IASET: Journal of Electronics and
Communication Engineering (IASET: JECE)
ISSN(P): Applied; ISSN(E): Applied
Vol. 1, Issue 2, Jul – Dec 2016, 1-6

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

STOCHASTICALLY OPTIMIZED HANDWRITTEN CHARACTER RECOGNITION SYSTEM USING HIDDEN MARKOV MODEL

SAURABH TALREJA

Assistant System Engineer, Tata Consultancy Services, Chennai, Tamil Nadu, India

ABSTRACT

Handwritten documents form the basis for communication. Due to large variations in the handwriting and the given ubiquity, it is difficult to recognize handwritten documents using machine. HMM models are used in identification, recognition and prediction system because they are very rich in there mathematical structure and thus forms a basis for quantitative proof. This paper proposes the application of augmented Hidden Markov Model for handwritten English character recognition. The model used here utilizes the pre-segmented and noise isolated images of the handwritten characters. After preprocessing, which includes the Binarization, Inversion and Skeletonization, local feature vector is extracted which is deployed for the use by the HMM. The experimental result shows that this augmented procedure is promising and yield more correct results than other conventional methods.

KEYWORDS: (HMM) Hidden Markov Model, Inversion, Local Feature Extraction

INTRODUCTION

Character recognition is an important topic in the field of pattern recognition, which is the backbone of machine perception. Invention of scanners and improvement in scanner technology has enriched the applications of automated character recognition. We have two type of recognition systems based on the way of capturing the handwritten characters: On-line, where data will be captured during the process of writing and Off-line, where data will be captured after the process of writing is over. This is the case which makes the recognition of off-line handwriting more complex than the on-line case. Off-line and on-line recognition systems are also discriminated by the applications they are devoted to. The off-line recognition is dedicated to bank check processing, mail sorting, reading of routine commercial forms etc.

We have used the most popular classifier for handwritten English alphabet recognition, the Hidden Markov Model (HMM). There are two reasons why HMM is so popular. First the model is rich in mathematical structure and hence can form the theoretical basis for use in wide range of application. Second the models, when applied properly, works very well in practice in several important application. HMM has its application in the area of speech recognition, bioinformatics, climatology and Acoustics etc mostly two types of problems occurs in these research area of the modeling. The first one is the training problem of HMM to optimize the model parameters such that it can accurately represent the training sequence and Second problem is of testing or prediction of sequence using the optimized HMM model. In this paper, we have taken application of English alphabet recognition in which HMM is used with an augmented procedure for recognition. We have applied the HMM in correspondence with the local features extracted of a pre-processed and noise isolated image of handwritten character.

DATA SET INFORMATION

The data set used for the practical application and also in this paper was taken from *http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/* this data set has maintained the 74000 character images thus have its name as 'chars74K'. The data set taken have more than 50 images of each character from which ten are used for training the HMM model optimizing its parameters and from the rest images six are used for testing the model's recognition rate.

THE HIDDEN MARKOV (HMM)

HMM is a probabilistic model useful for finite state stochastic sequence structures. Stochastic sequences are called observation sequences, i.e. . O = $o_1 o_{2.}$oT, where T is the length of the observed sequence. HMM with n states (S_1, S_2…...S_N) can be characterized by a set of parameters (A, B, π). λ is called the model of HMM, i.e. λ = (A, B, π).

In order to characterize an HMM completely following elements are needed [1] [2] [3] [4]

N: The number of states in the mod el

M: The number of distinct observation symbols M per state

A: The state transition probability distribution A = { a_{ij} }, $a_{ij} = p(q_t = s_j \mid q_{t-1} = S_i)$

B: The observation symbol probability distribution in state j B = $b_j(k) = P(V_k\ at\ t \mid q_t = S_j)$

π: The initial state distribution $\pi_i = P(q_1 = S_i)$

The three main problems of HMM are:

Evaluation problem: *Compute P (O| λ), the probability of the observation sequence O = $O_1\ O_2\ O_3 \ldots O_T$, given the model λ = (A, B, π).*

Decoding problem: In this, we attempt to uncover the hidden part of the model i.e. find the optimal state sequence, for the given observation sequences O = $O_1\ O_2\ O_3 \ldots O_T$, give n the model λ = (A, B, π).

