A comprehensive survey on on-line handwriting recognition technology and its real application to the Nepalese natural handwriting

Santosh K.C., Cholwich Nattee

A COMPREHENSIVE SURVEY ON ON-LINE HANDWRITING RECOGNITION TECHNOLOGY AND ITS REAL APPLICATION TO THE NEPALESE NATURAL HANDWRITING

[1]Santosh K.C.*,   [2]Cholwich Nattee

[1]Institut National de Recherche en Informatique et en Automatique (INRIA), LORIA - Campus Scientifique -BP 239 - 54506 Vandoeuvre-les-Nancy Cedex, France

[2]School of Information and Computer Technology, Sirindhorn International Institute of Technology, Thammasat University, Thailand

*Corresponding author: Santosh.KC@loria.fr

ABSTRACT

Handwriting Recognition Technology has been improving much under the purview of pattern recognition and image processing since a few decades. This paper focuses on the comprehensive survey on on-line handwriting recognition system along with the real application by taking Nepali natural handwriting (a real example of one of the cursive handwritings). The survey mainly includes pre-processing, feature vector and similarity measures in between the non-linear 2D sequences of coordinates, and their effective applications. A very highlighting topic ``Dynamic Time Warping Algorithm'' (DTW) is introduced, which has been popular in determining the distance between two non-linear sequences ranging from handwriting to speech recognition. Besides these contemporary research issues/areas, stroke number and order free Nepalese natural handwritten recognition system is presented in the second step.

Writing one's own style brings unevenness in writing units, which is the most difficult part to classify. Writing units reveal number, shape, size, order of stroke, and speed in writing. Variation in the number of strokes, their order, shapes and sizes, tilting angles and similarities among characters from one another are the important factors, which are to be considered in classification for Nepali. This paper utilizes structural properties of those alphanumeric characters, which have variable writing units. It uses a string of pen tip's positions and tangent angles of every consecutive point as a feature vector sequence of a stroke. We constructed a prototype recognizer that uses the DTW algorithm to align handwritten strokes with stored strokes' templates and determine their similarity. Separate system is trained for original and preprocessed writing samples and achieved recognition rates of 85.87% and 88.59% respectively. This introduces novel real time handwriting recognition on Nepalese alphanumeric characters, which are independent of number of strokes, as well as their order.

Key Words: Handwriting Recognition System, Pre-processing, Feature Vector, Dynamic Time Warping, Agglomerating Hierarchical Clustering, Nepali.

INTORDUCTION

What is handwriting recognition?
Human eye can see and read what is written/displayed either in natural handwriting or in printed format. The same work in case the machine does is called handwriting

recognition, which is the common definition. Handwriting recognition has been a popular area of research since few decades under the purview of pattern recognition and image processing. Handwriting recognition can be broken down into two categories: off-line and on-line.

• Off-line
Off-line character recognition takes a raster image from a scanner (scanned images of the paper documents), digital camera or other digital input sources. The image is binarized through threshold technique based on the color pattern (color or gray scale), so that the image pixels are either 1 or 0.

• On-line
In on-line, the current information is presented to the system and recognition (of character or word) is carried out at the same time. Basically, it accepts a string of (x, y) coordinate pairs from an electronic pen touching a pressure sensitive digital tablet.

Once the image is binarized in off-line case, the rest of the techniques for classification can be identical with only two basic differences. Firstly, off-line recognition happens after the writing completes and the scanned image is pre-processed. Secondly, it has no temporal information associated with the image due to which it is not known to the classifier about the way and order of writing. So, we can say, its knowledge about the character is limited. It means, off-line data only represents the final result after writing i.e. an image. Why does the global market demand follow the on-line recognition system? On-line handwriting recognition system, by contrast, captures the temporal or dynamic information of the writing, enhances the accuracy over off-line. Another advantage is interactivity, which means recognition errors can be corrected immediately with the series of test. Yet, adaptation of any drawings of character is also an advantage over off-line. When the user faces that some characters are not recognized accurately, user can alter his way of drawing until it recognizes. It means user can adapt to the machine. Conversely, recognizers are capable of adapting users' drawing, usually by storing possible samples from a large number of users for subsequent recognition. Thus, there is adaptation of user to machine and of machine to user. Electronic pen input is the direct method to compare with the both off-line and key-board entry to the system having recognition intelligence. In addition, on-line recognition improves the work-flow, the information is immediately available. However, the natural and comfortable style in writing effectively reduces difficulty at the threshold of using computers for common users. Moreover, it is recently showed that handwriting input is the most acceptable and welcomed input style.

A handwriting recognition system can further be broken down into two categories of writer independent and writer dependent.

• Writer Independent and Writer Dependent
A writer independent recognition system recognizes wide ranges of possible writing styles, while a writer dependent recognition system is trained to recognize only from specific users. Therefore, a writer dependent recognition system works on data with a smaller variability and therefore a chance of having higher reliability is achieved in contrast to writer independent recognition system. Writing one's style brings unevenness in writing units, which is the most difficult part. Variability in stroke numbers, their order, shape and size, tilting angle and similarity among characters from one another are found more often in writer independent recognition system. Broadly, there are two kinds of

writing styles. They are hand printed and cursive handwriting. In cursive style, strokes are deliberately linked forming one from many to draw the character, while in hand printed style possible number of strokes are used, each stroke has significant role to complete the character. In cursive style, the important information such as intersections, loops, curves, straight lines and hooks etc. are missing. Some times, both writing styles are mixed. Natural handwriting contains all types of styles in writing from any of the users. Specifically, the writing is said to be natural as if users write on a piece of paper.

With the introduction of portable hand held computers and computing devices such as PDAs (Personal Digital Assistant), non-keyboards and non-keypads based methods for inputting data are receiving more interest in both academic and commercial research communities. The most promising options are pen based and voice based inputs. Pen based method in inputting can be either off-line or on-line.

THE STATEMENT OF NEED
One can imagine how easy life we have in this busy dynamic world in case the portable machine can understand what we write either in discrete or in natural handwriting mode. It would certainly be difficult that writing/typing addresses, memos, important information and communication as well for those who are non natives to English. In such a case, writing would be more cleared and easy to understand in their own local languages. In addition, it is helpful for those such as, computer novices, old people in using computer conveniently without the use of both keyboard and keypad. Therefore, the system having the intelligence in recognizing the natural handwriting for all possible scripts around the world is the global market demand.

Frequently, handwritten data is entered into IT solutions by human operators using keyboards. One can think of the processing of the forms, questionnaires, and notes etc. not only costlier but also time consuming. The easiest solution is the handwriting recognition system, which converts handwritten data into the format that can be used in further computing. Therefore, handwriting recognition is the connector between handwritten information and the IT world.

Despite many years of research in the field of handwriting recognition technology, IT has not reached the masses in local languages. Nepali is the one. According to the most recent official census, conducted by His Majesty's Government of Nepal (HMG) in 2001, Nepali is the mother tongue for 11 million people, which reflects the need of building a complete handwriting recognition system with maximum reliability.

