

Implementation of Character Recognition using Hidden Markov Model

Karishma Tyagi, Vedant Rastogi

Department of Computer Science & Engineering, IET,
Alwar, Rajasthan-273010, U. P., INDIA

Abstract- This paper describes a complete system for the recognition of isolated hand written character as well as streams of images by using counter algorithm and Hidden-Markov model (HMM). The HMM has the property that its states are not defined as a priori information, but are determined automatically based on a database of handwritten numerals images. In this paper we have find the result of basic character recognition using HMM, we have also try to find out the wrong character recognition for colored text, colored images for character recognition as well as we have also checked for feature vector by default character recognition used to compare the various algorithms used.

Keywords- HMM, segmentation, neural network, character recognition, counter algorithm, otsu image threshold algorithm.

I. INTRODUCTION

Highlight in 1950's [1], applied throughout the spectrum of industries resulting into revolutionizing the document management process. Optical Character Recognition or OCR has enabled scanned documents to become more than just image files, turning into fully searchable documents with text content recognized by computers. Optical Character Recognition extracts the relevant information and automatically enters it into electronic database instead of the conventional way of manually retying the text. Optical Character Recognition is a process by which we convert printed document or scanned page to ASCII character that a computer can recognize.[3] The document image itself can be either machine printed or handwritten, or the combination of two.

OCR has three processing steps, Document scanning process, Recognition process and Verifying process. In the document scanning step, a scanner is used to scan the handwritten or printed documents. The quality of the scanned document depends up on the scanner. So, a scanner with high speed and color quality is desirable. The recognizing process includes several complex algorithms and previously loaded templates and dictionary which are crosschecked with the characters in the document and the corresponding machine editable ASCII characters. The

verifying is done either randomly or chronologically by human Intervention. Difference in font and sizes makes recognition task difficult if preprocessing, feature extraction and recognition are not robust. There may be noise pixels that are introduced due to scanning of the image. Besides, same font and size may also have bold face character as well as normal one. Thus, width of the stroke is also a factor that affects recognition. Therefore, a good character recognition approach must eliminate the noise after reading binary image data, smooth the image for better recognition, extract features efficiently, train the system and classify patterns.

Segmentation of a document into lines and words and of words into individual characters and symbols constitute an important task in the optical reading of texts. Presently, most recognition errors are due to character segmentation errors (1). Very often, adjacent characters are touching, and may exist in an overlapped. Therefore, it is a complex task to segment a given word correctly into its character components. The process of hand writing recognition involves extraction of some defined characteristics called features to classify an unknown handwritten character into one of the known classes. A typical handwriting recognition system consists of several steps, namely: preprocessing, segmentation, feature extraction, and classification, several types of decision methods, including statistical methods, neural networks, structural matching (on trees, chains, etc). The stochastic processing (Markov chains, etc.) have been used along with different types of features [1-5]. The advantage of HMM approach over ANN approach in optical character recognition is that it can be easily extendible to the recognition of handwritten characters.

In this paper, we will discuss how artificial neural network, genetic algorithm and fuzzy logic can be used in optical character recognition for the use of character recognition.

The remaining part of this paper is organized as follows:- In section II, we will discuss the hidden markov model for the character recognition and in section III we describe the technique we have used for character recognition using HMM, the experimental result of technique are given in section IV and in section V, we will conclude the paper and give the future scope of this paper.

II. HIDDEN MARKOV MODEL

A hidden Markov model is a doubly stochastic process, with an underlying stochastic process that is not observable (hence the word hidden), but can be observed through another stochastic process that produces the sequence of observations [1],[4],[6-8]. The hidden process consists of a set of states connected to each other by transitions with probabilities, while the observed process consists of a set of outputs or observations, each of which may be emitted by each state according to some output probability density function (PDF) [9-11]. Depending on the nature of this PDF function several kinds of HMMs can be distinguished.