Learning problem: Model parameters (A, B, π) are adjusted such that P (O| λ) is maximized.

In this paper, we have considered the third HMM problem i.e. the *Learning problem* or often called as *training and prediction problem* and tried to solve it.

PRE-PROCESSING STAGE

In Pre-processing stage we take an image and crop it to adjustable size. Then we convert this image into binary format where the intensity values are specified either 0 or 1. This is called Binarization of the image (refer to Figure-1 and Figure-2).

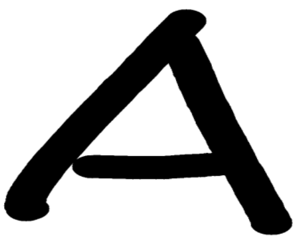

Figure 1: Sample Image for English Character 'A'

1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1
1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1
1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1
1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	1
1	1	1	1	1	1	0	0	0	0	0	0	1	1	1	1
1	1	1	1	1	0	0	0	1	1	0	0	1	1	1	1
1	1	1	1	0	0	0	1	1	1	0	0	1	1	1	1
1	1	1	1	0	0	1	1	1	1	0	0	0	1	1	1
1	1	1	0	0	0	1	1	1	1	1	0	0	1	1	1
1	1	0	0	0	1	1	1	1	1	1	0	0	0	1	1
1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1
1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0

Figure 2: Binarization

After binarizing the image we resize it in a 16x16 matrix then perform the Inversion (refer Figure 3) where we exchange 0 with 1 and 1 with zero. This is done so because if we not invert the intensity values we cannot apply the morphing process. Since the data set that we have taken has images with thick contour then we have to apply the Thinning process which is often referred as Skeletonization of the image (refer figure 4).

0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	1	1	1	0	0	1	1	0	0	0	0
0	0	0	0	1	1	1	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0	1	1	1	0	0	0
0	0	0	1	1	1	0	0	0	0	0	1	1	0	0	0
0	0	1	1	1	0	0	0	0	0	0	1	1	1	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	1	1	0	1	1	1	1	1	1	1	1	1	1	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0
1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0
1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Figure 3: Inversion

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0
0	1	0	0	0	1	1	1	1	1	1	0	1	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Figure 4: Skeletonization

FEATURE EXTRACTION

For a given classifier, better feature extraction yields the better recognition rates. Local Feature Extraction Method is used to extract the local information contained in the preprocessed character image. We have divided the image in eight equal blocks of 4x8 and then, from each block eight gradient features are extracted. Finally observation sequence contains 64 observations at the end (refer Figure 5).

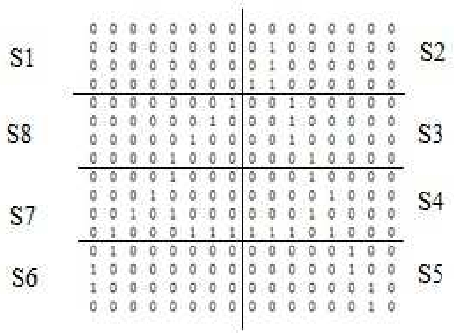

Figure 5: Local Feature Extraction

We have used *Forward algorithm* [1] to train our HMM model using observation sequence O obtained from the local feature extraction. At the end of training process we obtain the optimized value of A and B which is used for recognition purpose. The forward algorithm decode the observation sequence and provides the values of P (O| λ) using the HMM model. We have trained HMM model in first trail for 1000 times and the values of P (O| λ) are listed below:

TRAINING HMM MODELS

Training is also termed as Learning Process for HMM Model. We have taken twenty six unique HMM model's each corresponding to an English alphabet. We have used *Forward algorithm* [1] to train our HMM models using observation sequence O obtained from the local feature extraction of the pre-processed image. Let us suppose λ_i denotes the HMM model for the i^{th} English alphabet where i=1, 2… 26 and for each λ_i we have a training set T_i having j different images of i^{th} English alphabet. Following procedure is deployed in the practical application.