A pencil and paper are often preferable for every one to use during the first draft preparation instead of using keyboard and other computer input interfaces like this. In such a case, handwriting recognition has gained the advantages. In addition, the languages having a large set of symbols, alphabets and numerals, designing a keyboard is bulky and cumbersome to use as well. For example, Nepali, Chinese and Japanese are under this category. Specifically, the basic set of symbols in Nepali consists of 31 pure consonants, 13 vowels, 16 modifiers and 10 numerals. In addition to these, consonants occur together in clusters, often called conjunct consonants. Modifiers appear on the top, at the bottom, on the left and to the right of the consonants and vowels, forming syllables. The script's rich of set of conjuncts make it complex to read and write. Altogether, more than 500 different symbols are used in Nepali. As, characters are derived from the mentioned basic sets, its input from the keyboard is cumbersome. However, new pen

tablets offer the possibility of on-line handwriting when combined with handwriting recognition technology. The best way to solve the need is not to design the keyboard but to design the complete writer independent natural handwriting recognition system, such that one can write in one's writing style. In case the keyboard is designed, one needs to practice to use but no extra task is necessary for one to write one's style. To the best of my knowledge, the proposed recognition system will be the first step to the contribution of Nepali handwritten character recognition.
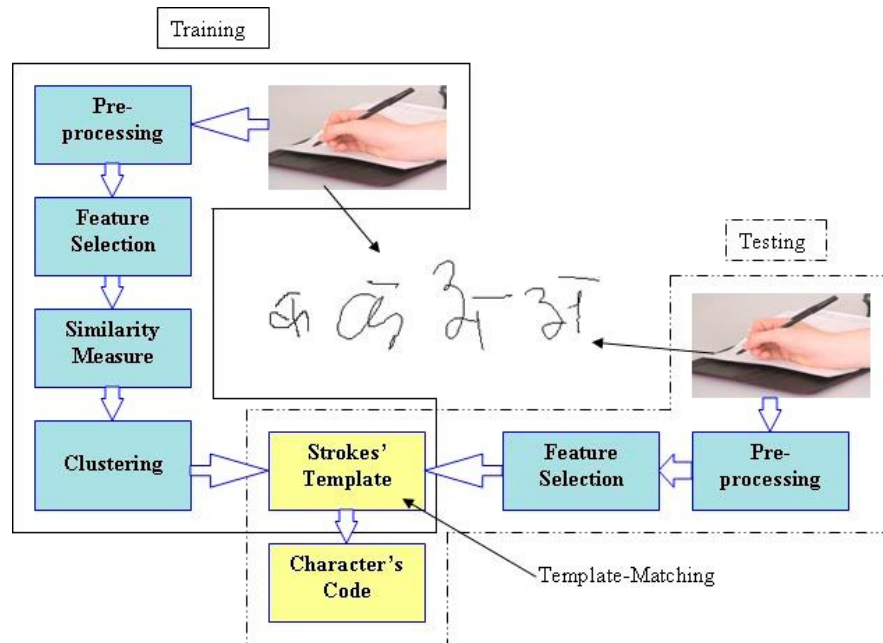

Figure 1: A basic block diagram of a template-based on-line character recognition system.

CONTEXT: ON-LINE CHARACTER RECOGNITION FRAMEWORK

On-line character recognition is the task of transforming a symbol represented in its spatial form of graphical marks into symbolic representation concurrently to the writing process (Bahlmann *et al.*, 2004). The recognition system typically comprises of two stages: a training stage and a test stage. In the training stage, data are refined (pre-processing), extracted their remarkable features (feature extraction), merged similar symbols (clustering) and stored their features' representatives as training samples (templates' management). Then in the test stage, matching takes place for identifying similar features with test features in classification. This is a refined example of a typical template-based on-line character recognition system. A basic block diagram of a template-based on-line character recognition system is shown in Fig. 1. A brief explanation of every block along with the existing methodologies and techniques are presented in the following sections.

BASIC TOOLS/TECHNIQUES

What kind of magic is required for a machine to become a competent handwriting classifier? All the components used in the classifier (varying case by case) are the requirements to fulfill the necessity of the competent classifier, which are explained below.

Digitizer Technology

People in various fields and generations come to handle the computers at their offices, school, and homes and on the street corners etc. That's why; an easier and flexible input device instead of keyboard is needed for faster processing. In such a case, computers with natural handwritten input is the possible solution, however, it is not a new in this era. A wide range of digitizer with different technologies is available in the market. In other words, use of digitizer tablets covers a wide range, based on their applications and reliability. Digital pen has been used as a human computer interface since few decades because of its flexibility in writing any kinds of texts, drawing graphics according to the users' desire. These are the cases, where the recognition block for handwritten graphics, texts and so on should be combined with the system. As there is large number of tablet digitizers, the main measuring precision is characterized by resolution, accuracy and sampling rate. A basic function of the digitizing tablet within a pen-computer is to detect the stylus on the writing surface and measure its position at its nominal sampling rate (Sandip, 2004). The sampling rate typically varies from 50-200 Hz depending on the application. Finer resolution is received with the higher sampling rate, which can accurately measure the fast strokes while reverse is the case for rough resolution. One of the interesting technologies is, the digitizing tablet is combined with the display screen, providing high level of interactivity. Users can perceive the drawing/writing on the same screen at the same time, which provides similar experience to that of drawing/writing using the conventional pen and paper. However, a special attention is needed to design a display screen along the digitizing surface.

Electromagnetic/electrostatic tablets have x and y grids of conductors, spaced from 0.1 to 0.5 and a loop of wire in the stylus tip (Tappert *et al.*, 1990). Either the loop or the grid is excited with an electromagnetic pulse and other detects an induced voltage or current in a   sinusoidal signal. In order to determine the precise location/position, the tablet conductors are scanned to locate pair closest to the loop and interpolation is performed.

Pressure sensitive tablets have layers of conductive and resistive materials with a mechanical spacing between the layers. An electrical potential is applied across one of the resistive layers in order to set up the voltage gradients that corresponds to the position. Pressure from the stylus tip at a point results in the conductive layer picking off the voltage from the resistive layer.

Resistive technology (Scholey, 2006) offers a fast, reasonably accurate, and affordable technology that recognizes the touch input from any stylus, finger, gloved hand, pen/tool. Further, due to its mainstream availability and low cost, resistive technology is the choice. It consists of a mechanical sensor mounted on the top of the display and an embedded controller. When the finger touches the surface, the surface touches the lower resistive layer and activates the signal. The control electronics alternates the voltage and hence (*x, y*) touch coordinates are recorded.

Another popular technology is the inductive technology (Scholey, 2006), used more than 20 years in graphics tablets and tablets PCs. A technology of single chip solutions made the tablet eight times smaller than it was before. It is comprised of printed circuit board sensor, a mixed IC controller, driver software, and stylus pen. The sensor emits an electromagnetic signal. The energy of the magnetic fields maintains the circuitry taking the energy from sensor to pen. An inductor/capacitor resonates to the frequency to determine its value. The energy is then reflected to the sensor, where it received the

analog signal. This analog signal is converted to digital signals, i.e. (*x, y*) coordinates.

Recently, electronic pen with the ultrasound technology is in use (Fujitsu, 2005). The ultrasound is generated from the special pen and the pen position is calculated when the ultrasound propagates to a receiver that is fixed to computer screen or paper. Accurate handwriting input, precise measurements with a compact device (0.1 mm resolution) and no blind spot in the writing areas in any corners are the main challenging properties.

WACOM, summarizes both the recent properties and possibilities in electronic pens (WACOM, 2006). As electronic pen does not need battery, a range of design options in terms of the weight and size (lightness, thickness etc.) are existing. The pressure can be detected and freely beset in between the standard purpose 256 steps and maximum level of 1024 steps. Utilizing the changes in capacitance and inductance are also proposed. Magnetic field noise is one of the difficulties, WACOM's pen-abled technology reduced by the use of combination of altering the arrangements of installed parts, adjusting the frequencies of parts, so that the sensor board is not affected.
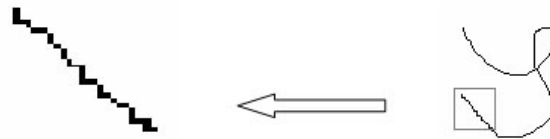

Figure 2: A sample of Jitter.

Pre-processing
Data, directly collected from users are often incomplete noisy and inconsistent, which are needed to be pre-processed before applying to the system in order to receive the correct classification. All the ways (techniques) to refine the data suitable for analyzing are included under the pre-processing technique. Sampling, noise elimination, discretization, integration, transformation are the basic techniques of pre-processing. Different systems use a variety of different techniques, which are varying from one script to another and depending on the goal as well. Some of the existing techniques are explained below.

Sampling
Pen-up and pen-down information is captured as an integral part of data acquisition. A string of coordinates as a function of time is recorded along the pen trajectory during the pen movement over the surface of the sensitive screen. This facilitates to track the number of strokes and their order within a character. The length of the strokes (number of coordinates in a string) varies even users write with the same speed. In order to achieve a constant number of coordinates in every string, re-sampling is necessary. One of the procedures to re-sample the sequence (Joshi *et al*., 2004) is explained below.

