RESEARCH PAPER | Engineering | Volume : 4 | Issue : 4 | Apr 2014 | ISSN - 2249-555X

Pre-processing and Self training techniques in Handwritten Character Recognition

KEYWORDS: Handwritten character recognition, noise reduction, pre-processing techniques in character recognition, pattern matching, strokes, fixed-language, training neural networks, Gabor filter

Magesh Kasthuri	Dr.V.Shanthi
Research Scholar, SCSVMV University, Tamilnadu	Professor, Dept. of MCA., St. Joseph's College of Engineering, Chennai-119.

ABSTRACT *Handwriting recognition is a widely implemented technology to electronically identify handwritten text. In online handwriting recognition, text written on a touch surface of an electronic touch device is dynamically recognized based on the movements of a writing device (digital pen, a finger, or a stylus) and presented to the user after each continuous stroke such as a character or word is entered.*

Therefore, the user is able to edit a character or a word in the event of incorrect recognition of text by the electronic device as soon as it is presented to the user. On the other hand, in offline handwriting recognition, recognition of characters occurs by identifying characters from an image of handwritten text instead of the user writing on a touch surface. Here, the recognized text is presented to a user only when the entire handwritten text represented in the image is recognized.

An Artificial Intelligence based Neural system is successful when it is trained properly. The basis of training a neural network is to prepare the system to recognize various combinations of possibilities of characters. From a larger picture this is the key requirement for training a neural network. But there are underlying principles in training a neural network.

In other words, training is not a simple record and process technique but also a technique which decides the working logic of the processing algorithm.

This paper discusses some of the key features of training a neural network along with some key training methodologies for Handwritten character recognition system.

I. Introduction
The primary challenge associated with offline handwriting recognition is that the handwritten text cannot be edited or re-entered by the user until the entire handwritten text is recognized.

A user, thus, may not be able to provide a feedback for correction after the recognition of each character or word in case of an incorrect recognition of that character or word. Thus, the errors in the offline handwriting recognition of that handwritten text may keep on accumulating in the absence of user supervision. Therefore, it is desirable to increase the accuracy of recognition of handwritten text in systems that implement offline handwriting recognition.

It is very important to understand these topics from Soft computing base as they are not only complex in nature but there is no concrete solution or algorithm in solving various problems around the topic of Handwritten character recognition.

Handwriting recognition is a widely implemented technology to electronically identify handwritten text. In online handwriting recognition, text written on a touch surface of an electronic touch device is dynamically recognized based on the movements of a writing device (digital pen, a finger, or a stylus) and presented to the user after each continuous stroke such as a character or word is entered.

A typical offline character recognition system is based on training and built-in learning system. In the simplest case such a system works on documents with standard fonts. Other such systems that can interpret non-standard fonts and hand-written characters need extensive training to build a reference knowledge-base. However, such systems operate based on selection of languages and fail to address mixed-language content.

Therefore, the user is able to edit a character or a word in the event of incorrect recognition of text by the electronic device as soon as it is presented to the user. On the other hand, in offline handwriting recognition, recognition of characters occurs by identifying characters from an image of handwritten text instead of the user writing on a touch surface. Here, the recognized text is presented to a user only when the entire handwritten text represented in the image is recognized.

II. Problem description
One challenge associated with offline handwriting recognition is that the handwritten text cannot be edited or re-entered by the user until the entire handwritten text is recognized. A user, thus, may not be able to provide a feedback for correction after the recognition of each character or word in case of an incorrect recognition of that character or word. Thus, the errors in the offline handwriting recognition of that handwritten text may keep on accumulating in the absence of user supervision. Therefore, it is desirable to increase the accuracy of recognition of handwritten text in systems that implement offline handwriting recognition.

Another challenge is that various users write in varied handwriting styles and in different languages. Each character in the handwritten text may have been written in multiple handwriting styles by different users. It is desirable that a character be correctly recognized in spite of having been written in varied handwriting styles. In addition, a character in one language may be similar, but not same, to a character of a different language and is, thus, prone to be incorrectly recognized. Therefore, there is need for greater accuracy in offline handwriting recognition of such handwritten text.

