Paper ID: NITETE&TC05

THE HANDWRITTEN DEVNAGARI NUMERALS RECOGNITION USING SUPPORT VECTOR MACHINE

Rupali Vitthalrao Suryawanshi
Department of Electronics Engineering, Bharatratna Indira Gandhi College,
Solapur, India, 413006 suryawanshi.rupali15@gmail.com

ABSTRACT

Handwriting has continued to persist as a mean of communication and recording information in day-to-day life even with the invention of new technologies. Natural handwriting is one of the easiest ways of information exchange between a human and a computer. Handwriting recognition has attracted many researchers across the world since many years. Recognition of online handwritten Devanagari numerals is a goal of many research efforts in the pattern recognition field The main goal of the work is the recognition of online hand written Devanagari numerals using support vector machine. In the data collection phase, co-ordinate points of the input handwritten numeral are collected as the numeral written; various algorithms for pre-processing are applied for normalizing, resampling and interpolating missing points, smoothing and slant correction. Two low-level features i.e. direction angle and curvature are extracted from the pre processed data. These features along with the x and y coordinates of the input handwritten character are stored in a .csv file and fed directly to the recognition phase. Recognition is done using four kernel functions of SVM by partitioning the data into different schemes. The recognition accuracies are obtained on different schemes of data using the four kernel functions of SVM for each scheme.

KEYWORDS— Data Collection, Pre-processing, Feature Extraction, SVM, Handwriting Recognition.

INTRODUCTION

The primary modes of data input between a user and a computer are still the conventional input devices such as keyboards and mouse. These devices have some limitations when compared to the input through natural handwriting. For scripts such as Chinese and Japanese which have a very large alphabet set and due to complex typing nature of script such as Devanagari and Gurumukhi, it becomes difficult to input data to the computer through the conventional input devices. Natural handwriting is one of the easiest ways to exchange information between a human and a computer. Thus, the Devnagari numerical recognition field has better communication between the user and the computer by using SVM

SVM stands for Support Vector Machine and this method of recognition is gaining immense popularity now-a-days. SVM classifier gives better results as compared to the other classifiers for the handwritten numeral recognition of Kannada script in [3]. In [6], SVM approach is used to recognize strokes in Telugu script. The set of strokes are segmented into subsets based on the relative position of the stroke in a character. An SVM based classifier is built for recognition of strokes in each subset. A rule based approach is used to recognize a character from the sequence of stroke classes given by the stroke classifier. SVM is a new classifier that is used in many pattern recognition applications with good generalization performance. SVM has been used in recent years as an alternative to popular methods such as neural network.

PHASES OF ONLINE HANDWRITING RECOGNITION SYSTEM

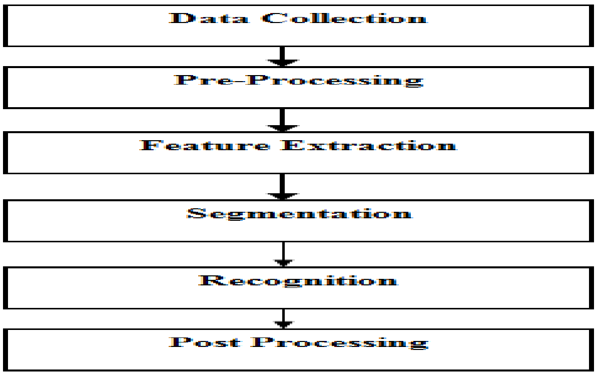

Fig.1: Flow of Numerical Recognition System

These techniques were based on electromagnetic/electrostatic and pressure sensitive techniques the combination of digitizer and display in the same surface has become very common since many years. There is an established procedure to recognize online handwritten data which includes the following phases or components: Data collection, pre-processing, feature extraction, segmentation, recognition and post processing.

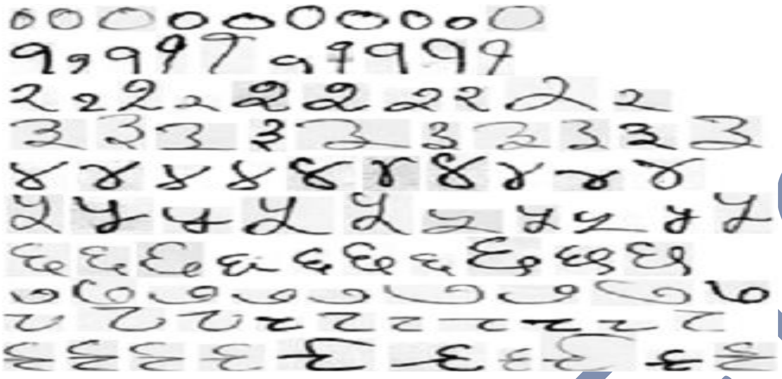

Figure 2. Sample Devnagari Numerals from ISI database.