• **Single stroke character**
The total length of the trajectory is divided by the number of intervals required after sampling.

• **Multi stroke character**
The length of the stroke is divided by the total length of the character. In this case two steps are to be covered; one is to find the length of each stroke and next is to re-sample

each stroke according to the ratio of the stroke length to the total length of the character. The length of the complete character is needed to preserve the proportion of the strokes included in a character. Re-sampling also helps to reduce the anomalous cases such as having a large number of samples at the same position when the user holds down the pen at a point.

Noise Elimination
A stroke inevitably contains noises, typically come from many sources. Hand fluctuation during writing and digitizing error of the input devices are the major sources. The noisy sequence in addition to the information sequence may not rigorously harm the character in off-line graphical representation, while severely affect on on-line data sequence. Varieties of different filters are used based on the kinds of noises.

Figure 3: Words with different orientations and skews.

Joshi et al., 2004 reduced the effect of noise with the help of 5-tap low pass Gaussian filter, where each stroke is smoothed separately.

$$x_n^{filt} = w_n * x_n^{orig}$$

and

$$y_n^{filt} = w_n * y_n^{orig}$$

where,

$$w_n = \frac{e^{-\frac{n^2}{2\sigma^2}}}{\sum_{n=-(N-1)/2}^{(N-1)/2} e^{\frac{-n^2}{2\sigma^2}}}$$

is the filter's coefficient.

Filtering can be done by the use of convolution with one dimensional Gaussian kernels (Namboodiri *et al*., 2004), which reduces the noise due to pen vibrations and errors in the sensing mechanism. They proposed Gaussian low pass filter for smoothing the strokes. Malik *et al*., 2005 proposed time domain filter by using the convolution of input sequence with a finite impulse response for smoothing a jitter appeared in the sequence (Fig. 2). Although many techniques/filters are used in smoothing by suppressing the noisy coordinates, it is very difficult to select the filter such that it can work equally for all strokes appeared (Sezgin, 2001).

Normalization
Basically, recognition is better for only in case of nominal size of writing as well as standard orientation (normally horizontal) and a nominal slant (normally up-right) (Beigi *et al*., 1994). However, this is not happened from every writing. Therefore, the need of normalization is facing before feeding into feature extraction block. Normalization includes basic techniques like, scaling, translation, and rotation etc. Writing in any slope

can be changed into full upright position with the use of their base line, top line, and middle line. In contrast to a word processing, changing the orientation of the character into the full up-right position is more difficult. In case of a character, determining the orientation and skew angle is the main problem especially in Nepali. In contrast to Roman script, the diffculty in determining the orientation of the Nepali word is demonstrated in Fig. 3. In Nepali, an additional work has to be done because the orientation and the skew of both text and horizontal line (shirorekha) are not always the same. Further, size normalization is carried out to a standard size for all characters, which is an essential case in writer independent recognition but not so crucial in writer dependent (Nathan *et al*., 1995).

He *et al*., 2005 demonstrated the role of size normalization in recognition by taking static handwritten numerals. They normalized images into different sizes and applied to the same classifier and features to observe the relationship between normalized images and recognition rates.
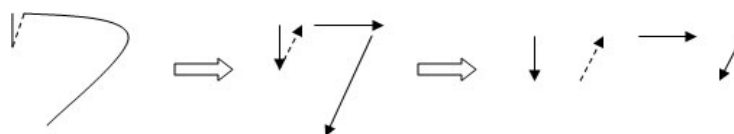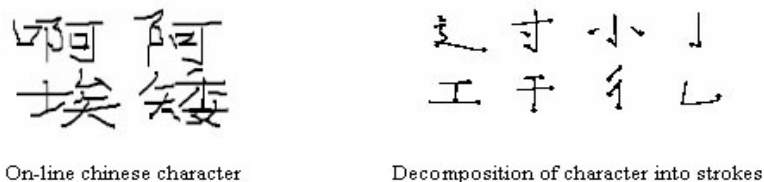

Figure 4: Feature design with the help of directional codes (Lee *et al*., 2000) (line segment approximation).

Figure 5: Feature design (Gao *et al*., 2000) with the help of standard strokes.

Repetition Removal
The digitizer is so sensitive because it detects every slight movement of the pen, even when the tip is not quite touching the digitizer but is over the plane. This will cause the co-occurrence of coordinates at a point. Further, very slow handwriting will generate repetition of coordinates at the same position, usually at the dominant points (for instance, corners). These unnecessary points are to be removed by any means for better performance of the system such as, retrieving quality image of the symbol and enhancing the speed by reducing the length of the symbol etc.

Use of excessive pre-processing is undesirable because it may result in premature, limiting distortions or loss of information. In a sense, a designed filter may not appropriate for all images. It means, the techniques used in pre-screening can work for

some of the characters but not for all. Sometimes, it may deteriorate the some classes of characters. Hence, the techniques, covering wide ranges are preferred. K. C. *et al*., 2006 attempted to prove an effective use of the preprocessing for Nepali (Best paper award).

Features

If you have complete address of your friend then you can easily find him/her without an additional help of other people on the way. The case is similar in character recognition to know the stroke with the help of the feature. Here, an address refers to a feature. Therefore, the complete/sufficient feature selection from the provided input is the necessary point. Elegant feature selection can greatly decrease the workload and simplify the subsequent design process of the classifier.

Features should contain information required to distinguish between the classes, be sensitive to irrelevant variability of the input, and also be limited to permit efficient computation of discriminant functions and to limit the amount of training data required (Lippmann, 1989).
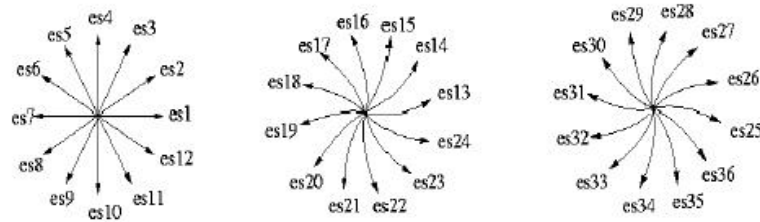

Figure 6: Sanparith's proposed elementary strokes for on-line handwriting. Elementary strokes are designed by 12 straight lines (es1 to es12), 12 convex strokes (es13 to es24), and 12 concave strokes (es25 to es36).

It is guaranteed that quality feature selection affects the classification rate. It is easy for classifier to recognize the symbol if it has feature with sufficient distinguishing characteristics. Different systems use different varieties of feature, having different accuracies and goals as well. The feature used in one system may not fit with the other systems. Features taken from the same input data can be variably used. It is noted that the feature used in on-line handwriting is different from off-line because of the different input techniques.

The root of the on-line handwriting recognition is the real time data collection by using the sampling phenomenon. Common devices are digitizing tablets and touch pads, where the written data is digitized. Digitized data points are in the form of either two dimensional (x,y) or three dimensional (x, y, z) coordinates (z is the pressure sensitivity). Broadly, many features such as geometric features, ink related features, directional features, and global features etc. are variably used from one task to another (Watt *et al*., 2005). On the other hand, number of strokes, length of a stroke, width-height ratio, number of intersection points, number of loops, number of hooks, point density, initial point position and direction, end point position and direction, and initial-end direction etc. are the basic features.

Toyozumi *et al*., 2004, proposed symbol segmentation method for handwriting mathematical formula by using positional stroke relation and geometrical features such as

shape, sizes and so forth. They clearly segmented the mathematical formula by using the knowledge of stroke neighbor relation.

Another primitive knowledge of the character is the standard strokes (structural representation). Mostly in Chinese, Japanese, and Korean characters, the case is used. Since large number of symbols is used to complete a character, many standard strokes/line segments are defined as the basic components of the character. Typically, strokes as the directional arrows are of eight types, coded from 0-7. This can be expressed as,

$$\begin{pmatrix} \nwarrow & \uparrow & \nearrow \\ \leftarrow & \cdots & \rightarrow \\ \swarrow & \downarrow & \searrow \end{pmatrix}$$
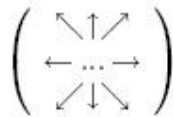
As directional arrows provide only the directional feature of the strokes/line segments, it would carry more information if the length of the standard strokes is taken into account (Cha et al., 1999). Not only in the mentioned scripts (Japanese, Chinese, and Korean) but also in roman, the directional codes are used as feature (Fig. 4: line segment approximation).

