses about a novel research area in computer-human interaction context. Hence still most of the systems do not provide a very high level of accuracy there are a lot of researches and algorithm optimizations are commenced in order to improve the accuracy and the efficiency. There are a number of methods and techniques that are employed in these systems.

I have discussed the process of online handwriting recognition in five major steps; preprocessing, segmentation, feature extraction, classification & recognition and post processing. I have organized the review paper on that concept of major steps involved in online handwriting recognition systems. Some of these major steps are further discussed explaining the most commonly used techniques. Preprocessing of handwritten data is determinant about reduction of information, eliminating imperfections and normalization which is achieved through a number of techniques. Segmentation and feature extraction is determinant about segmenting the handwritten data into units of identification which is characters in most of the systems. Then the features of the characters are extracted which is required to identify the character in unique. Classification & recognition step attributes a known character(s) to the writing unit based on extracted features.

Online handwriting recognition systems needs to be further improved to recognize more languages that are commonly used. Some of the widely used languages such as Mandarin need to be improved for a better level of accuracy and efficiency which is hard to acquire because of the complexity of the characters and the size of the alphabet unlike in English.

8 CONCLUSION

The process of on-line handwriting recognition can be broken down into the following steps. The data gathered from the digitizer hardware is processed and categorized to generate computer recognizable characters.

ACKNOWLEDGMENT

I would like to pay my sincere gratitude to supervisor of Independent studies, Dr. L. Ranathunga, Head of the Department of Information Technology, for his guidance and valuable suggestions which inspired me on this accomplishment.

REFERENCES

[1] Andreas Holzinger & Lamija Basic, B. P. (n.d.). "Handwriting Recognition On Mobile Devices, State of the art technology usability and business analysis."

[2] J. D. Foley, V. L. Wallace, and P. Chan. "The human factors of computer graphics interaction techniques," IEEE Comput. Graphics Appl., vol. 4.

[3] J. Kim and C. C. Tappen, "Handwriting recognition accuracy versustablet resolution and sampling rate," in Proc. 7th Inr. Conf. Pattern Recognition, 1984,

[4] A. M. Day, J. R. Parks, and P. J. Pobgee, "One-line written input to cornmiters." in Machine Perception of Pictures and Patterns.
London, England: Inst. Phys., 1972.

[5] K. Ikeda, T. Yamamura, Y. Mitamura, S. Fujiwara, Y. Tominaga, and T. Kiyono, "On-line recognition of hand-written characters utilizing positional and stroke vector sequences," in Proc. 4th Int. Joint Con& Pattern Recognition, Nov. 1978,

[6] C.C. Tappert, Adaptive on-line handwriting recognition, IEEE 7th Int. Conf. on Pattern Recognition, pp.1004-1007.

[7] Normalizing And Restoring On-Line Handwriting Wacef Guerfali And Rejean Plamondon, Pattern Recognition, Vol. 26, No. 3, pp. 419 431, 1993

[8] Romesh Ranawana, Vasile Palade, and G.E.M.D.C. Bandara "An Efficient Fuzzy Method for Handwritten CharacterRecognition"

[9] Charles C. Tappert,Ching Y. Suen,Toru Wakahara "The State of the Art in On-Line Handwriting Recognition", IEEE Transciions On Pattern Analysis And Machine Lntelligence Vol. 12, No. 8, August 1990

[10] Deepali Thombre, Toran Verma "Neural Network Approach to Recognize Online Handwriting Script"

[11] Giovanni Seni "Large Vocabulary Recognition of On-Line Handwritten Cursive Words ", August 1995

ISSN 2229-5518
[12] Joarder Kamruzzaman, S. M. Aziz "A Neural Network Based Character Recognition System Using Doublebackpropagation", Malaysian Journal of Computer Science, Vol. 11 No. 1, June 1998
[13] Han Shu S.B., "On-line Handwriting Recognition Using Hidden Markov Model", February, 1997
[14] Deepali Thombre, Toran Verma "Neural Network Approach to Recognize Online Handwriting Script"