Ahsanullah University of Science and Technology

Department of Computer Science and Engineering

A Simplified Method for Handwritten Character Recognition from Document Image

Bachelor of Science in Computer Science and Engineering

Group No: 14

Submitted By

Ashfaq Ali Shafin	14.01.04.111
Tarikul Islalam	14.01.04.112
Irtija Ahasan	14.01.04.122
Abid Hasan Prottoy	14.01.04.125

Submitted To

Mr. Mohammad Imrul Jubair

&

Mr. Imtiaz Naved

Introduction

Hand-written character recognition is one of the challenging areas in the field of Image Processing and Pattern Recognition. It has many applications such as: reading aid for blind, conversion of handwritten character into computerized text. This project presents a simple but effective technique for recognizing handwritten character from a machine readable input image file.

Proposed Work

Procedures and dataset, that we followed for this technique is described below.

Dataset

We used *"The Chars74K dataset: Character Recognition in Natural Images"* (http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/) hand-drawn characters. 55 samples per class. The pen stroke trajectories are also provided, so this dataset can also be used to evaluate on-line handwritten character recognition methods.

Extracting Features:

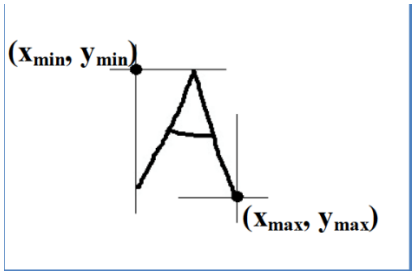
Each image has a white background and black foreground color image. Let **X** be the input character image with size m x n pixels.

1. At first **X** is converted to a grayscale image X_b. As we need to convert the image to binary image.

2. Then the X_b is again converted to binary image. So the background pixels will have the intensity of One(1) and the foreground object intensity will have Zero(0).

3. The foreground character object is extracted from the background. For this purpose, coordinates (x, y) of the black (0) pixels are taken into account and maximum and minimum values are counted among the x and y coordinates where

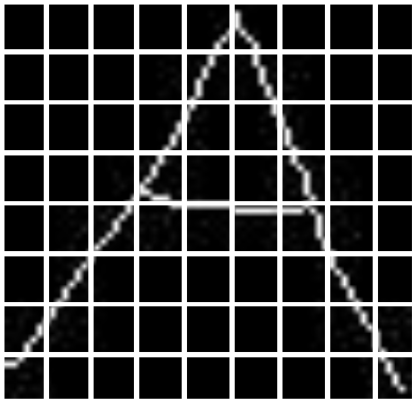
(x_{min}, y_{min}) is the upper left corner's coordinate and (x_{max},y_{max}) is the lower right corner's coordinate. Let $\mathbf{X_E}$ be the image matrix containing extracted object.



4. After $\mathbf{X_E}$ is resized in a global m_r x n_r size. $\mathbf{X_{ER}}$ is the resized image.

5. This resized $\mathbf{X_{ER}}$ is then complemented so the background have the intensity of zero (0) and the foreground have the intensity of one(1). $\mathbf{X_{ERT}}$ is obtained by the morphological thinning operation on inverted $\mathbf{X_{ER}}$.

6. Then, N number of cells are partitioned from $\mathbf{X_{ERT}}$ where each cell size is s_c x s_c.



7. For each cell, proportion of 1's and 0's had been calculated and the value was assigned as the estimated value for the corresponding cell. If cell C1 has total n_w number of 1 values and total n_b number of 0 values, the proportion is

$$\boldsymbol{P_1 = \frac{n_w}{n_b}}$$

P_1 is the estimated value for C_1. Similarly for C_2, C_3,…C_N cells the corresponding estimated values P_2, P_3, …P_N is calculated.

0	0	0	**255**
0	0	0	**255**
0	**255**	**255**	0
255	0	0	0

Here n_w = 5 and n_b = 11, so P_i = 5/11 = 0.454

8. Estimated P_i values are then stored in an array R_x for an individual image where i= 1, 2, 3, … N.

Classification

a. For an induvial training example image $\mathbf{X_{T1}}$, all the feature values ($P_{1,1}$, $P_{1,2,}$ …$P_{1,N}$) in its array R_{X1} is considered as the attributes for training. The class label for $\mathbf{X_{T1}}$ is labeled as "A", class label for $\mathbf{X_{T2}}$ is labeled as "B" and the next labels as "C", "D", "E" … and so on.