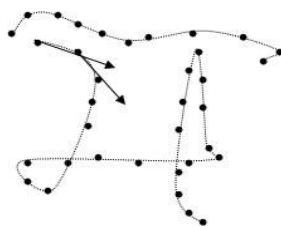Figure 7: Malik's proposed components of Urdu character.

Figure 8: A sample of slope measurement along the pen trajectory (i.e. feature: ($ft= (x,\ y,\ \theta t)$).

In some cases, extra standard strokes (properties of arcs) are also designed Gao *et al*., 2000 (Fig. 5); Rowely *et al*., 2002. Murakatat *et al*., 2003 and Murakatat *et al*., 2004 proposed both low-level and high level elementary strokes from on-line handwriting, shown in Fig. 6. Also, the slope angle obtained from two consecutive coordinates along the pen movement is quantized into eight levels as in the arrow matrix above, which makes the feature invariant to noise (Joshi *et al*., 2004). Dominant features are also used along the sequence. Dominant points are those points in the sequence where the slope changes noticeably/drastically. This is easier for finding the change of quantized slope.

Dominant points are extremely important in configuring the shape of the symbol.

Depending upon the applications, local features are rather used for higher confidence in classification. For instance, hat feature is an unique feature in Urdu, which cannot be ignored as in Roman. Along with this, characters are divided into horizontal, vertical, and slant line. How did they break a character into components are shown in Fig. 7.

Malik *et al*., 2005, used analytical segmentation on the input string of 2D coordinates in order to achieve the feature. The feature selected from the input string are starting and ending coordinates, slope, writing direction, size/length, and hat features. Both starting and ending coordinates are used to find the size, direction, and location (with respect to other) of the component. The slope is used to measure the slant line.

$$Slope = \frac{VerticalChange}{HorizontalChange}$$

A simple demonstration for the slope measurement along a pen trajectory path is shown in Fig. 8. Both the pen position and pen direction features (*x, y, θ*) are used (Okumura *et al*., 2005) instead of using only the string of coordinates. This is because that the pen coordinate feature is not stable. A combination of Δx, Δy, *cosθ* and *sinθ* (Nathan *et al*., 1995) comprised of a feature vector instead of taking only the pen-tip position, and the tangent angle. But in effect, it suffers from time complexity. On the other hand, reaching higher classification would be the basic goal.
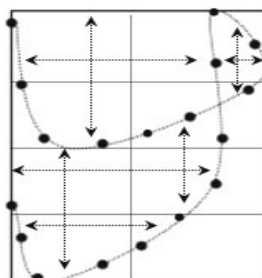

Figure 9: Zoning information from 8 different regions.

Verma *et al*., 2004, proposed a feature extraction technique for on-line handwriting, where they used directional and zoning information and combined them to make a single feature vector. Pen-down and pen-up calculation, direction of start and end point, change of writing direction, width height ratio, and zone information are the parameters used in feature.

• Pen-up
A stroke is a series of points from pen-down to pen-up. The number of strokes used in completing a character is equivalent to how many times the pen-down occurs. Pen-up calculation is needed as only the pen-down is not enough because the different characters may get the same number of strokes. It is used to check how well the recognized character matches to the standard one (the average in database). It is calculated as,

$$PEN - UP = e^{|average - x|}$$

where, x is the real strokes.

• Zone Information

It is carried out by using x_{max}, y_{max}, x_{min} and y_{min} from all numbers of strokes within a character, which are used to determine the boundary of the whole character and then separated into possible regions. Each region contains distinct information. A simple zoning information is demonstrating in Fig. 9. Interestingly, first and second derivatives of (x, y) coordinates, and relative change in pressure are also used as feature set (Vural *et al.*, 2005). They also added number of neighboring points and relatives height of each point with respect to the base line. As 2D coordinates (x, y) are not translation invariant, the first and second derivatives are taken, which are calculated as,

$$dx_t = \frac{\sum_{\theta=1}^{\Theta} \theta * (x_{t+\theta} - x_{t-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2}$$

where, x_t is the x coordinate at time t and Θ is the half window width. The percentage change in pressure can be determined as,

$$dp_t = \frac{p_{t+1} - p_{t-1}}{2 p_t}$$

where, p_t is the pressure at time t.

Alvertos *et al.*, 1989 used a large set of geometric information such as, existence of closed curves, number of intersections, number of free ends, horizontal and vertical symmetry, and existence of the diagonal neighbors to classify Greek character of any size. Even though a large set of geometric information about the character, this feature is reliable for only printed characters instead of natural on-line handwriting. More interestingly, on-line information is retrieved from off-line handwriting (Lallican *et al.*, 2000), experimentally proved that the recognition rate is close to those of an on-line recognition system.

Both the local and global features are variably used from one application to another. Besides, higher the dimensions used in a feature selection, greater the time complexity. A new application with the use of Nepali scripts is discovered where a very simple feature $(f_t = (x_t, y_t, \theta_t))$ for distinguishing symbols is used by K. C. *et al.*, 2006.

Distance Metrics

A quick overview of distance measurement techniques between the data points/sequences are carried out in the following subsections. It is often useful in image processing in order to calculate the distance between two pixels in an image, but this is not as easy as it seems. The presence of the pixel grid makes several so-called distance metrics possible, which often give different answers to each other for the distance between the same pair of points.

Euclidean

The most familiar distance metric, which is widely applicable from all ranges of data. The Euclidean distance function measures the 'as-the-crow-flies' distance. The distance between two sequences X and Y is,

$$d_{Euclidean} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

where,

$$X = [x_1, x_2, \ldots, x_i, \ldots, x_n], Y = [y_1, y_2, \ldots, y_i, \ldots, y_n]$$

Euclidean distance between two data points involves computing the square root of the sum of the squares of the differences between corresponding values.

Euclidean Squared

Mathematically speaking, it uses the same equation as the Euclidean distance metric, but does not take the square root. As a result, clustering with the Euclidean Squared distance metric is faster than clustering with the regular Euclidean distance. The output of Jarvis-Patrick and K-Means clustering is not affected if Euclidean distance is replaced with Euclidean squared. However, the output of hierarchical clustering is likely to change.

Manhattan

This is also known by the name City-Block distance metric. The distance is calculated as,

$$d_{Manhattan} = \sum_{i=1}^{n} |x_i - y_i|$$

where,

$$X = [x_1, x_2, \ldots, x_i, \ldots, x_n], Y = [y_1, y_2, \ldots, y_i, \ldots, y_n]$$

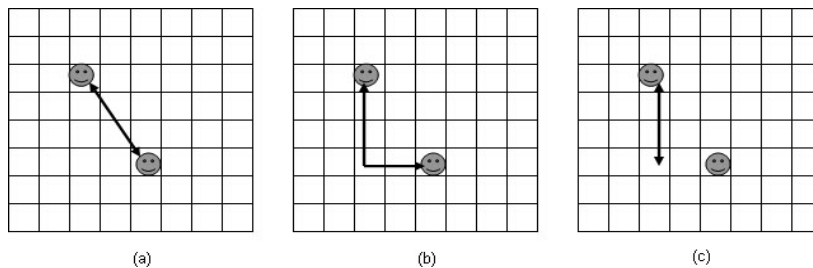A very simple graphical demonstration is shown in Fig. 10.

Figure 10: Examples: (a) Euclidean, (b) City-Block, and (c) Chess-Board distance measures.

Pearson Correlation

It is used to measure the similarity in shape between two profiles. Mathematically, Pearson correlation distance is,

$$d_{Pearson} = 1 - r$$

where, r is the dot product of the z-scores of the vector x and y. It is expressed as,

$$r = \frac{z(x).z(y)}{n}$$

and

$$z(x) = \frac{x - \bar{x}}{\sigma}$$

Pearson Squared

The Pearson Squared distance measures the similarity in shape between two profiles, but also captures inverse relationship. Mathematically,

$$d_{PearsonSquared} = 1 - 2r$$

where, r is the Pearson correlation defined above.

