Online Character Recognition using Hidden Markov Model

A dissertation submitted in partial fulfillment
of the requirements of M.Tech. (Computer Science)
degree of Indian Statistical Institute, Kolkata
by

Ramu Padala

under the supervision of

Utpal Garain

Indian Statistical Institute
203, Barrackpore Trunk Road
Kolkata-700 108.

July 2002

Indian Statistical Institute

203, Barrackpore Trunk Road,

Kolkata-700 108.

Certificate of Approval

This is to certify that this thesis titled **Online Character Recognition using Hidden Markov Model** submitted by **Ramu Padala** towards partial fulfillment of requirements for the degree of M.Tech in Computer Science at Indian Statistical Institute, Kolkata embodies the work done under my supervision.

Utpal Garain 26/7/02

Utpal Garain
Computer Vision and Pattern Recognition,
Indian Statistical Institute,
Kolkata-700 108.

26.07.02
(External Examiner)

Acknowledgments

I take pleasure in thanking Utpal Garain for his friendly guidance throughout the dissertation period. His pleasant and encouraging words have always kept my spirits up.

I take the opportunity to thank my family ,my friends and my classmates for their encouragement to finish this work.

Ramu Padala

Contents

Chapter 1

Introduction

Nowadays people are interested to interact with computers in their local languages. It is very tedious task to make keyboard for all existing characters and their combinations. Users will be more comfortable to input characters as handwritten rather than keying-in.

The widely accepted approach for online character recognition is to capture a character as a sequence $(x(t), y(t))$ of points while the character is being written on a digitizer tablet using a pen kind of tool.

This is the only information available for a character. Each point of the sequence can be coded as a finite set of features like direction, pen-ups, velocity of the pen etc$\cdots$. Different people have used different features. We have used direction code and length of the segment.

There are many different methods have been proposed for character recognition. We have used Hidden Markov Model(HMM) to recognize a character.

Chapter 2

Description of Hidden Markov Model

There are N number of states $s_1, s_2, \cdots, s_N$. We have given the probabilities of going from any state s_i to any state s_j, where $i, j \in \{1, 2, \cdots, N\}$. At each and every state there are m observation symbols $O_1, O_2, \cdots, O_m$. Also we have given the probabilities of getting each observation symbol and the probabilities of starting from any state.

Notations :
N = number of states = 24.
M = number of distinct observation symbols at each state $\pi = \{\pi_i\}_{i=1}^{24}$, where π_i is probability of starting from $i'th$ state.

$A = \{a_{ij}\} \ = P(i_{t+1} = j/i_t = i)$
= Probability of being at $j'th$ state at time $t+1$ given that we are in state i at time t.
$B = \{b_i(k)\}$ = Probability of getting $k'th$ observation symbol when we are in state i.

Hidden Markov Model (HMM) is denoted by $\lambda = (A, B, \pi)$, where A, B, π are as defined above.

Problem :
Given any observation sequence $O_{i_1}, O_{i_2}, \cdots, O_{i_l}$ of length l, where $O_{i_j} \in \{O_1, O_2, \cdots, O_m\}$ for $1 \leq j \leq l$. What is the prabability of getting this sequence ?

$$
\begin{aligned}
P(O/\lambda) &= \textstyle\sum_I P(O/I,\lambda)P(I/\lambda) \\
&= \textstyle\sum_I \pi_{j_1} b_{j_1}(O_{i_1}) a_{j_1 j_2} b_{j_2}(O_{i_2}) \cdots a_{j_{l-1} j_l} b_{j_l}(O_{i_l})
\end{aligned}
$$

where I is set of all possible state sequences of length l.
This can be computed with the fallowing procedure.

