Reason - A Technical Magazine
Vol. – IX, 2009 & 2010

A TECHNIQUE FOR DETECTION AND RECOGNITION OF OPTICAL CHARACTER IN DIGITAL IMAGES

Surajit Biswas* and Satyendra Nath Mandal**
* I .T. , Final Year Student, Kalyani Govt. Engg. College, Kalyani, Nadia, West Bengal, India
email: surajit_1176@yahoo.co.in
** Assistant Professor, Dept. of I.T, Kalyani Govt. Engg College, Kalyani, Nadia, West Bengal, India
email: satyen_kgec @rediffmail.com

Abstract : Human eye have the ability to detect and recognize a character of some language, which they know, by the power of their brain as they are accustomed to that shape of the character font. But this thing is typically challenging for computers as it has no ability to read and learn automatically. All it needs to work is to provide it an artificial technique by human beings to make its artificial brain to read and understand from a character image, an optical character image. There are lots of algorithms available for the optical character recognition. Although no such technique yet exists, which can guarantee that its steps are 100%? In this paper, a new technique has been proposed to detect optical characters from digital images. Here the basic algorithm for character contour tracing algorithm using Fourier descriptors is used and then the proposed algorithm is used to recognize the optical character so that it can help the basic recognition technique more and more accurate.

Keywords : Digital image, character recognition, Fourier descriptors, contour tracing, shape representation, shape similarity measure, feature extraction.

1. Introduction

Most people learn to read and write during their first few years of education. By the time they have grown out of childhood, they have already acquired very good reading and writing skills, including the ability to read most texts, whether they are printed in different fonts and styles, or handwritten neatly or sloppily. Most people have no problem in reading the light prints or heavy prints; upside down prints; advertisements in fancy font styles; characters with flowery ornaments and missing parts; and even characters with funny decorations, stray marks, broken, or fragmented parts; misspelled words; and artistic and figurative designs. At times, the characters and words may appear rather distorted and yet, by experience and by context, most people can still figure them out. On the contrary, despite more than five decades of intensive research, the reading skill of the computer is still way behind that of human beings. Here it is tried to develop the software system that can do this job of Character Recognition. While drawing something on a white paper or on a blank datasheet and that image is scanned by scanner peripheral of a computer, the machine have to identify then whether any character of any particular language is there on the scanned image or not. A man can easily tell by looking into the image, if there are any character image on the scanned document or not, which a machine have also to identify by its capability. Sometimes, when human beings also gets confused , machine can do it accurately, although that type of situation is rare, for example, some garbled images or some unclear image or some very light printed image or some time any unzoomable image. That is why the importance of character recognition system came into front.

2. Some Previous Works

Several works on this project– Character Recognition have been already made yet. Most important application of those are Pattern Recognition. In this field, the name of Dr. Ching Y. Suen must be mentioned [1]. He worked on basic handwriting recognition based on the Pattern Recognition algorithm and applying some extra constraints like as– Thinning, Skew ness detection and slant correction and like as other corrections in and input image he made to facilitate the recognition technique. Another great work, done [2] by Qing Chen in the year of 2003, when he was working on his thesis to be submitted to the institute in partial fulfillment of the requirements for the degree of Master of Applied Science. He gave ideas on three basic features a) Shape representation b) Shape Indexing

and c) Shape similarity matching. The trickiest part is the Shape Representation , which he made by the help of Shape contour tracing using CHAIN CODE algorithm and the Fourier Descriptors and second part is Region Based Image Invariant, using HU's seven moment invariants. Another idea was given by Ganapathy and Liew [3] based on Shape boundary Box tracing and using Artificial Neural Network (ANN). In this field it should also be mentioned about Pattern Recognition and Feed-forward Networks by Christopher M. Bishop [4], a Microsoft Researcher. Here A feed-forward network can be viewed as a graphical representation of parametric function which takes a set of input values and maps them to a corresponding set of output values. Also another idea, submitted [5] by Araokar, indicated matrixifying an input image and feed this all to a Neural Network. The other input images are all together form an weighted matrix, where most upper values indicates the black pixels and lesser values or negative values indicate white pixels . This system is capable of also finding out the correctness or accuracy by easy methods of calculating the weighted values of input image to library image and total count or sum of positive or upper values and taking their ratio.

Using Neural Networks, an adaptive character recognition idea was proposed by Faaborg [6]. Here the basic importance feature is that it is capable of comparing Handwriting Style vs. Recognition Accuracy. Using simple Neural Network Architecture, matrixifying the input character image and by java application this procedure is done. In their project report [7] by Tarachandani and Nath of IIT Kanpur in 2000, they mentioned the idea of slant estimation, which is also facilitates the Character Recognition technique.