Train_HMM (i)

- For each image k in Training set T_i where k=1, 2…, j.
    - Calculate the observation sequence $O = o_1o_2o_{3\ldots}o_t$, where t is the length of the observation sequence
    - *Max* = *Decode sequence* (i, O)
    - Store the *Max* value.
- Store the maximum value of P (O| λ) among j values.
- Store M = mean of the j values of P(O| λ)
- Store SD = standard deviation of j values of P(O| λ)
- Create range for character i as [M-SD, M+SD]

Decode Sequence (i, O)

- Take a small value d (say in range of 0.001 to 0.009)
- Initialize x=100
- While $x>0$
    - Alter λ_i with amount d

- Decode the observation sequence *O* by using the Forward Algorithm and get the value of P($O|\lambda$)
- Store the value of P($O|\lambda$)
- $x = x-1$

- If the values P(O| λ) changes significantly go to step- (1)
- Return the maximum value of P ($O|\lambda$) till obtained.

RECOGNITION PROCESS

For testing we input an unforeseen character image. Pre-process it and generate the observation sequence *O* as described earlier. Following procedure is deployed in the practical application.

Testing_by_HMM ( )

- For i= 1,2…, 26
- *x = Decode sequence (i, O)*
- if (*x* lie in the range created for i^{th} English character)
- Matched with i^{th} English character

EXPERIMENTAL RESULTS

We have taken 10 sample images for each alphabet, pre-processed them and have trained the corresponding HMM model's by using the above described procedures. For testing we have taken 6 different images of each English alphabet and using the same testing procedure described earlier tested them. The accuracy of recognition is plotted in Figure 6.

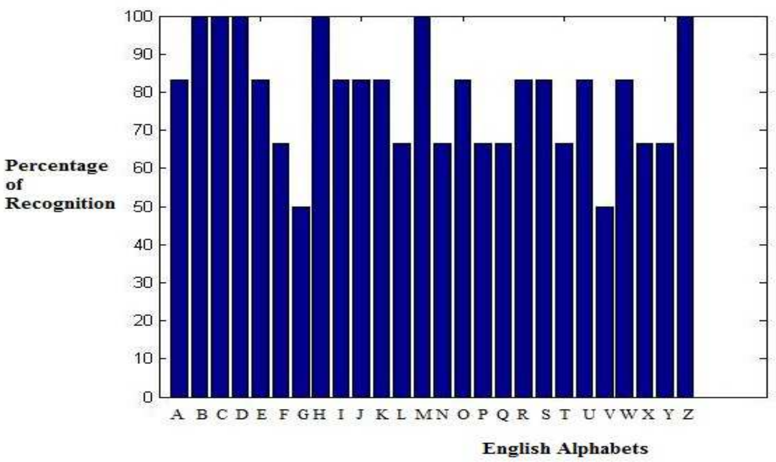

Figure 6: Experimental Results

CONCLUSIONS

The objective taken in this paper is to present handwritten English alphabet recognition, which can recognize alphabets with the same recognition accuracy as humans, but at a faster rate. In the present work we have devised novel method. In the pre-processing, local feature extraction methods have been used which helps to create the feature vector. These features have been used to get HMM model.

The models have been tested considering all the alphabets and the recognition rate found in between 79.615 to 99.99%. It is felt that, finding additional hidden unique features in the English alphabets can further increase the recognition rate. A combination of HMM along with neural network classifier or with genetic approach for optimizing the HMM parameters can also be considered and recognition may become better.

REFERENCES

1. Rabiner L.R.,"A tutorial on HMM and Selected Applications in Speech Recognition", Proceedings of the IEEE, Vol. 77, NO. 2, P267-296.
2. R. Sundararajan, "Stock market trend and predection using Markov models", Department of ECE, Amrita Vishwa vidyapeeth, Coimbatore, RTCSP 09.
3. S. Kwong,C.W. Chau, K.F. Man, K.S. Tang, "Optimization of HMM topology and its model parameters by genetic algorithm", Pattern recognitions 34 (2001) 509-522.
4. The Metropolis Algorithm, "Statistical Systems and Simulated Annealing".