Forward - Backward Procedure :

Consider the fallowing variable $\alpha_t(i)$ defined as :

$$\alpha_t(j) = P(O_{i_1}, O_{i_2}, \cdots, O_{i_t}, j_t = j/\lambda)$$

i.e., the probability of the partial observation sequence up to time t and the state i at time t. $\alpha_t(j)$ can be computed inductively as fallows.

$$\alpha_1(j) = \pi_j b_j(O_{i_1}), 1 \leq j \leq N$$

for $t = 1, 2, \cdots, l-1,\ 1 \leq j \leq N$

$$\alpha_{t+1}(j) = \left[\sum_{k=1}^{N} \alpha_t(k) a_{kj}\right] b_j(O_{t+1})$$

then we have

$$P(O/\lambda) = \textstyle\sum_{k=1}^{N} \alpha_l(k)$$

Chapter 3

Model Generation

Our model consists of 24 states.We are considering only few basic telugu characters.For each character 20 samples are taken by different persons.HMM will be developed based on these 20 samples for each character. Model generation consists of two steps.First one is preprocessing.In preprocessing we are reducing the number of points of a stroke without disturbing the shape of the stroke.Second one is the generation of HMM.It consists of computing initial and transition probabilities.

Some Basic Telugu Characters:

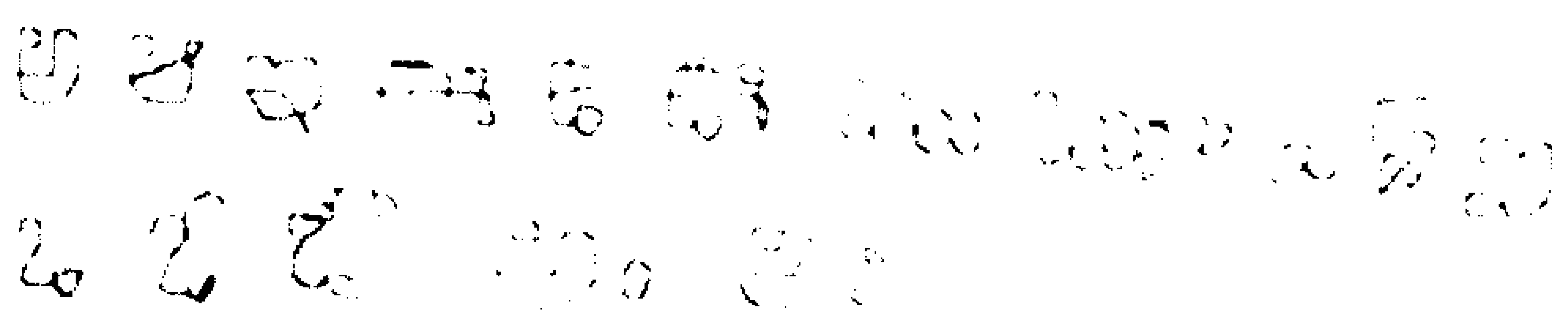

Figure 1: Some Basic Telugu handwritten characters

Variations in Writing:

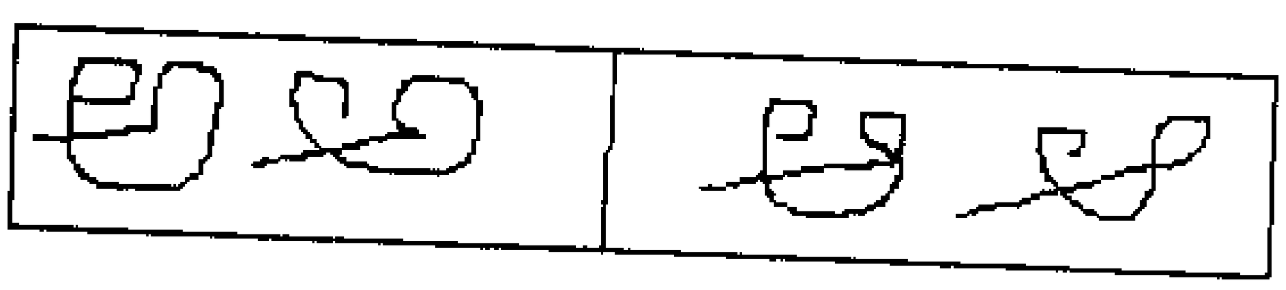

Figure 2:

Direction Codes:

Direction code of a segment $\overline{XY}$ is a number of a region of the following figure in which it falls.