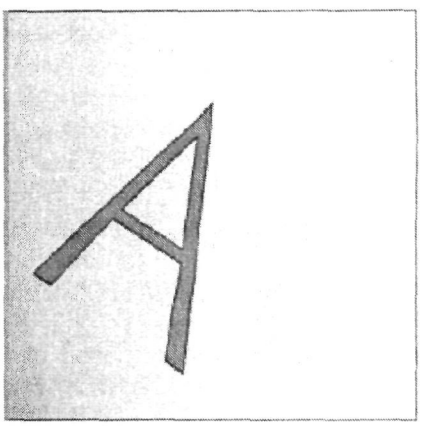
3. Procedure

The discussion on basic procedure of Character Recognition Technique is made below. An input image may come from scanning a document by a scanner and store it to the computer's memory. Or it may be any hand drawn character image, using some input machine like as mouse or a digital pen etc., and saved in the somewhere in computer's memory. Now the job is to recognize whether there is/are any character image of some language's alphabet or not by matching it to the stored library of character images. This is done by applying some technique, called character recognition technique. Here, first it will mention the basic technique before image feature extraction, applied here, step by step.

STEPS:

STEP 1
Scan and store the character image. Or store the hand drawn character image using some input machine like as mouse or a digital pen and next convert it to RGB to GRAY scale.

For example, consider about the following image:



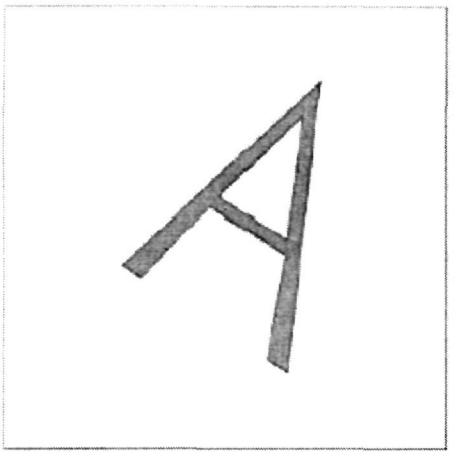
(Sample image : " IMG A")

STEP 2
Let one now eliminate the complex background by using the Local Gray Level Thresholding method. After eliminating the complex background, the example is shown below for the previous sample image.

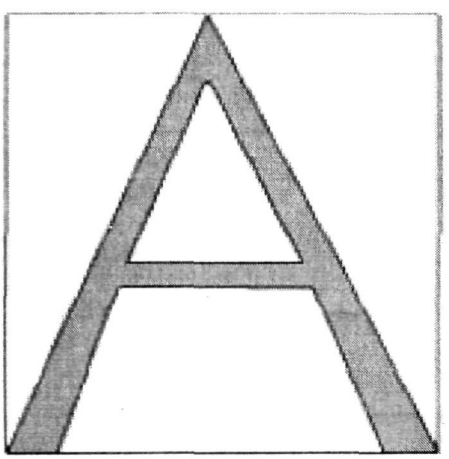
(After Complex Background Elimination of the Sample image "IMGA")

STEP 3
Next it is to convert the image from GRAY scale to BINARY scale.

STEP 4
The next step is to centralize the expected portion of the character image, where character can be found. This is also shown by the following example for the image "IMG A".



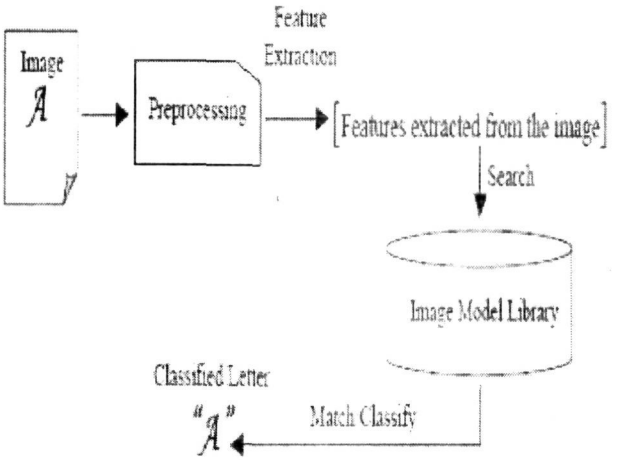
STEP 5
Now it is to trace the boundary of the character in the character image and set the area of the actual position. where character is in the image. The following figure explains this.



(After determining the bounding box)

3. Feature Extraction

The image is now ready for the preprocessing. The entire procedure is shown below by a diagram, that how this technique will work.



The preprocessing techniques have already been discussed in the previously mentioned steps. Thus next stage comes here is to image FEATURE EXTRACTION or to Extract the properties of the character image input. The proposed algorithm is used here.

5. Proposed Algorithm

This algorithm is actually based on the image information extraction and depending on that information a character is recognized if it is in the image , to be processed.