METHODOLOGY

A. DATA COLLECTION: -

Data collection is the first phase of online handwriting recognition that collects the sequence of co-ordinate points of the moving pen. A transducer is required to capture the handwriting as it is written. The most common devices used for this purpose are electronic tablets or digitizers. A digital pen is used for writing on these devices. Digital pen is also sometimes called as "stylus". A typical digital pen includes two actions namely Pen Down and Pen Up. The connected parts of the pen trace between Pen Down and Pen Up is referred to as a stroke. It is considered as a smallest unit in handwriting recognition. All the unique strokes of a script are manually identified and given unique labels. Stroke is basically a smallest physically identifiable unit in online handwriting. The pen traces are sampled at a constant rate and thus, these pen traces are evenly distributed in time but not in space. The most common examples of electronic tablet or digitizers include Personal Digital assistant (PDA), tablet PC, cross pad (or pen tablet) and a digi memo. A digimemois a portable device which digitally captures and stores the ink written on ordinary paper using a digital pen and pad, thus providing a natural interface for collecting handwritten data samples.

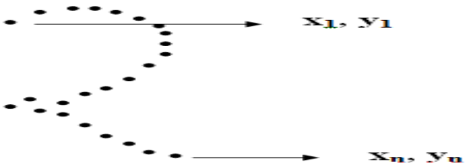

Figure 3. Points of pen movement collected while writing

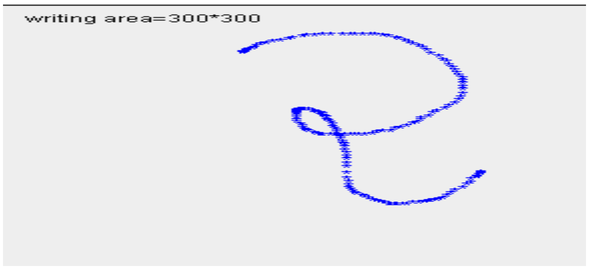

Fig 4.Writing area for writing the numerals

B. PRE-PROCESSING:-

While inputting data through a pen on the digitizer tablets, there may be certain noise distortions present in the input text due to some limitations, which may make the recognition of input difficult. Irregular size, missing points due to fast movement of the pen, uneven distances of points from neighboring positions are various forms of noise and distortions. These noise and distortions present in the input text are removed in the second phase of online handwriting recognition i.e. pre-processing phase.
The pre processing phase includes five common steps namely:

- size normalization and centering
- interpolating missing points
- smoothing
- Slant correction.
- resampling of points

This problem of bend or slant in handwriting is solved using the slant correction methods. Duplicate points are redundant and do not contain any information. Thus, these are removed from the captured ink. Resempling of points is done to fix the number of points to be used for recognition and the fixed points are selected in such a way that the original character can be retraced from those points.

SIZE NORMALIZATION AND CENTERING

Size normalization and centering are important processes that are used to recognize a character. The input stroke varies in size and depends upon how the writer moves the pen on the writing pad. The stroke is generally not centered. As the input handwritten texts differ in their size and location, size normalization and centering is required to fix the size of the input text to a constant frame and to place the text at a fixed distance from the origin. This can be achieved by comparing the input character border frame with an already assumed fixed size frame and the input character can be move along to an assumed center location for centering.

