Disambiguated Achieve Rectification

Mahimol Eldhose
Dept. of Applied Electronics
Sri Lakshmi Aammal Engineering College
Chennai

C. Rekha
Faculty of dept. of ECE
Sri Lakshmi Aammal Engineering College
Chennai

ABSTRACT
This paper presents entry for aura response for character recognition and the handwritten or printed text translation into editable text. The objective is to identify handwritten characters with the help of neural networks and facilitates the conversion of handwritten documents to editable text from document images. Handwritten contentedness boasts challenges that are seldom encountered in machine-printed text. The translation basis is either mechanical or electronic translation. This is not easy since different people have different handwriting styles. Assigning distinct templates to each and every alphabet and numbers is the approach described. This concept can be a trademark in data entry applications. The suggested method is simple, have promising discrimination accuracy and less time complexity.

General Terms
Pattern Recognition

Keywords
Handwriting recognition, template, electronic translation

1. INTRODUCTION

Handwriting recognition is comparatively difficult, as different people assault different handwriting styles. The image of the written text may be sensed from a piece of paper by optical scanning or any pronouncement methods. OCR engines are primarily focused on machine printed text and ICR for hand text written in capital notes. There is no OCR engine upon supports handwriting recognition as of today. Handwriting recognition engages the instinctual modifying text in an image into letter ciphers which are usable within computer and text-processing applications. There is a hail to recognize a wide variety of fonts, however handwriting and script fonts that mimic handwriting are still problematic, therefore neural network technology is required. There are different approaches to improve script and handwriting recognition in the way of developments. Neural network technology is utilized to analyze the stroke edge, the line of discontinuity between the text notation, and the background, processes the scans to differentiate between images and text and determine what letters are comprised. Handwritten image analysis and conversion involves the tasks of text block segmentation, text line copping, character cropping and recognition. The choice of training of data and neural network is very significant. With most befitting training pair, neural network can be trained and can get a good end results. The problem of well-defined datasets lies in chosen algorithm attributes. The individual features are extracted one by one and map the target output for training purpose and train it properly to accomplish good results.

2. PREVIOUS WORKS
A method for distinction between machine printed and handwritten character images using directional and symmetric features as a input of a neural network had developed but failed in some exceptional cases. The method for discriminating handwritten text, extracting low level features and classify using feed forward multilayer perception neural network was endeavored by the developers. Automatic separation of machine printed and handwritten text lines for two Indic scripts are developed using structural and statistical features as nodes of tree for tree classifier. Hidden Morkov Models (HMM) for extracting handwritten text words from printed text documents also give rather better results.

Image extracted the gradient and luminance histogram and applied a neural network to segment a gray level document image in to handwritten character. This is what we want in converse effect. Considerable accuracy is obtained using SVM classifier and using filters like Gabor and run length histogram features etc and further improved the result by implementing a Markov Random field based post processing step. A trainable approach to discriminate between machines printed and handwritten text using simple structural characteristics and discriminant analysis for classification is another possible way. There exists some research on discrimination of machine printed and handwritten text. The older concept is simple. The image before giving as password is converted in to its Read, Green and Blue (RGB) values and these values will be normalized using normalization function. Image into matrix or text is used to give as input to the neural network. Conversion of image to matrix reads the color of each pixel of the image. Once changing image into a matrix that consists of set of numbers we can give it to the neural network as input and can train it using username and image matrix as a training sample. In earlier approaches, user has the choice of selecting either text or an image depending on the requirement and security expects.