Chebychev

The Chebychev distance between two sequences is the maximum distance between the elements in any single dimension. The distance between the sequences $X = [x_1, x_2, ..., x_i, ..., x_n]$ and $Y = [y_1, y_2, ..., y_i, ..., y_n]$ is computed using the formula,

$$d_{Chebychev} = max_i \, |x_i - y_i|$$

where, x_i and y_i are the values of the i-th elements in both the sequences X and Y respectively. The Chebychev distance may be appropriate if the difference between the sequences is reflected more by differences in individual dimensions rather than all the dimensions considered together. It is noted that this distance measurement is very sensitive to outlying measurements. It is also called by chessboard distance metric.

Spearman Rank Correlation

It measures the correlation between two sequences. The two sequences are ranked separately and the differences in rank are calculated at each position, i. The distance between sequences is computed by,

$$d_{Sp\_Rank} = 1 - \frac{6\sum_{i=1}^{n}(rank(x_i) - rank(y_i))^2}{n(n^2 - 1)}$$

where, x_i and y_i are the values of the i-th elements in both the sequences X and Y respectively. The range of Spearman correlation is from -1 to 1. Spearman correlation can detect certain linear and non-linear correlations. However, Pearson Correlation may be more appropriate for finding linear correlations.

Dynamic Time Warping (DTW)

It is very easy to align two sequences having equal lengths but, how is it possible to align two non-linear sequences? The question is answered by the DTW itself. DTW overcomes the shortcoming of Euclidean distance measure and other common and simple techniques that are used for determining the distance between two sequences having equal length. In both speech and character recognition areas, such a case (sequences having different lengths) is occurred. Keogh *et al*., 1999 provided a concrete difference between the

simple Euclidean distance and DTW. It greatly focused on DTW in case of massive datasets.

A brief explanation of DTW with an example is described in the following. Consider two test T and reference R sequences, of length n and m respectively.

$$T =[t_1, t_2, \ldots, t_i, \ldots, t_n], R =[r_1, r_2, \ldots, r_i, \ldots, r_m]$$

In order to align two variable length sequences a matrix of size $n \times m$ is constructed. An element of that matrix contains the distance of two points t_i and r_j. Euclidean distance between two points can be expressed as,

$$d(i, j) = \sqrt{(t_i - r_j)^2}$$

Each matrix element (i, j) corresponds to the alignment between the points t_i and r_j. A warping path WP, now can be defined in the set of matrix elements that defines between T and R. The k-th element of W is, $w_k = (i, j)_k$. Now, we have,

$$W = w_1, w_2, \ldots, w_k, \ldots, w_K$$

with the condition that $max(m, n) \leq K < m + n-1$.

Following constraints are considered in warping path:

• **Boundary Condition**
$w_1 = (1, 1)$ and $w_k = (n, m)$, is simply stated. This is necessary the warping path to start and finish diagonally in opposite corner cells in the matrix.
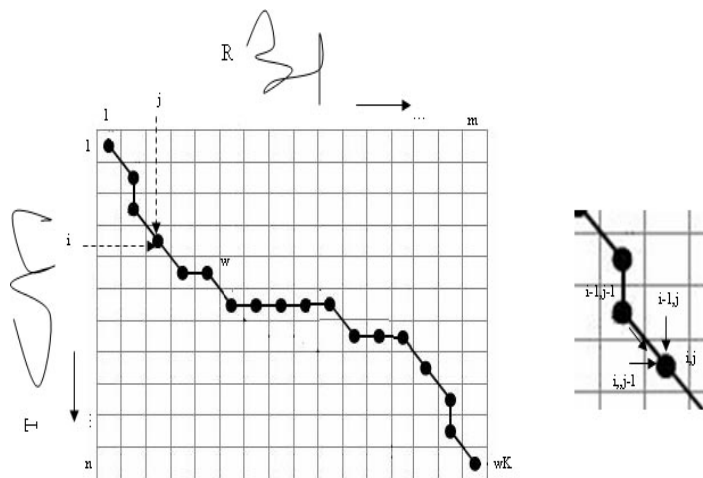

Figure 11: An example of warping path during the alignment of two non- linear sequences by using DTW.

• **Continuity**
Given $w_k = (a, b)$ then $w_{k-1} = (a', b')$, where $a-a' \leq 1$ and $b-b' \leq 1$. This makes the path runs diagonally not to other adjacent cells.

• **Monotonicity**
Given $w_k = (a, b)$ then $w_{k-1} = (a', b')$, where $a-a' \geq 0$ and $b-b' \geq 0$. This forces the points in W to be monotonicity space time.

On the other hand, the warping path can be efficiently determined by using Dynamic Programming. It can be expressed as,

$$D(i, j) = \min[D(i-1, j-1), D(i-1, j), D(i, j-1)] + d(t, r)$$

$D(i, j)$ is the cumulative distance between two sequence T and R from three adjacent cells and Euclidean distance in the (i, j), which is shown in Fig. 11. The element at the end of the matrix gives the distance between the sequences. Euclidean distance is the special case of DTW where the k-th element of W is constrained such that $w_k = (i, j)_k$, $i = j = k$. However, it is defined only in case the sequences are of same length. One of the main properties of DTW is time complexity. The time complexity is defined as $O(nm)$.

Typically, two situations are faced in matching the sequences.

• **Whole Matching**
Matching takes place between two sequences having identical lengths. In such a case, Euclidean distance can be possible.

• **Subsequence Matching**
It is time consuming to match two sequences having different lengths. Consider a test sequence T is smaller in length in comparison to reference sequence R. The test sequence slides along the subsections of every possible subsection of R to find the best match.

EXPERIMENT ON CURSIVE HANDWRITING – NEPALI

Handwriting Properties and Major Problems in Recognition of On-line Handwritten Nepali Characters.
Handwriting recognition is notoriously a difficult task. However, it has been investigated since a long time and can be put into a general framework, under pattern recognition and image processing. One of the main difficulties in on-line character recognition is, writing that looks identical with the printed characters (off-line graphical representation) may have a chance of mis-classification because of different sequential representation (on-line). This is not only the problem of Nepali but also belongs to other scripts as well. Main problems associated with on-line handwritten character recognitions are classified into following categories:

• **Similarity in drawings among many classes of characters**
Many of the characters are similar one another in structure. In addition, users writes two or more classes of character in a similar way with the use of same number of strokes in a same direction and their order, which can be the most difficult part to classify correctly. In Nepali, (क, फ), (छ, ध), (य, प), (ढ, द), (इ, ड), (स, झ), (भ, म), (च, य), (ध, घ), (थ, य), (न, त), (ञ, ज), (व, त), and (ढ, ट) etc. are the some similar characters' pairs, which are hard to analyze their dissimilarity from one another in case of natural handwriting.

46

• Writing Units

- Numbers of strokes, their order, shapes and sizes, directions, skew angle etc. are the considerable writing units. Writing units are always vary from one user to another. It is no guarantee that same user writes in a same way from time to time writing.
- Strokes may vary in direction in different peoples' styles.
- The length of a string of 2D coordinates of a specific stroke is variable.
- The number and order of strokes are variable within a specific letter. The number of strokes to complete a discrete letter varies from one to four (without modifiers) in case of Nepali.
- The size (big or small) of a specific symbol is variable.
- The speed in writing is variable, which determines how densely the coordinates collected and what shapes (either smooth or rough) of the strokes are.
- Writings can be tilted by some angle with respect to the horizontal line (either on the left or to the right).