Specification of proposed algorithm :

Make The IMAGE MODEL LIBRARY

1) Determine the bounding boxes of all standard character images.
2) Set the all input standard character image sizes to 200x200 pixels.
3) Digitize the character image (I_n). Define black pixel by I(i,j)=1 , white pixel by I(i,j)=0.
4) For each (I_n) , define matrix (M_n) such that

   If I (i, j) =1 ,Then M(i, j) =1;
   Else:
   If I (i, j) = 0 , Then M(i, j) = -1;

5) Define weight matrix to every library image which is identical to its M_n matrix and affix this matrix to every corresponding Image IDs.

STEP to recognize the character image

1) After determining the bounding box define standard image resolution size to 69x65 (a standard) pixels
2) Digitize the input image (I_n)

3) Define column by i, row by j.
4) Define black pixel by I(i,j)=1 , white pixel by I(i,j)=0.
5) For each (I_n) , define matrix (M_n) such that

```
If  I (i, j) =1Then M(i, j) =1
    Else:
 If   I (i, j) =?0 Then M(i, j) =-1

 {
     for all j=1 to y
      {
        W_n (i,j)=W_n+M_n (i,j) ;
       }
  }
```

6) Construct (W_n) (for all images in the library to the input image) such that

```
for all i=1 to x
 {
   for all j=1 to y
    {
     W_n (i,j)=W_n+M_n (i,j) ;
    }
 }
```

7) Calculate candidate score :C ,where C=$\sum W_n(i,j) * I(i,j)$
8) Calculate Ideal Weight-Model Score (m) : where,

```
    for i=1 to x
{
     for j=1 to y
      {
           if  W_n(i,j) >0  then
          {
             m = m +W_n(i,j)
           }
             }
       }
```

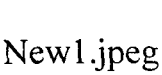
9) Calculate Recognition Quotient "Q" , where Q=C/m
10) For which Image ID the character image results a value of 0 5<Q<1, let's construct an array of those images.
11) Next, out of those IDs, the maximum is to be considered as the Character Image which is recognized by the image.

6. Results

Some samples are used to show the result of the performance of the proposed algorithm. These samples are nothing but some character images printed on optical-data-sheet. A few of the samples, used to work on this proposed algorithm, are given here.