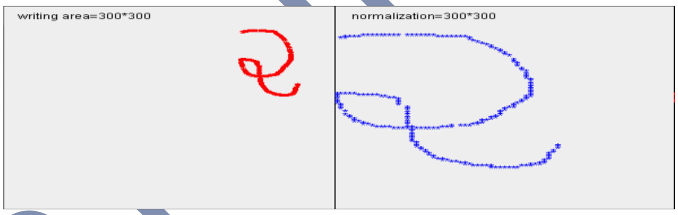

Fig 4. Handwritten characters after normalization

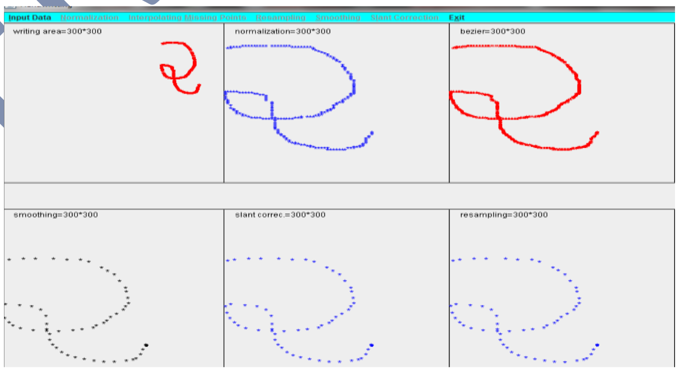

Fig 5. Pre processing steps

C. FEATURE EXTRACTION:-

Feature extraction is a very crucial step as the success of a recognition system is often attributed to a good feature extraction method. The feature extractor determines which properties of the preprocessed data are most meaningful and should be used in further stages. Vertical position of a point, curvature, Pen Up/Pen Down, writing direction, aspect, slope are amongst the various features extracted in [5]. In [6], six scalar features are extracted from each sub stroke. In the present dissertation, two low level features have been extracted namely:

1. DIRECTION ANGLES

Direction angle is the angle that the two successive coordinate points of the input handwritten character make with the x-axis in anti-clockwise direction. After there sampling done on the input character, the number of coordinate points is fixed to 64.These 64 resample points are then used to calculate the direction angles. The direction angles are calculated from point 1 to point 2, then from point 2 to point 3 and so on. The angle for the first point of stroke sample has been considered as zero because of the non availability of previous consecutive point.

2. CURVATURE

Curvature is found by drawing a curve between three points and then drawing a curve joining those three points. Radius of curvature value of that curve is then found using the above mentioned formula. Curvature is found for points taken in triplet e.g. 1, 2 and 3; then 2, 3 and 4; and so on. Total of 64 curvature values are obtained for the pre processed64 resample points. A value of 1000000 is assigned to the curvature in case the line joining the three points is not a curve but a straight line. The curvature for the first poin tof stroke sample has been considered as zero because of the non-availability of previous consecutive point.

D. SEGMENTATION:-

Segmentation is the phase in which data is represented at character or stroke level so that nature of each character or stroke can be studied individually. Segmentation is classified into two categories [8] namely:

- External Segmentation
- Internal Segmentation.

Range of direction angles according to 8-directional code

Range of Angles	Assigned number
Θ (j)>22.5 && Θ (j)<=67.5	1
Θ (j)>67.5 && Θ (j)<=112.5	2
Θ(j)>112.5 && Θ (j)<=157.5	3
Θ(j)>157.5 && Θ (j)<=202.5	4
Θ(j)>202.5 && Θ (j)<=247.5	5
Θ(j)>247.5 && Θ (j)<=292.5	6
Θ(j)>292.5 && Θ (j)<=337.5	7
(Θ(j)>337.5 && Θ (j)<=360) ||(Θ(j)>=0.0 && Θ (j)<=22.5)	8

E. RECOGNITION OF NUMERALS:-

The process has been implemented on the handwritten Devanagari numeral data collected in this work. Numerals are written by ten writers. Devanagari numeral system consists of a set of ten numeral symbols .Each writer was asked to write ten samples of each numeral. Thus, 1000 samples in all are created during this work. The co-ordinate points of the input handwritten character are collected as the character is written. The input handwritten character then undergoes pre-processing and the pre-processing phase resample points are obtained. These resample points are then fed to the feature extraction phase where two low level features i.e. direction angle and curvature are extracted. Together all this data i.e. x, y coordinate points, direction angle and curvature for each point of each character are stored in .csv file. The data from this file is then fed to the next phase i.e. recognition phase. In the second level of experimentation, the data that is fed into the recognition phase is partitioned into the following six schemes:-

SCHEME 1: x, y, direction angle, curvature
SCHEME 2: x, y and direction angle
SCHEME 3: x, y and curvature
SCHEME 4: direction angle and curvature
SCHEME 5: direction angle only
SCHEME 6: curvature only

F. SVM KERNELS

Linear kernel: Linear SVM is linearly scalable with the size of the training data set. It in by the following formula:

k(x, y)= x* y

Where, k(x, y) is the kernel function

Polynomial kernel: It is a non-stationary kernel which is well suited for problems where the training data is normalized. It is given by the following formula

k (x, y) □□□□□x^{T} y □□c)^{d}

Where, α is the slope, c is the constant term and d is the polynomial degree.