Even when, numbers, order, shape and size, direction of strokes and speed varied from time to time writing even within the same class of character and from the same user, the writing samples are readable. The degree of variation depends on the style and speed of writing. In Nepali, variable number of strokes is used for the same symbol to complete. Fig. 12 demonstrates the variable writing styles for the first consonant क. In purely cursive mode of writing, user uses only one stroke to complete the text in a character, while two or more than two strokes are used in purely discrete mode. In addition, it is not sure that the direction(s) of stroke(s) while drawing the same character in different time by using the same number of stroke(s) is/are same from the same user. Nepali characters are mostly influenced with the variability in writing units. This is how it is regarded as one of the cursive scripts. It is therefore difficult to classify correctly not only in cursive handwriting but also in purely discrete mode. In case of purely discrete mode, lots of strokes can be there, in which each stroke independently may not have any information about the symbol.
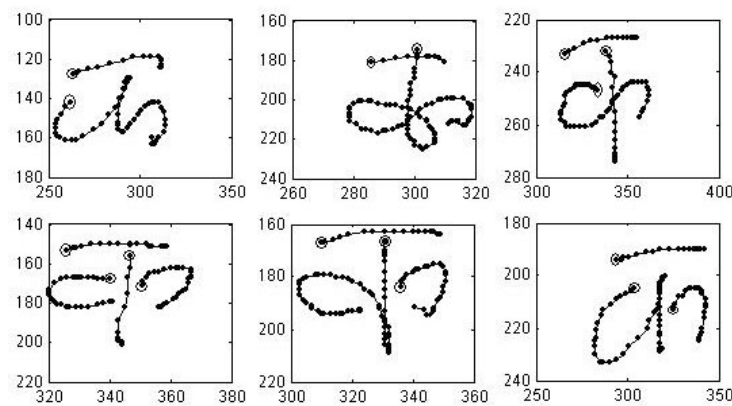

Figure 12: A demonstration of variable natural handwriting for the first consonant क.

• Digitizer: Problem emerges into two major forms.
Pens' or mouse's physical position on tablet (or mouse pad, in the case of using mouse), a continuous movement as a function of time, is mapped into discrete (*x, y*) pixels in screen coordinates, thus there is an information loss due to digitization. Noise due to physical condition of devices, for example, dirt on the tablet surfaces and dirt on the mouse's

wheel.

• Traditional Technologies

Traditional technologies can not reach the line of the global market demand in the field of handwriting recognition technology. Most of the commercial machines are reliable only on the box discrete characters and character with purely discrete strokes. They are limited to natural (cursive handwriting). To the best of my knowledge, many of the existing works were concentrated on printed and off-line writings in Devanagari. From the practical experience, Optical Character Recognition (OCR) has been used in Devanagari. However, new demands are still there to be done in this script. The necessity of new technology along with the novel idea for efficient and faster machine to classify the characters and numerals is to be taken.

As a character can be written in different varieties of drawing, the fundamental properties of writing which makes communication/recognition possible is that the differences between the different characters are more significant than differences between different drawings of the same character.
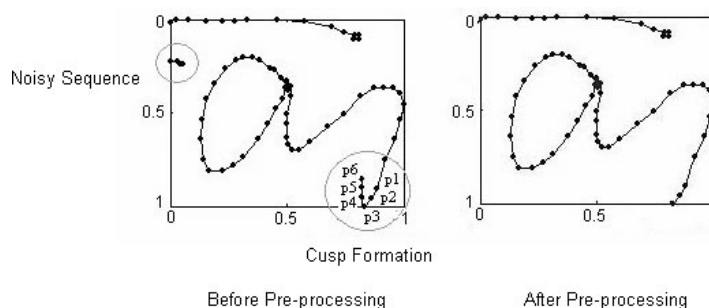

Figure 13: A demonstration of the pre-processing: size normalization, repeated coordinate deletion and cusp elimination (cusp - noisy sequence) for the first consonant क.

Experimental Abstract

This following experiment on Cursive Nepali handwriting recognition utilizes structural properties of those alphanumeric characters, which have variable writing units. Pre-processing techniques are varying with the script used from one system to another. Generally, real time data are noisy and incomplete. In Nepali, it is very difficult to identify and eliminate the noisy sequences. However, this paper eliminates cusps or undesirable hooks at ascenders and descenders, based on the nature of slopes. This is chopped by the use of angles among 8-10 two-dimensional coordinates if the angle changes drastically (approx. 90 degrees is considered). In addition, size normalization is needed because of the variable size of writing samples (sometimes very big and sometimes very small). Size normalization is done to change the variable sized alphanumeric character to the standard size. The new window size is designated as,

$$p_{new} = (\frac{p - p_{min}}{p_{max} - p_{min}})(p_{max'} - p_{min'}) + p_{min'}$$

where, p_{max} and p_{min} are the maximum and minimum coordinate points of all strokes within a letter. $P_{max} = (1,1)$ and $p_{min} = (0,0)$ give the size of new window where all writing

samples are restored. The way how we pre-processed the strokes is demonstrated in Fig. 13.

It uses a string of pen tip's positions and tangent angles of every consecutive point as a feature vector sequence of a stroke for stroke classification. Mathematically, a complete feature of a *j*-th stroke is,

$$F_{i,j} = [(p_{i,j,1}, f(p_{i,j,1}, p_{i,j,2})), (p_{i,j,2}, f(p_{i,j,2}, p_{i,j,3})), \ldots, (p_{i,j,l-1}, f(p_{i,j,l-1}, p_{i,j,l}))]$$

where,

$$f(p,q) = \arctan(\frac{y_q - y_p}{x_q - x_p})$$

Fig. 14 tells the simplest way to extract feature from the strokes. Two steps are covered: firstly, similarity measure and secondly, single-linkage agglomerative hierarchical clustering, based on similarity in order to complete the clustering. Distance finding technique between the features of the strokes is a common approach to estimate the similarity.

Consider two feature vector sequences: *A* and *B* of length *N* and *M* respectively. $D(n, m)$ is the global distance up to (n, m),

$$D(n,m) = min[D(n-1,m-1), D(n-1,m), D(n,m-1)] + d(n,m)$$

with an initial condition $D(1, 1) = d(1, 1)$, where $d(n, m)$ is an element of the local distance between the events i.e.,

$$d(n,m) = |e_n^A - e_m^B|$$

The global distance between *A* and *B* is, *Dist.* (*A*, *B*) = $D(N, M)$. This is often called DTW-matching score in recognition process. Fig. 15 shows two pairs of numerals ((*1,1*) and (*1,2*)) associated in matching process for similarity determination. Two feature vector sequences of strokes are said to be similar if the matching-score in between them is smaller than other pairs.
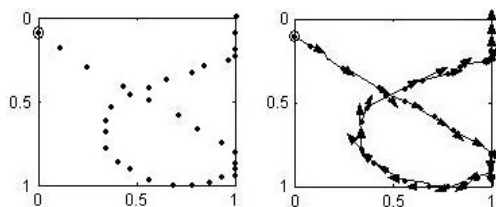

Figure 14: Feature Selection: both the pen-tip and the tangent at every position are extracted as a feature along trajectory path for a numeral 4.

Experimental Results

No directions, constraints and limitations were given to the users but they were encouraged to write alphanumeric characters with variable stroke number and order along the horizontal line. 25 users were employed to write two times for each letter, (i.e. 25 users $\times$ 2 times$\times$46 classes). Training data was composed of 60% of data from every class

and remaining 40% were for testing. The dataset was comprised of natural handwritings. We trained separate system using original and pre-processed writing samples. Table 1 reveals the experimental results. We constructed a prototype recognizer that uses the DTW algorithm to align handwritten strokes with stored stroke templates and determine their similarity. Separate system is trained for original and preprocessed writing samples and achieved recognition rates of 85.87% and 88.59% respectively. Pre-processing includes both size normalization of the multi-strokes characters, noise removal in every stroke's sequence and deletion of repeated pen-tips position.
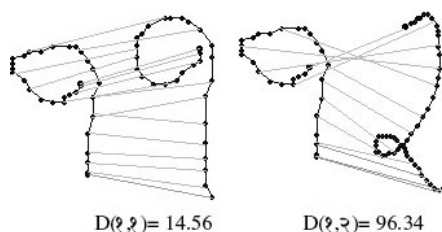

D(१,१)= 14.56   D(१,२)= 96.34

Figure 15: Two pairs of numerals (१, १) and (१, २) are taken for finding distance between them. With the help of DTW, the pair: (१, १) are found to be similar as distance between them is smaller in comparison to (१, २).

All simulations were done using MATLAB 7.0.4 on a 1.81 GHz, 1.00 GB RAM, PC running Microsoft Windows XP professional.