b. Suppose a testing sample image $\mathbf{X_{S1}}$ is needed to be classified. For this purpose, array of feature values $\mathbf{R_{XS1}}$ is obtained from $\mathbf{X_{S1}}$ and the array's element is the feature values $Q_{1,1}$ $Q_{1,2}$ $Q_{1,3}$ … $Q_{1,N}$.

c. For classification distances between test sample X_S and every training objects X_{T1}, X_{T2}, … X_{TM} (where M is the total number of training objects). For an example

$$\boldsymbol{Distance_{Euclidean}(X_{T1}, X_{S1}) = D_1 = \sqrt{\sum_{j=1}^{N} (P_{1,j} - Q_{1,j})^2}}$$

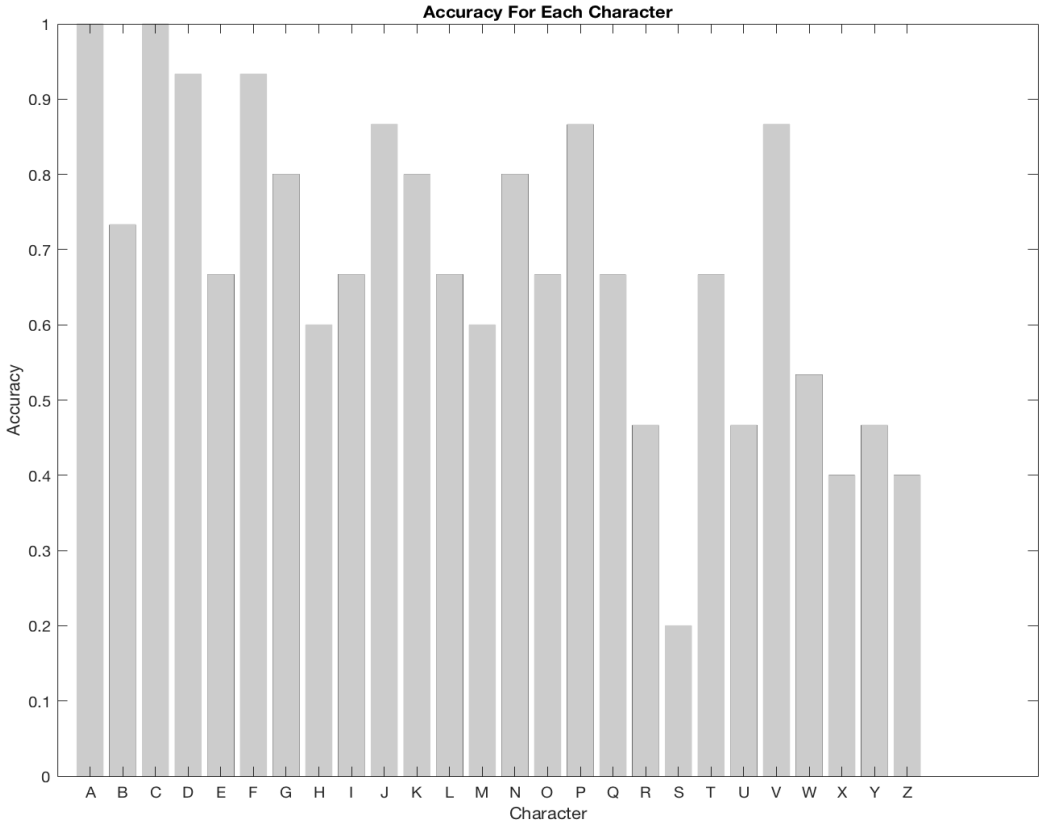
Similarly, all the distances D_2, D_3, … D_M are calculated in order to find the nearest neighbors. To classify the unlabeled test sample, its k nearest neighbors are identified and the class labels of these nearest neighbors are then used to determine the class label of the testing sample object.

We used default implemented K-nn algorithm provided by the Matlab in this case.

Simulation

1. Program written using Matlab.
2. Every Image is resized to global size 64 x 64.
3. Image is divided to 64 cells where each cell size is 8 x8.
4. So, there are total 64 elements in $\mathbf{R_x}$ array. This means each object has 64 number of attributes.
5. Estimated feature values are stored as attributes for training the classifier. There can be several objects under each class to train.
6. We will be using 75:25 ration for training and testing for each character. This means each character will have 40 images for training and 15 images for testing.

Result Analysis



Summary

This project is comparatively easy to implement. The complexity of the above procedures was less than any other handwritten character recognition method. At the same time the result will be crucial to justify how good is this project.