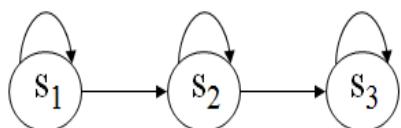


Figure 2.1: Structure of hidden states

III. TECHNIQUE USED FOR CHARACTER RECOGNITION USING HMM

Optical Character Recognition can be applied to recognize text from any multimedia such as image, audio, video. Automatic multimedia recognition is based on the computer vision and pattern recognition application.[1] We can use image processing , character positioning , character segmentation , neural network to solve the problem of image to text recognition.

Using a HMM, we can calculate the hidden states chain, based on the observation chain and using classification algorithm like viterbi alogithm or counter algorithms of hmm ,one can find the most likely result.

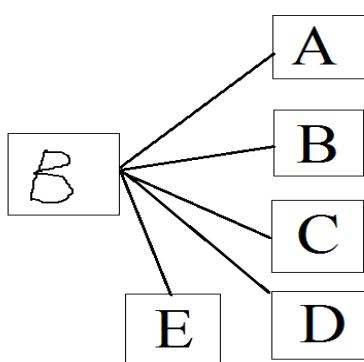


Figure 3.1: figure showing what exactly software looks for

Stage at which HMM contour algorithm is applied as classifier for Recognition

There are two steps in building a classifier: Training and testing. These steps can be broken down further into sub-steps.

1. Training

- Pre-processing* – Processes the data so it is in a suitable form for training.
- Feature extraction* – Reduce the amount of data by extracting relevant information —usually results in a vector of scalar values. (We also need to normalize the features for distance measurements!)
- Model Estimation* – from the finite set of feature vectors, need to estimate a model (Usually statistical) for each class of the training data.

2. Testing

- Pre-processing*
- Feature extraction*
- Classification* – Compare the feature vectors to various models and find the closest match. One can use a distance measure also.

In this paper we have used the pattern classification process for recognizing the characters which is shown below in diagrammatic form.

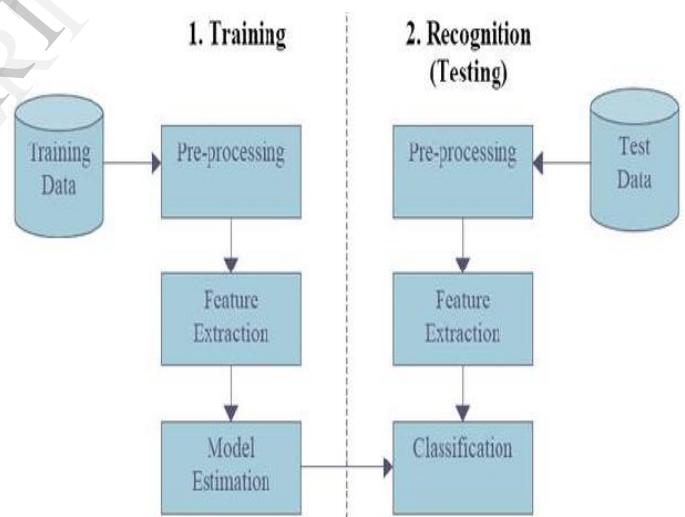


Figure 3.2 : pattern classification process used

We have used the feature extraction process which is given below:

Given a segmented (isolated) character, what are useful features for recognition?

1. Moment based features

Think of each character as a PDF. The 2-D moments of the character are:

$$m_{pq} = \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} x^p y^q f(x, y)$$

From the moments, we can compute features like:

1. Total mass (number of pixels in a binarized character)

2. Centroid - Center of mass

3. Elliptical parameters

4. Eccentricity (ratio of major to minor axis)

5. Orientation (angle of major axis)

6. Skewness

7. Kurtosis

8. Higher order moments

9. Hough and Chain code transform

10. Fourier transform and series

There are different methods for feature extraction or finding an image descriptor, these methods lie into two categories

1. one which uses the whole area of the image

2. other that uses the contour or edges of the object

All the above methods use the contour of the object to collect the object's features.