TABLE 1. Error rates for both original and preprocessed strokes

Dataset	Char.'s Type (Class)	Test Char.s	Original Strokes		Pre-processed Strokes	
			Char.s (Err)	Avg. Err	Char.s (Err)	Avg. Err
Training	Consonant (31)	930	26 (02.79%)	02.75%	22 (02.36%)	02.31%
	Vowel (5)	150	08 (05.34%)		07 (04.67%)	
	Numeral (10)	300	04 (01.34%)		03 (01.00%)	
Test	Consonant (31)	620	95 (15.32%)	14.13%	81 (12.58%)	11.41%
	Vowel(5)	100	17 (17.00%)		13 (13.00%)	
	Numeral (10)	200	18 (03.00%)		11 (02.00%)	

TABLE 2. Error Type (Test Data)

Error Type	Original Strokes (Err)	Pre-processed Strokes (Err)
Similar Features	77 (08.15%)	56 (07.28%)
Diminished or very long Ascenders and Descenders	16 (01.73%)	13 (01.41%)
Miswriting	18 (01.95%)	18 (01.95%)
Rewriting Strokes	13 (01.41%)	13 (01.41%)
Miscellaneous	06 (00.65%)	05 (00.64%)

DISCUSSIONS

This section investigates the overall performance of the classifier based on the approach/ideas used along with analysis of errors observed. At first, how the difficulties

mentioned above are recovered, are analyzed.

• **Normalization**: No matter how big/small the size of the strokes. Use of size normalization for variable sized characters provides an ease to the users in writing naturally.

• **Cusp Elimination**: Fig. 16 references an effective example of how cusp elimination works for higher classification. Many of the similar characters are classified correctly through the use of cusp elimination.

• **Tangent as a Feature**: Because of the use of tangent angle at every pen tip's position, it is flexible for any tilted writings. Tangent at every pen-tip along the trajectory gives the true structural skeleton of the symbol/stroke.

• **DTW**: Non-linear strokes alignment is only possible with the use of DTW algorithm and hence its advantage has been taken here.

• **Template Based Approach**: This approach is completely based on what users write and hence it provides a confidence in classifying natural handwriting rather than printed and box characters.

Furthermore, investigation is carried out in mis-recognized characters and their reasons, based on the experimental results. Table 2 analyzes the overall error occurred in the experiment. During the experiment, every character was checked in case of mis-recognition to know how and why it was mis-recognized?, which is the important part to be dealt with those problems. Based on this, error types are clearly mentioned below for all mis-classified characters.

• **Similarity features**: Out of many confusion pairs as discussed in section **1**, Fig. 16 demonstrates three confusion similar characters. Most of the similar characters' pairs like, (भ, म), (थ, य), and (ध, घ).

• **Diminished and/or very long ascender and/or descender**: Due to lack of skill in writing by using digitizer, some of the strokes have very long/small (diminish) ascender/descender. Having very long/small strokes' sequence causes problems in classifying the character. The demonstration of this problem is shown in Fig. 17. However, most of these cases are classified correctly.

• **Re-writing strokes**: Writing a stroke at the end to complete the previous strokes is often called re-writing stroke. These strokes are helpful in giving perfect/complete graphical (off line) representation but the information carried by only a re-writing stroke is nothing. These are often mis-classified (Fig. 18). Complete information of the specific stroke will exist if we combine the previous strokes with re-writing stroke. However, it is difficult for the characters, which have large number of strokes i.e., how to know which one is the re-writing stroke and which stroke is to be connected to the re-writing stroke. One of the ideas used in pre-processing: noisy sequence elimination, sometimes deletes the re-writing strokes, from which it has a chance of recognizing.

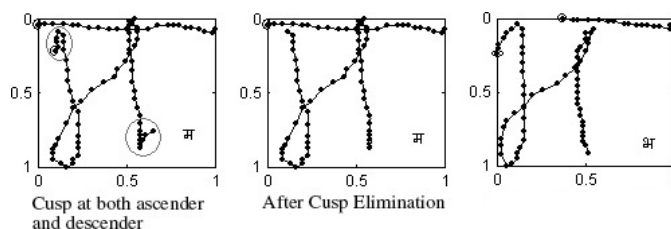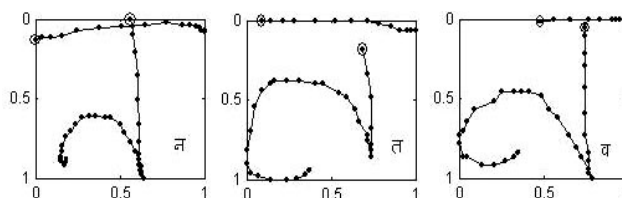Cusp at both ascender and descender   After Cusp Elimination

Figure16: It shows a true effect in correct classification with the use of effective stroke

pre-processing technique i.e. cusp elimination. Due to this, 'म' is very rarely confused with 'भ'.

Figutr 17: It illustrates three drawings of confusion characters due to diminishing descenders. The classifier classified both leftmost and rightmost as the middle one. Human is also confused with these pairs.

• **Mis-writing and Miscellaneous**: A collection of a set of strokes having no information about the characters is grouped under mis-writing. These characters are often rejected. These are sometimes happened in cursive handwritings. The worst drawing is found in miscellaneous category.

From the mentioned error analysis, the system is limited to tremor handwriting (specially children handwriting), characters with re-writing strokes and mis-writing characters. Tremor handwritings are often producing a collection of strokes but no information.
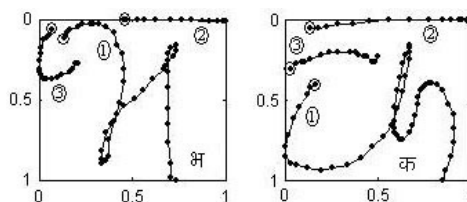

Figure 18: It shows two examples of re-writing strokes. Every third stroke is used to complete the first stroke. Even though it is readable for human, the last stroke makes system in confusion. User writes third stroke at the end of writing in order to isolate the drawings from their similar characters, i.e. to isolate भ from म and क from फ.

CONCLUSIONS AND FUTURE DIRECTION

A comprehensive study of pre-processing, feature selection, and distance metric techniques for determining the similar sequences is explored. In addition a novel idea and its efficacy for Nepali natural writer independent handwritten character recognition is explored. The highest performance of the proposed recognition system is 89% for natural cursive Nepali handwriting. This uses only the structural properties of the symbols for classification. We proposed that we will use both the structural properties of the multi-strokes character and their spatial information within the strokes of every character for achieving higher classification confidence as well as efficacy. Further, the unique feature of Nepali character: the 'shirorekha' is used as the reference feature sequence (specifically location of the shirorekha with reference to other symbols), which helps to achieve the accuracy more than 95% and a perfect comparison is made with the classifier, where spatial information is not in use. This later research paper is going to be published

soon.

A syllable level recognition in Nepali is the direction to move in the days to come. We plan to extend the same spatial relation among the strokes of the complete syllable by extending the number of regions. We will carry extra stroke-pre-processings/post-processing for only those characters, which have low level of recognition confidence.

REFERENCES

1   Anoop M. Namboodiri, Student Member, IEEE, Anil K. Jain, 2004. Fellow, IEEE, "On-line Handwritten Script Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 26, No. 1 pp. 124–130.

2   B. B. Chaudhuri and U. Pal, 1997. "Skew Angle Detection of Digitized Indian Script Documents", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No 2 pp.182–186.

3   Brijesh Verma, Jenny Lu, Moumita Ghosh and Ranadhir Ghosh, 2004 "A Feature Extraction Technique for On-line Handwriting Recognition", IEEE International Joint Conference on Neural Networks,pp.1337– 1341.

4   C. C. Tappert, C. Y. Suen and T. Wakahara, 1990 "The State of Art in On-line Handwriting Recognition", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.12, No.8, pp. 787–80.

5   Chun Lei He, Ping Zhang, Jian Xiong Dong, Ching Y. Suen and Tien D. Bui, 2005. "The Role of Size Normalization on the Recognition Rate of Hanwritten Numerals", The 1st IAPR TC3 NNLPAR workshop, pp.8–12.