RBF kernel: It is defined on the interval [-1, 1] and is given by the following formula

K(x, y) □□exp (□‖ x □□y ‖^{2}/2□$^{□}$□)

Sigmoid kernel: With gain κ and offset ϴ , the formula for a sigmoid kernel is given by:

k(x, y) □□tanh(□(x.y) □□□)

In feature extraction, the edges of the segmented and morphologically filtered image are found. Canny edge detector algorithm is used to find the edges of the image. Then a contour tracking algorithm is applied to track the contour.

RESULTS

Results obtained by the recognition using SVM for all the four kernels for each of the six data schemes are depicted as below. 75% of the data is used for training the system and the rest 25% is used for testing.

SCHEME 1

In this scheme, all the four features i.e. x coordinate, y coordinate, direction angle and curvature are used for recognition. The results obtained by this recognition are shown.

Recognition accuracy with SCHEME 1

Linear	98.700
Polynomial	98.100
RBF	96.500
Sigmoid	10.000

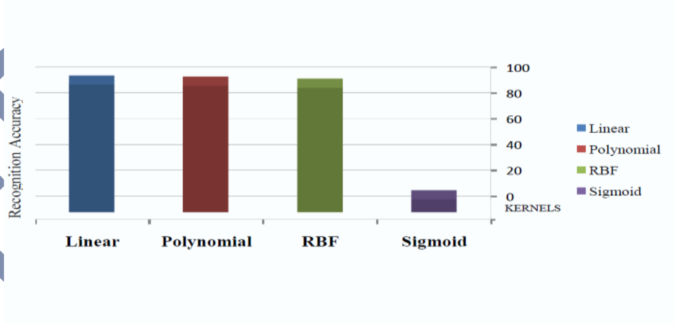

Fig. 6.1 Recognition of numerals with SCHEME 1

SCHEME 2

In this scheme, x-y coordinate of a point along with the direction angle are considered for each point in the input handwritten character. The results obtained by this recognition are shown in Table. and Figure 6 graphically depicts these results.

Recognition accuracy with SCHEME 2

Linear	98.900
Polynomial	98.100
RBF	98.200
Sigmoid	97.800

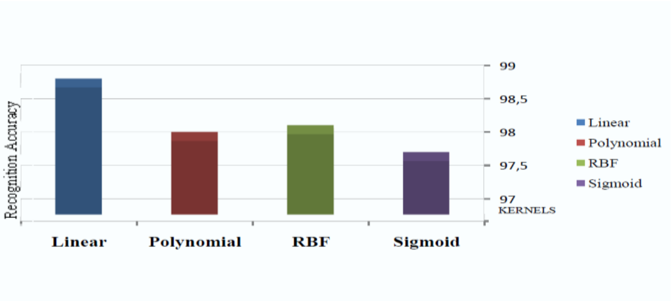

Fig. 6.2 Recognition of numerals with SCHEME 2

SCHEME 3

In this scheme, x-y coordinate of a point along with the direction angle are considered for each point in the input handwritten character. The results obtained by this recognition are shown in Table. and Figure 8 graphically depicts these results.

Recognition accuracy with SCHEME 3

Linear	97.400
Polynomial	97.400
RBF	96.900
Sigmoid	10.000

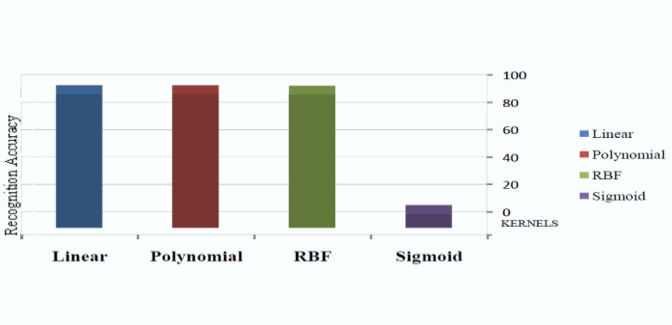

Fig.6.3 Recognition of numerals with SCHEME 3

SCHEME 4

In this scheme, the coordinate points are not considered. Only the direction angle and curvature values are considered from the data collected. The recognition results obtained with this scheme are