IV. EXPERIMENTAL RESULTS

We have applied contour algorithm on the edges of printed and handwritten characters. With the help of contour algorithm, a feature vector of an image or text is developed when the character is first trained and this is used to compare with the feature vector of the input during testing or recognition phase. Contour algorithm works on sub-algorithms like feature vector algorithm, sector node algorithm, pixel node algorithm and track node algorithm but if the character is colored this contour algorithm will not give you correct output. For this, the character or input first has to be converted into gray image using otsu image threshold algorithm and then the same process is applied as in hmm counter algorithm for training and testing on the updated image which will give you the correct output for the colored printed and handwritten characters in English and Hindi.

Few imp. code of contour algorithm is illustrated below:

```
Contour(x, x2);
x2.Save(destin.Text) ;
get_stuff(x2,1) ;
picBox1.Image=x2 ;
button1.Enabled=true ;
public void feature_read(int pos,FileStream fileread)
{ //FileStream fileread=new
FileStream(path1.Text, FileMode.OpenOrCreate, FileAccess.
Read);
StreamReader rite=new StreamReader(fileread) ;
string st1;
st1 =rite.ReadToEnd() ;
int i=0,f=0;
string []strVals = st1.Split('v');
for ( i=0;i<6;i++)
{for ( int j=0;j<4;j++)
{for ( int k=0;k<8;k++)
```

```
{ double currVal = double.Parse(strVals[f]);
f++
;featurein[pos].tracks[i].sectors[j].relations[k]=currVal: } } }
```



Figure 4.1: input for the image

Feature vector for the input is again used during testing. Here is a part of code illustrating it.

```
featurein = new cfeature_vector[size];
distince=new double [size] ;
for (int i=0; i<size; i++)
{featurein[i]=new cfeature_vector() ;
```



Figure 4.2: feature vector developed



Figure 4.3: Recognition of the input using contour algorithm for black and white character along with its total mass

For the colored text, contour algorithm gives the wrong output as shown below:



Figure 4.4: input for the image



Figure 4.5: figure illustrating the contour algorithm with wrong output for colored text

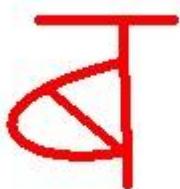


Figure 4.6: colored input for handwritten character



Figure 4.7: Figure illustrating the conversion of colored image into grayscale image using image threshold algorithm

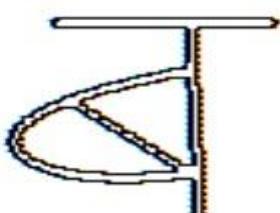
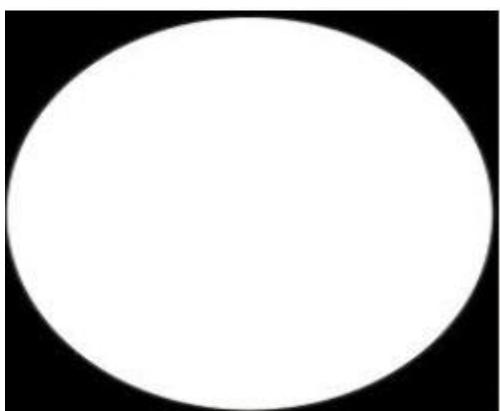


Figure 4.8: Feature Extraction and recognition using counter algorithm with otsu threshold algorithm for colored text with correct output

It is important to note here that the output here is recognized as 'ba'. It is because we have used 'ba' in code for the feature vector of Hindi character during training.

V. CONCLUSION AND FUTURE SCOPE

In this present work we have proposed an HMM based approach for recognition of isolated handwritten Devnagari characters as well as English characters along with the total mass of character. The recognition result obtained from this work varies from character to character. In HMM, contour algorithm is used for training and testing. If input is colored character, the above algorithm will give wrong output, so for that we have implemented otsu image threshold algorithm. There are still some problems regarding the letter segmentation. Sometimes adjacent letters are joined to each other in such a way that they cannot be vertically separated in the normal vertical histogram approach.