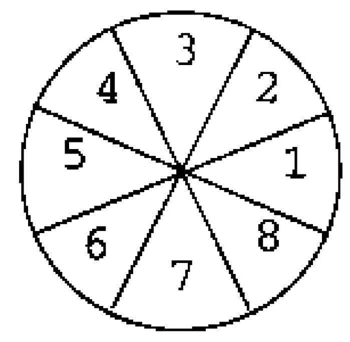

Figure 3:

3.1 Preprocessing

1. for each character
2. for each individual stroke
3. for each three successive points X, Y, Z
4. if absolute value of the difference between angles of $\overline{XY}$ and $\overline{YZ}$ with X-axis is $\leq \frac{\pi}{8}$, remove point Y from the stroke.
5. connect successive strokes to form a single stroke by joining end and beginning points of the corresponding strokes.

3.2 Generating HMM

1. for each character
2. $L = 0$
3. for each pair of successive points X, Y

   $d =$ Find_Dir_Code(X, Y)

   $l =$ Length(X, Y)

   $\theta =$ angle between $\overline{XY}$ and X-axis

   $size =$ Size(X, Y)

   $L = L + l$
4. write $(d, l, \theta, size)$ into structure $VECTOR[i]$
5. $i = i + 1$

6. goto step ~~??~~ 3

7. for $j = 1$ to i
   $VECTOR[j].len = \frac{VECTOR[j].len}{L}$

Find_ Dir_ Code (X, Y)
{
    return the segment number of a segment in which $\overline{XY}$ lies according to the Figure 3
}
Size(X, Y)
{
    if $\overline{XY} \leq 0.003$
        return 1;
    else if $\overline{XY} \leq 0.006$
        return 2;
    else
        return 3;
}
For each charcter there are 24 states. Each state is consisting of (d, s), where d is direction code and s is size.

3.3 Computing Initial Probabilities

1. for each character

2. for i=1 to 24 initialize $ST[i]$ to 0

3.   for each character sample file
        read first tuple $(d, \theta, l, size)$
        $state = 3 \cdot (d - 1) + size$
        increase $ST[state]$ by 1

4. for $i = 1$ to 24
   $\pi_i = \frac{ST[i]}{No\_Samples}$

3.4 Computing Transition Probabilities

1. for each character

2. initialize $TR[\ ][\ ], SG[\ ], theta[\ ], len[\ ]$ to 0

3. for each sample character

   read two successive tuples $(d_1, \theta_1, l_1, s_1), (d_2, \theta_2, l_2, s_2)$ from the file

   $m = 3 \cdot (d_1 - 1) + s_1$
   $n = 3 \cdot (d_2 - 1) + s_2$
   increment $TR[m][n], SG[m]$ by 1
   increment $theta[m], len[m]$ by θ_1, l_1 respectively

4. for $i = 1$ to 24
   for $j = 1$ to 24
   $a_{ij} = \frac{TR[i][j]}{SG[i]}$

5. for $i = 1$ to 24
   $\mu_{i_1} = \frac{theta[i]}{SG[i]}$
   $\mu_{i_2} = \frac{len[i]}{SG[i]}$
   $V_i = \frac{1}{SG(i)}[\sum_{O_t \in i} (O_t - \overline{\mu_i})^T (O_t - \overline{\mu_i})]$
   /* $O_t = (\theta, l)$, $\overline{\mu_i} = (\mu_{i_1}, \mu_{i_2})$ */

3.5 Computing observation probabilities & Testing

1. Get the observation sequence from the character to be tested using the method described earlier.

2. Compute the probability of getting this observation sequence for each character using Forward - Backward Procedure discussed earlier. In this procedure $b_j(O_t)$ can be dynamically computed with the following equation

$$\frac{1}{(2\pi)^{D/2}|V_i|^{1/2}} exp[-\frac{1}{2}(O_t - \overline{\mu_i})V_i^{-1}(O_t - \overline{\mu_i})^T]$$

3. Report the character which gives the maximum probability.

Chapter 4

Results

In our experiment the system has been tested with a few basic Telugu characters. Twenty most frequent characters are considered for testing. For each character 20 samples are taken from two different writers (10 from each). Test results are shown in the following table.