3. SOLUTION APPROACH
To solve the outlined hand written character recognition, MATLAB computation software with Neural Network Toolbox and Image Processing Toolbox is employed. In this approach basic preprocessing steps are performed like gray scale conversion, threshing, binary conversion, connectivity test to check for the maximum connected component which is the box of the shape. Once locating the box, the individual characters are then cropped into different sub images that are the raw data for the feature extraction routine. The size of the sub-images are not mounted since they are expose to noises, affect the cropping process to be vary from one to another. This will cause the input of the network become not standard and thus, forbid the information from feeding through the network. To resolve this drawback, the sub-images have been resize and then by finding the average value in each 10 by 10

blocks, the image will be right down to five by seven matrices, with fuzzy worth and become thirty five inputs for the network. However, before resize the sub-images, processes must be gone through to eliminate the spaces in the boxes.

a. Dividing text in lines

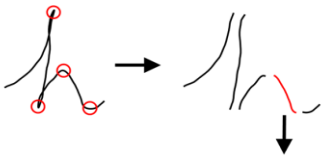
Document image to text lines and words conversion is a critical task towards handwritten document recognition. Existence of overlapping, variable character size causes some challenges to take out text lines. Due to high varying writing styles this method do not use any prior knowledge and adapt to the properties of the document image. First each line is to be identified. It requires that the document should assume in a text line, successive words are not attached to each other. The matrix value of each letter is sorted out by clipping the lines. Each character from the text is separated, for that the image boundary is located, including left, right, top and bottom sides separately. Finding each side one by one is somewhat easy compared to other steps. The inter character spacing is (the space between the letter cropped and the next line) is determined.

b. Read letter and creating templates

The letter is resized to compute the correlation between template and input image which read each letter. The global and local templates are assigned and constructed first. For every English alphabet from A to Z assign a value from 1 successively, same for numbers from 0 to 9, following the alphabets. Pre-assigning templates for the characters are saved for concatenation.

c. Feature Extraction

The sub-images have to be cropped sharp to the border of the character in order to standardize the sub-images. The image standardization is completed by finding the maximum row and column with 1s and with the peak value, increase and reduce the counter till meeting the white space, or the line with all 0s.



Polynomial coefficient

Fig 1: Feature extraction

The image pre-processing is followed by the image resize again to meet the network input requirement, where the value of 1 will be assign to all pixel where all 10 by 10 box are filled with 1s Finally, the matrices is concatenated into a stream so that it can be feed into network 35 input neurons. The input of the network is really the negative image, where the input range is 0 to 1, with 0 equals black and 1 indicate white, while the value in between show the intensity of the relevant pixel. By this, we are able to extract the character and pass to another stage for classification and training of the neural network.

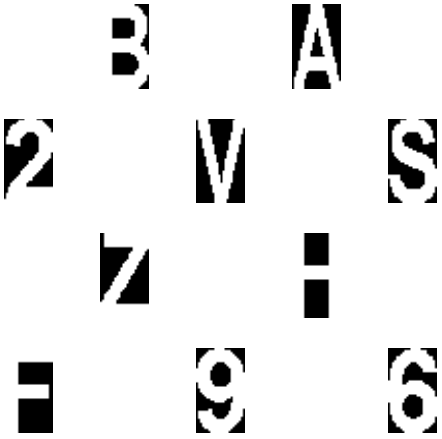
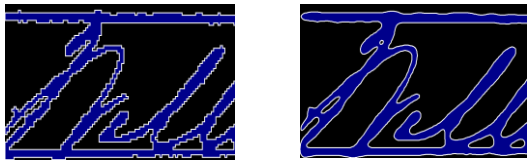


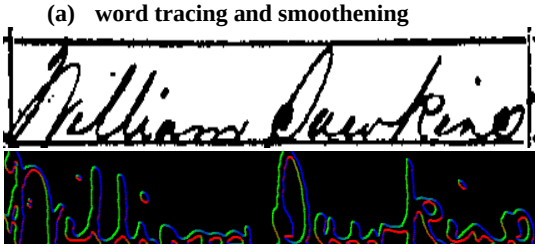
Fig 2: Examples of Templates

d. Neural Network Training

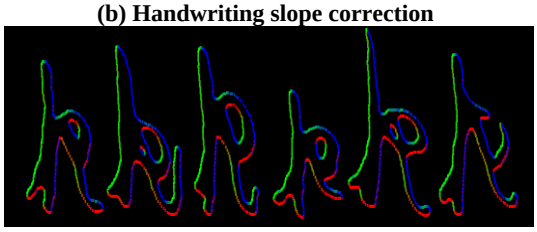
Creating Vectors data for the Neural Network creates training vector and testing vector for the neural network. This can be to match the input accepted by the neural network function. The training of the network and evaluation of the performance is such a way that the net with different characters are trained. It includes numbers and characters, input is a matrix values to define network target as numbers.