REFERENCES

- [1] Gang Wang and Guodong Wang, An Energy Aware Geographic Routing Protocol for Mobile Ad Hoc Networks, *Int J Software Informatics*, Vol. 4, No. 2, June 2010, pp. 183-196.
- [2] Adel Gaafar A. Elrahim and et al., An Energy Aware WSN Geographic Routing Protocol, *Universal Journal of Computer Science and Engineering Technology*, 1(2), 105-111, Nov. 2010.
- [3] S.Corson and J. Macker, "Routing Protocol Performance Issues and Evaluation Considerations," Naval Research Laboratory, Jan.1999.
- [4] B. Karp and H. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in the Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM), pp.243-254, Boston, August 2000.
- [5] Ko Y, aidya NHV. Location-aided routing (LAR) in mobile ad hoc networks. Proc. The ACM/IEEE International Conference on Mobile Computing and Networking, 1998. 66{75}.
- [6] Ma XL, Sun MT, Zhao G, et al. An efficient path pruning algorithm for geographical routing in wireless networks. *IEEE Trans. Vehicular Technology*, 2008, 57(4): 2474{2488}.
- [7] Kim Y J, Govindan R, Karp B, et al. Geographic routing made practical. Proc. the 2nd Symposium on Networked Systems Design and Implementation, 2005. 217{230}.
- [8] Watanabe M, Higaki H. No-Beacon GEDIR: Location-Based Ad-Hoc Routing with Less Communication Overhead. Proc. the International Conference on Information Technology, 2007.
- [9] Singh S, Woo M, Raghavendra CS. Power-Aware routing in mobile ad hoc networks. Proc. the ACM/IEEE International Conference on Mobile Computing and Networking, Oct. 1998.
- [10] Basagni S, Chlamtac I, Syrotiuk VR. A distance routing effect algorithm for mobility (DREAM). Proc. the ACM/IEEE International Conference on Mobile Computing and Networking, 1998.
- [11] Kuhn F, Wattenhofer R, Zhang Y, et al. Geometric ad-hoc routing: Of theory and practice Proc. the 22nd ACM Symposium on Principles of Distributed Computing, 2003. 63-72.
- [12] Zeng K, Ren K, Lou W, et al. Energy Aware Geographic Routing in Lossy Wireless Sensor Networks with Environmental Energy Supply. Proc. the 3rd International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, Waterloo, Canada, Aug. 2006.
- [13] Stojmenovic I. A scalable quorum based location update scheme for routing in ad hoc wireless networks. Technical Report TR-99-09, SITE, University of Ottawa, Sep. 1999.
- [14] Stojmenovic I. Home agent based location update and destination search schemes in ad hoc wireless networks. Technical Report TR-99-10, SITE, University of Ottawa, Sep. 1999.
- [15] Li J, Jannotti J, Douglas S J De Couto, et al. A scalable location service for geographic ad hoc routing. Proc. the 6th Annual International Conference on Mobile Computing and Networking, Aug. 2000. 120-130.
- [16] Kuruvila J, Nayak A, Stojmenovic I. Progress and location based localized power aware routing for ad hoc and sensor wireless

networks. International Journal of Distributed Sensor Networks, 2006, 2(2): 147-159.

[17] Kim YJ, Govindan R, Karp B, *et al.* Lazy Cross-Link Removal for Geographic Routing. Proc. the ACM Conference on Embedded Networked Sensor Systems, Nov. 2006. 112-124.

[18] Y. -C. Tseng, S. -L. Wu, W. -H. Liao and C. -M. Chao, "Location Awareness in Ad Hoc Wireless Mobile Networks," *IEEE Computer*, Vol. 34, No. 6, June 2001, pp. 46-52

IJERT