Recognition accuracy with SCHEME 4

Linear	97.400
Polynomial	97.400
RBF	96.900
Sigmoid	10.000

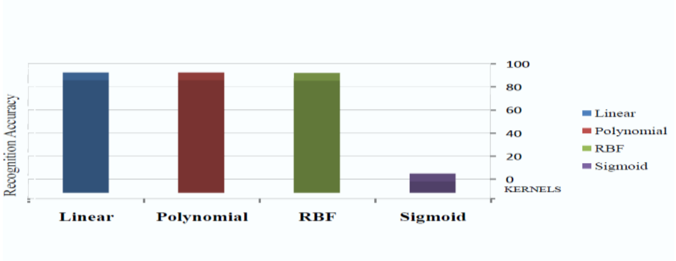

Fig.6.4 Recognition of numerals with SCHEME 4

SCHEME 5

In this scheme, only the direction angles are considered. Table depicts the recognition results are

Recognition accuracy with SCHEME 5

Linear	97.400
Polynomial	97.400
RBF	96.900
Sigmoid	10.000

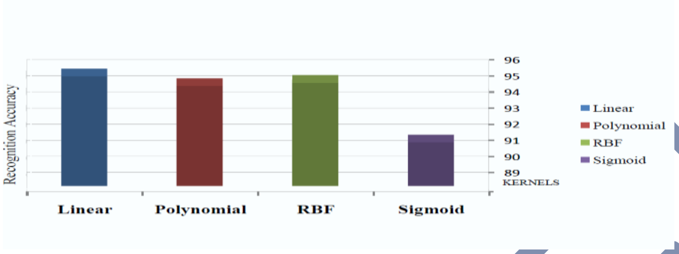

Fig.6.5 Recognition of numerals with SCHEME 5

SCHEME 6

In this scheme, only the curvature is considered. the recognition results are

Recognition accuracy with SCHEME 6

Linear	97.400
Polynomial	97.400
RBF	96.900
Sigmoid	10.000

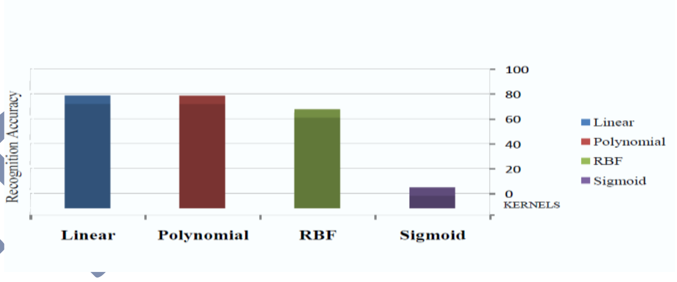

Fig.6.6 Recognition of numerals with SCHEME 6

CONCLUSION

In this paper the main goal is online handwritten Devanagari numeral recognition system. This thesis describes the pre-processing, feature extraction and the recognition phase. Pre-processing and feature extraction are done prior to recognition to increase the efficiency of the character recognition system. Direction angle and curvature are the two features extracted. Recognition is done using four kernel functions of SVM by dividing the data into six schemes depending on the features extracted. Good recognition accuracies have been obtained for all the six schemes and the kernels. Results obtained are reasonably good when the linear kernel is used as compared to the other kernels. The highest accuracy shown by the linear kernel is 98.900%. The results also prove that direction angle and curvature are two very important features and enhance the recognition process. The two features showed good results even when used individually for character recognition especially the direction angles. In this paper, we have reported the various work done on Gurumukhi script. We have organized the review around work have done on handwritten characters/numerals. In this review paper, compared the various feature extraction techniques, classifiers & different datasets which are used in Gurumukhi script for improving the recognition performance. [22]

REFERENCES

[1] M, Sharma, R.K & Jindal, M.K Kumar, "Efficient feature extraction techniques for offline handwritten Gurmukhi character recognition," National Academy Science Letters, vol. 37(4), pp. 381-391, 2014.

[2] Raja shekararadhya, S. V. and Ranjan, P. V., 2009. Support Vector Machine Based Handwritten Numeral Recognition of Kannada Script. IEEE International Advance Computing Conference, pp. 381-386.

[3] Jayaraman, A., Sekhar, C. C. and Chakravarthy, V. S., 2007. Modular Approach to Recognition of Strokes in Telugu Script. Ninth international Conference on Document Analysis and Recognition, vol. 1, pp. 501-505.