6   Eammon J. Keogh and M. J. Pazzani, 1999. "Scaling Up Dynamic Time Warping to Massive Datasets", 3rd European Conference on Principles and Practice of Knowledge Discovery in Databases, Vol.1704, pp.1–11.

7   Lippmann R., 1989. "Pattern Classication using Neural Networks" IEEE Communications Magazine, pp. 48.

8   Homayoon S. M. Beigi, Krishna Nathan, Gregory J. Clary and Jayashree Subrahamonia, 1994. "Size Normalization in On-line Unconstrained Handwriting Recognition", IEEE, ICIP, pp.163-173.

9   Fujitsu laboratories, 2005. "Fujitsu - The Possibilities are Infinite, Ultrasound Electronic   Pen".

10   Francois Bouteruche, Eric Anquetil and Nocolas Ragot, 2005. "Handwritten Gesture Recognition Driven by Spatial Context of Strokes", 8th International Conference on Document Analysis and Recognition (ICDAR), pp. 1221–1225.

11   Ian Scholey, 2006. "Digitizer technology Comparison: Inductive and Resisitive Technology" Pen Computing Magazine: http://pencomputing.com/features/wacom digitizer comparison.html.

12   M. Blumenstein, B. Verma and H. Basli, 2003. "A Novel Feature Extraction Technique for the Recognition of Segmented Handwritten Characters", 7th IEEE International Conference on Document Analysis and Recognition, pp.137.

13   Panu Somervuo and Teuvo Kohonen, 1999. "Self Organizing Maps and Learning Vector Quantization for Feature Sequences", Neural Network Processing Letters", Vol.10, No.2, pp.151–159.

14   R. M. K. Sinha and H. N. Mahabala, 1979. "Machine Recognition of Devanagari Script, IEEE Transactions System", Man and Cybernetics, Vol.9, No.8, pp.435– 441.

15   Scott D. Connell and Anil K. Jain, 2001. "Template-based online character recognition. Pattern Recognition", Pattern Recognition, Vol.34, No.1, pp.1–14.

16   Niranjan Joshi, G. Sita, A. G. Ramakrishnan, Deepu V., Sriganesh Madhavanath, 2005. "Machine Recognition of On-line Handwriteen Devangari Characters", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 12, No. 8, pp. 787–807.

17   Sanparith Marukatat, Rudy Sicard, Thierry Artieres, Patrick Gallinari, 2003. "A Flexible Recognition Engine for Complex On-line Handwritten Recognition", 7th International Conference on Document Analysis and Recognition (ICDAR), pp. 1048–1052.

18   Sanparith Marukatat, Thierry Artieres, Patrick Gallinari, 2004. "A Generic Approach for On-line Handwriting Recognition" 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR), pp. 401–406.

19   Sanparith Marukatat, Thierry Artieres, 2004. "Handling Spatial Information in On-line Handwriting Recognition", 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR), pp. 14–19.

20   Santosh K. C. and Cholwich Nattee, 2006. "Structural Approach on Nepalese Natural Handwriting Recognition", 2nd IEEE International Conference on Cybernetics Intelligent Systems, pp.711–716.

21   Santosh K. C. and Cholwich Nattee, 2006. "Stroke Number and Order Free Hanwriting Recogntion for Nepali", Lecture Notes in Computer Science (LNCS), Subseries: Lecture Notes in Artificial Intelligence (LNAI), Vol. 4099, pp. 990-994.

22   Santosh K.C. and Cholwich Nattee, 2007. "Template-based Nepali Handwritten Alphanumeric Character Recognition", Thammasat International Journal of Science and Technology (TIJSAT), Thammasat University, Thailand, Vol. 1.

23   Santosh K. C., Cholwich Nattee, 2006. "Effect of Pre-processing and feature Selection   in Recognition for Nepali" The first International Conference on Knowledge, Information and Creativity Support Systems (KICSS), Ayuthya, Thailand, pp. 139-146.

24   Sandip S. 2004. "The Digital Pen as a Human Computer Interface" IEEE International Symposium on Consumer Electrnoics, pp. 629–634.

25   Scott D. Connell, R. M. K. Sinha and Anil K. Jain, 2000. "Recognition of Unconstrained On-Line Devanagari Characters", 15th IEEE International Conference on Pattern Recognition, pp.368–371.

26   Veena Bansal and R. M. K. Sinha, 2002. "Segmentation of Touching and Fused Devanagari Characters", Pattern Recognition, Vol.35, No.4 pp. 875–893.

27   W. Guerfali and R. Plamondon, 1993 "Normalizing and Restoring On-line Handwriting", Pattern Recognition, Vol.2, No.3, pp.419–431.

28   C. Bahlmann and H. Burkhardt, 2004. "The Writer Inependent On-line Handwriting Recognition System Frog on Hand and Cluster Generative Statistical Dynamic Time Warping", IEEE transactions and Pattern Analysis and Machine Intelligence, vol.26, no.3.

29   Wacom Technology, 2006. "Wacom Components: Technology: EMRrTechnology" http://www. wacom-components.com/english/technology/emr.html.

30   Sumaila Malik, Shoab A. Khan, 2005. "Urdu Online Handwriting Recognition", IEEE International Conference on Emerging Technolgies, pp. 27–31.

31   Krishna S. Nathan, Homayoon S. M. Beigi, Jayashree Subrahmonia, Gregory J. Clary, Hiroshi Maruyama, 1995. "Real-time On-line Unconstrained Handwriting Recognition Using Statistical Methods", International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Vol. 4, pp. 2619– 622.

32   Stephen M. Watt, Xiaofang Xie, 2005. "Recognition of Large Sets of Handwritten Mathematical Symbols", IEEE International Conference on Document Analysis

and Recognition (ICDAR), pp. 39–43.

33   Kenichi Toyozumi, Naoya Yamada, Takayuki Kitasaka, Kensaku Mori, Yasuhito Suenaga, Kenji MaseTomoichi Takahashi, 2004. "A Study of Symbol Segementation Method for handwritten Mathematical Formula Recongition using Mathematical Structure Information", IEEE 17th International Conference on pattern Recogntion (ICPR), pp. 630–633.

34   Daiki Okumura, Seiichi Uchida, Hioraki Sakoe, 2005. "An HMM Implementation for On-line Handwriting Recognition based on Pen-Coordinate Feature and Pen-Direction Feature", Bioinformatics, Vol. 18 No. 5 pp. 735–46.

35   Esra Vural, Hakan Erdogan, Kemal Oflazer, Berin Yanikoglu, 2005. "An Online Handwriting Recognition System for Turkish", Document Recognition and Retrieval XII: In proceedings of SPIE Electronic Imaging Symposium, Vol. 5676 pp. 56–65.

36   Sung-Hyuk Cha, Yong-Chul Shin, Sargur N. Srihari, 1999 "Approximate Sequence String Matching Algorithm for Character Recognition and Analysis" Fifth International Conference on Document Analysis and Recognition (ICDAR), pp. 53–56.

37   Jiang Gao, Xiaoqing Ding, Jing Zheng, 2000. "Image Pattern Recognition Based on Examples– A combined Statistical and Structural-Syntactic Approach", In Proceedings of Joint IAPR International Workshops, SSPR 2000 and SPR 2000., Vol. 1876, pp. 57–66.

38   Henry A. Rowley, Manish Goyal, John Bennett, 2002. "The Effect of Large Training Set Sizes on On-line Japanese Kanji and English Cursive Recognizers", 8th IEEE International Workshop on Frontiers in Handwriting Recognition (IWFHR), pp. 36–40.

39   Nicolas Alvertos, Ivan D'cunha, 1989. "Optical Machine Recognition of Greek Characters of Any Size", IEEE Proceedings: Energy and Information Technologies, Vol. 2, pp. 623 627.

40   P. M. Lallican, C. Viard-gaudin, S.Knerr, 2000. "From Off-line to On-line Handwriting Recognition", 7th International Workshop on Frontiers in Handwriting recognition, International Unipen Foundation, pp. 303–312.