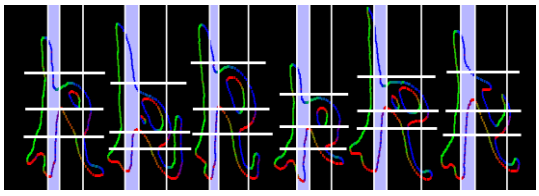
(a) word tracing and smoothening



(b) Handwriting slope correction



(c)Segmentation



(d) Spelling from possible letter readings

Fig 3: Stages of handwritten image analysis

The encoding is made with simple code table, during which each index gets a replacement value. The character recognition application used in different ways to use already pre-prepared Graphical User Interface and works like the workflow of recognition process. For test purposes, the new set of characters that are not ordered is chosen. In some cases, recognition will not be correct. False recognition is in fact logical, because the neural network is not properly trained for those characters. It is trained only for some characters. The problem lies in pre-processing and data retrieval. Datasets from other source were quite large, they contained a lot of data but too little information on attributes and image extractions is available and not able to test it on own example. Image preprocessing to get the training data for the neural network is based on input training.bmp image which has less than 50 characters. The training data is for the whole alphabet which is a unique feature of this method. Few characters were proved false.

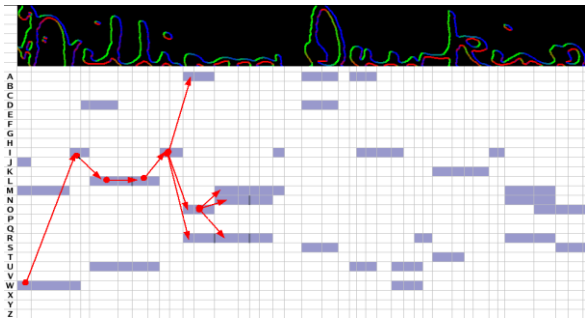


Fig 4: Template comparison

4. CONCLUSIONS AND RESULTS

Training and testing the neural network is a matter of MATLAB commands and use Multilayer Perceptron with two hidden layers of 35 and 10 neurons as a neural network. User can click on the buttons and test the character recognition approach is user friendly. GUI uses already pre-trained neural network from precedent computation steps. When the document is straight, gives more accurate results compared to severe curved fonts. Most of the characters are recognized as positive match. Now, the experiment deals only with English alphabets and numbers. This can be extended to any font, sizes, languages, symbols. There is a room for the improvement in character cropping and incorporating use of dictionary words to correlate instead of templates to attain better results.

5. ACKNOWLEDGMENTS

We are very grateful to the reviewers for their valuable comments. We also extend our gratitude to Mr. Arul Siju of Loyola College, Chennai for giving the valuable suggestions during this works.

6. REFERENCES

[1] Alejandro h. toselli, Enrique vidal, “Handwritten text recognition for ancient documents”

[2] Anoop m. Namboodiri, Michigan state university, 2004,”On-line handwritten document under standing

[3] Georgios Vamvakas, department of informatics and telecommunications, national and kapodistrian university of Athens “Processing and recognition of handwritten documents”

[4] Vassilis papavassilioua, themos stafylakisa,b, vassiliskatsourosa, george carayannisa,”Handwritten document image segmentation into textlines and words”.

[5] Han shu, Massachusetts institute of technology, 1996,”On-line handwriting recognition using hidden markov models”