SAMPLES

S



New1.jpeg

B

New2.jpeg

G

New3.jpeg

I

New4.jpeg

J

New5.jpeg

R

New6.jpeg

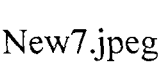
U

New7.jpeg

W

New8.jpeg

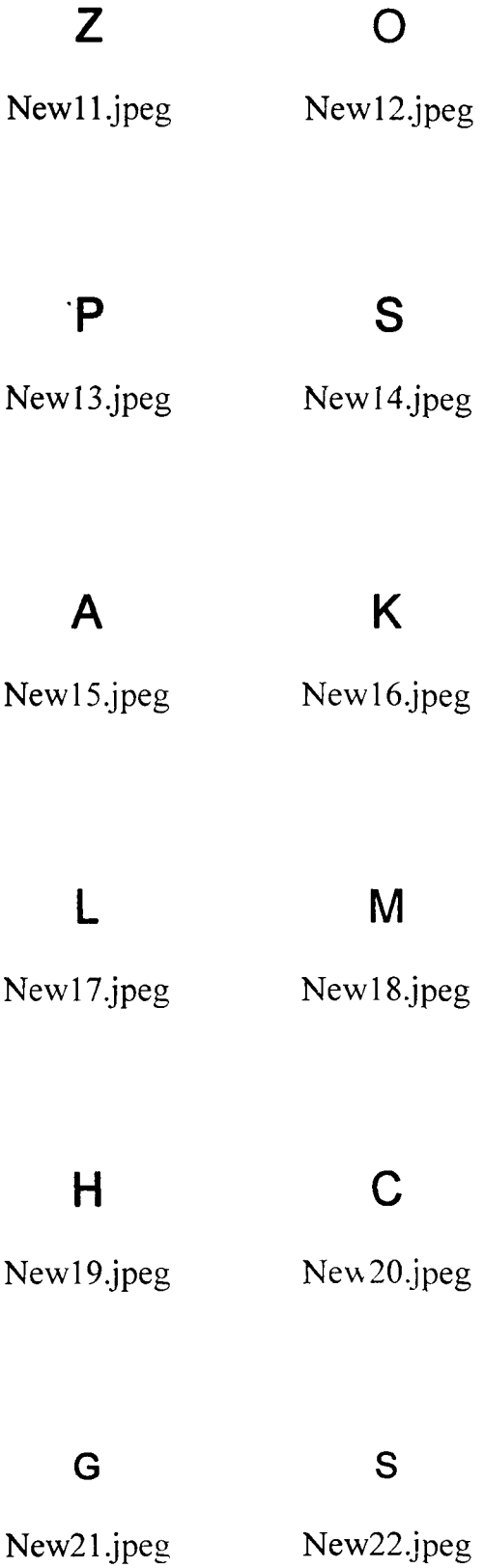
X



New9.jpeg

Y

New10.jpeg



OCR results

Sample No.	Sample Name	Accuracy	Original Character
1	New1	0 6570	S
2	New2	0 9092	B
3	New3	0 6027	C
4	New4	0 8356	I
5	New5	0 6877	J
6	New6	0 6811	R
7	New7	0 8558	U
8	New8	0 6463	W
9	New9	0 6784	X
10	New10	0 9723	Y
11	New11	0 9801	Z
12	New12	0 7708	O
13	New13	0 7572	P
14	New14	0 7408	S
15	New15	0 9925	A
16	New16	0 7782	K
17	New17	1	L
18	New18	0 8529	M
19	New19	1	H
20	New20	XXX	C
21	New21	0 5113	G
22	New22	0 6697	S
23	New23	0 7517	Q
24	New24	0 5473	Q
25	New25	0 7514	S
26	New26	0 5630	O
27	New27	0 6662	J
28	New28	0 7540	O
29	New29	0 5927	Q
30	New30	0 8860	S
31	New31	0 8932	U
32	New 32	0 7215	W
33	New33	0 6126	C
34	New34	0 6806	J
35	New35	0 5697	G
36	New36	0 7552	S
37	New37	0 6617	C
38	New38	0 6403	G
39	New39	0 7010	R
40	New40	0 7471	J
41	New41	0 8062	S
42	New42	0 7918	R

7. Performance

In the first phase, 42 samples are used and out of these, correct result is obtained for 41 samples (as shown in OCR result)

On the second phase, 40 samples are used and out of these, correct result is obtained for 38 samples.

Hence, it can easily be concluded that the performance is =(41+38)/(42+40) * 100 = 96.34146 %

8. Future Scope

To mention the future scope, first the drawbacks of this algorithm are to be mentioned here.

i) The algorithm works under the base of Input Inbuilt Image model library which is defined first at the time of first trial. So, it needs a lots of data-collection of different shapes and fonts of character images to built the image model library.

ii) The gathered result is typically obtained by gathering some samples within a limited boundary of our known. So the result may vary from the original 96.34% to some other extend while working with some other samples. So, it needs a lots of data collection approach to implement it to a software standard.

So, this data-collection approach is actually required to implement the BEST-OCR.

Finally it can be stated that what actually needed are :

i) Good inbuilt library of character images.

ii) Lots of number of collected data of character images of each single letter.

The ultimate goal is :

i) The software can recognize any optical character without any error.

ii) Run under a supportable platform which is independent of MATLAB

References

[1] Cheriet, M., Kharma, N., Liu, C.L., Suen, C.Y., Character Recognition Systems, Wiley, ISBN : 9870471415701

[2] Chen, Q., 2003, Evaluation of OCR Algorithms for Images with Different Spatial Resolutions and Noises, Ottawa, Canada, Date of access : 24/07/2010.

[3] Faaborg, A.J., 2002, Using Neural Networks to Create an Adaptive Character Recognition System, Cornell University, Ithaca NY.

[4] Ganapathy, V. and Liew, K.L., Handwritten Character Recognition Using Multiscale Neural Network Training Technique, Date of Access : 22/07/2010 www.waset.org/journals/waset/v39/v39-6.pdf

[5] Araokar, S., Visual Character Recognition using Artificial Neural Networks, Date of Access : 17/07/2010, www.cse.iitb.ac.in/~bikashbag/MTP/ocr1/vcrannsa.pdf

[6] Malik, S., 2000, Hand-Printed Character Recognizer using Neural Networks 95.407A Project, (219762).

[7] Mamedov, F., Hasna, J.F.A., Near East University, North Cyprus, Turkey via Mersin-10, KKTC, Character Recognition Using Neural Networks.

[8] Matan, O., Kiang, R.K., Stenard, C.E., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubberd, W., Jackel, L.D. and Chun, Y.Le., Handwriting Recognition Using Neural Network Architecture, AT&T Bell Laboratories, Holmdel, N.J. 07733, yann.lecun.com/exdb/publis/pdf/matan-90.pdf

[9] Bishop, Ch.M., 1996, Neural Networks for Pattern Recognition, Clarendon Press, Oxford.

[10] Hogan, M. and Shipman, J.W., OCR (Optical Character Recognition): Converting paper documents to text, infohost.nmt.edu/tcc/help/pubs/ocr/ocr.pdf

[11] Hansen, J., A Matlab Project in Optical Character Recognition (OCR), Date of Access: 22/07/2010, www.ele.uri.edu/~hansenj/projects/ele585/OCR/OCR.pdf

I have never let my schooling interfere with my education.
– Mark Twain