No. of char.	No. of Samples/char	Recognition	Rejection	Misrecognition
20	20	95%	3%	2%

There are few cases in which it is giving wrong results. Some of the characters which look same in the shape and generate confusions during recognition are shown in the following figure.

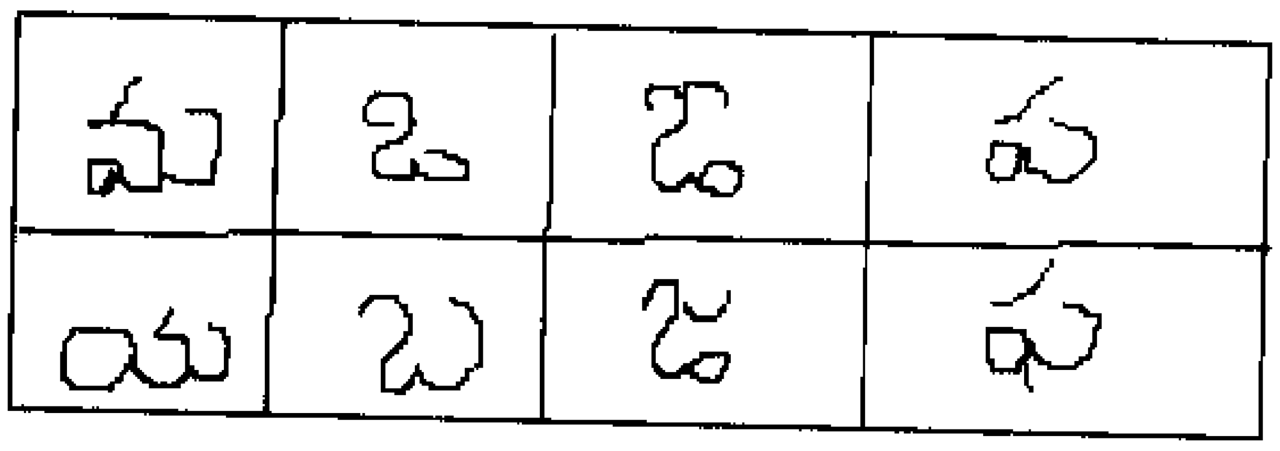

Figure 4: Some similar Telugu characters

4.1 Discussions

HMM is a general model. We can apply HMM to recognise online handwriting in other Indian scripts. In this project, we have used it to recognise online Telugu characters. Input has been taken from only two persons. Better recogntion rate may be possible if we consider variations in handwritten characters collected from sufficiently large number of writers. Fixing the number of states is the main concern in devolopping HMM. We have avoided this problem by giving 24 different states for each character and each state can be uniquely expressed in terms of direction code and length of a segment.

4.2 Acknowledgement

We sincerely acknowledge some useful discussions with Prof. S. K. Parui of CVPR Unit of our institute.

Bibliography

[1] H. Yasudha, K. Takahashi and T. Matsumoto
A Discrete HMM for Online Handwriting Recognition International *Journal of Pattern Recognition and Artificial Intelligence* vol. 14, No. 5(2000) 675-688.

[2] Rakesh Dugad and U.B. Desai
A Tutorial on Hidden Markov Models
Technical Report No. : SPANN-96-1.

[3] Utpal Garain, B.B. Chaudhuri, and T. Pal
Online Handwritten Indian Script Recognition: A Human Motor Function based Framework
to appear in Int. Conf. on Pattern Recognition (ICPR), Quebec City, Canada, 2002.