[4] R., Kolhe, S.R., Patil, P. M., & Pal, U. Jayadevan, "Offline recognition of Devanagari script: A survey.," Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 41(6), pp. 782-796, 2011.

[5] M., Jindal, M. K., & Sharma, R. K. Kumar, "K-nearest neighbor based offline handwritten Gurumukhi character recognition.," In Image Information Processing (ICIIP), 2011 International Conference on IEEE, pp. 1-4, November 2011.

[6]Mohiuddin, S., Bhattacharya, U. and Parui, S. K., 2011. Unconstrained Bangla Online Handwriting Recognition based on MLP and SVM. Proceedings of the Joint Workshop on Multilingual OCR and Analysis for Noisy Unstructured Text Data, Article. 11, pp. 1- 6.

[7] D., & Jhajj, P. Sharma, "Recognition of isolated handwritten characters in Gurumukhi script," International Journal of Computer Applications, vol. 4(8), pp. 9-17, 2010.

[8]M., Jindal, M. K., & Sharma, R. K. Kumar, "Classification of characters and grading writers in offline handwritten Gurumukhi script," in In Image Information Processing (ICIIP), 2011 International Conference on, IEEE, 2011, pp. 1-4.

[9] M., Jindal, M. K., & Sharma, R. K. Kumar, "Offline Handwritten Gurumukhi Character Recognition Using Curvature Feature," in In Proc. of International Conference on AMOC, 2011, pp. 981-989.

[10] M., Sharma, R. K., & Jindal, M. K. Kumar, "SVM Based Offline Handwritten Gurmukhi Character Recognition,"

[11] K. S., Jangid, M., Dhir, R., & Rani, R. Siddharth, "Handwritten Gurmukhi Character Recognition Using Statistical and Background Directional Distribution," 2011.

[12] D. V., Lehal, G. S., & Kathuria, P. Sharma, "Digit extraction and recognition from machine printed Gurmukhi documents," In Proceedings of the International Workshop on Multilingual OCR, ACM, p. 12, July 2009.

[13] Aggarwal. A., & Dhir. R. Singh. S., "Use of Gabor Filters for Recognition of Handwritten Gurmukhi Character," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 5(2), pp. 234-240, 2012.

[14] P., & Budhiraja, S. Singh, "Offline Handwritten Gurumukhi Numeral Recognition using Wavelet Transforms," International Journal of Modern Education and Computer Science (IJMECS), vol. 4(8), p. 34, 2012.

[15] K. S., Dhir, R., Rani, R., Jangid, M., Dhir, R., Rani, R., & Singh, K. Siddhartha, "Comparative Recognition of Handwritten Gurmukhi Numerals Using Different Feature Sets and Classifiers," In Proceedings of International Conference on Image Information Processing (ICIIP 2011), 2011.

[16] S., & Dhir, R. Singh, "Recognition of Handwritten Gurumukhi Numeral using Gabor Filters.," International Journal of Computer Applications, vol. 47(1), pp. 7-11, 2012.

[17] G. S. R., & Dhir, R. Rani, "Handwritten Gurumukhi Numeral Recognition using Zone-based Hybrid Feature Extraction Techniques.," 2012.

[18] Navghane, A., & Dandawate, Y. Dixit, A., "Handwritten Devanagari character recognition using wavelet based feature extraction and classification scheme," in In India Conference (INDICON), 2014 Annual IEEE, 2014, pp. 1-4.

[19] Britto AS, Sabourin R, Bortolozzi F, Suen CY (2001) A two-stage hmm-based system for recognizing handwritten numeral strings. In: Proceedings of the 6th international conference on document analysis and recognition, Seattle, 10–13 September 2001, pp 396–400

[20]. Bunke H, Roth M, Schukat-Talamazzini EG (1995) Offline cursive handwriting recognition using hidden markov models. Patt Recog 28(9):1399–1413

[21] Chen MY, Kundu A, Zhou J (1994) Off-line handwritten word recognition using a hidden markov model type stochastic network. IEEE Trans Patt Anal Mach Intell16(5):481–496

[22]. Cho W, Kim JH (1994) Off-line recognition of cursivewords with network of hidden Markov models. In:Proceedingsof the 4th international workshop on the frontiersof handwriting recognition, Taipei, Taiwan, 7–9 December1994, pp 410–417