III. Study of Literature
In one embodiment[1] called Gabor filter based handwritten character recognition system, a character recognition method executed on an electronic device is disclosed, the method comprising: receiving, at the electronic device, an image representing a character including one or more central strokes; determining a set of parameters associated with each of the one or more relative (associative) strokes; comparing, for each of the one or more relative strokes, the associated set of parameters with a plurality of stored sets of adjacent

Volume : 4 | Issue : 4 | Apr 2014 | ISSN - 2249-555X

parameters, wherein each of the plurality of stored adjacent strokes is associated with a stored set of relative parameters; identifying next stroke, from among the plurality of stored strokes, corresponding to each of the one or more strokes based on the comparison to identify the possible character comprising these strokes in order

In another embodiment (US Patent WO 2003023696 A1), a non-transitory computer-readable medium is disclosed, the non-transitory computer-readable medium storing instructions that, when executed by one or more processors, causes the processor to perform operations comprising: receiving, at an electronic device, an image representing a character including one or more first strokes; determining a set of first parameters associated with each of the one or more first strokes; comparing, for each of the one or more first strokes, the associated set of first parameters with a plurality of stored sets of second parameters, wherein each of the plurality of stored second strokes is associated with a stored set of second parameters; identifying a second stroke, from among the plurality of stored second strokes, corresponding to each of the one or more first strokes based on the comparison; and identifying the character based on the identified one or more second strokes.

IV. Steps involved in training

An offline interpretation of hand-written, mixed language, multi-lingual document must be agnostic to the style of hand-writing of different writers of the document and also agnostic to the languages used. The overall steps involved are:

1. Receive an image representing one or more characters having multiple strokes or segments forming the character
2. Pre-process the image (noise reduction, transformation, make-up and optimize for processing)
3. Segment the image into one or more strokes
4. Determine a set of first parameters associated with each of the one or more first strokes.
5. For each pair of strokes, compare associated stroke elements (adjacent cell comparison)
6. Identify next stroke and process the step of identifying the strokes in pairs or combination
7. Identify the character based on identified group of strokes

From a hard copy document, image will be extracted for offline character recognition. There are quite a few conventions in determining the input mode of such offline character recognition viz:

i) complete document read / scan
ii) sequential reading (word / sentence wise) from the document (scan)
iii) Reading character / word / sentence directly from the digital image of the document

This required offline character recognition from a handwritten, multi-lingual document with mixed-language content.

Following steps briefly discusses step by step process involved in a typical training system based on the study of above literature study and solving the problems associated with different solutions

Steps involved:
Step 1: Obtain a stroke
Step 2: Normalize stroke
Step 3: Generate Index
Step 4: Obtain a stroke
Step 5: Create index structure
Step 6: Index retrieval
Step 7: Grouping characters
Step 8: Store the character set with index

Step 1:
The Indexing Module (IM) obtains a feature / stroke from the Makeup Module (AM).

Step 2:
The IM then transforms the obtained stroke using the method of spatial translation, resize normalization and thickness normalization.

Step 3:
The IM then analyze the normalized stroke and generates the corresponding metadata structure for the stroke in the index structure format.

a. The IM then checks for existence of the stroke in the index-store using the method of retrieval of a stroke. In case the entry is already found then the IM discards the obtained stroke.
b. Otherwise, the IM creates a new index-entry by using the hash-indexing method for the obtained-normalized stroke.
c. And generated index-entry is stored in the index-store.

Step 4:
The Indexing Module (IM) obtains a feature / stroke from the Makeup Module (AM).

Step 5:
For the group of strokes which we want to retrieve from index-store, MM module creates index structure so that index comparison from hash table of indexes would be possible by IM module

Step 6:
Verify the element set of the matching stroke if the strokes are matching. If found what is returned is: exact match index with weightage (possibility of match). If not found then it will return a highest possible match with corresponding weightage.

Step 7:
Create character sets from the above list of feature groups of character-id for each language

Step 8:
Store the character set as an associative array with language-id as key and character set as elements.

Set	Stroke ID	Stroke
A1	1	
A2	2	
A3	3	
B1	4	
B2	5	
B3	6	

Mapping table 1: Mapping between sets of parameters and strokes and the character to identify (right-side)

In an exemplary scenario, a plurality of sets of second parameters, each set associated with a stored second stroke, may be stored in the memory of the electronic device. Various combinations of these sets of second parameters may be classified by different handwriting styles and stored with their associated characters.

Volume : 4 | Issue : 4 | Apr 2014 | ISSN - 2249-555X

Here, handwriting style 2 may represent A in a different manner than handwriting style 1 because of different handwriting styles of two users. Therefore, the processor, while storing these second parameters, may interpret the character to be including three strokes according to style 1 and two strokes according to style 2.

V. Experimental setup

Consider a simple example of learning a character 'A' based on mapping tables given above. The steps involved in this learning process are explained below:

Input

From an input source (scanner), the image representing the text may be received by the electronic device by means of scanning a handwritten document.

For example, a document may represent a handwritten text in one or more languages. The document may be scanned by a scanner that is communicatively coupled externally to the electronic device or built into the electronic device. On scanning the document, the scanner may capture an image of the document which is further provided to the electronic device.

Pre-process

A processor of the electronic device may preprocess the received image. The preprocessing may include digitization of the received image.

Each pixel of the received image is associated with a pixel value between 0-255. The digitization may include converting the pixel value of each pixel to either a pixel value 0 or a pixel value 1. The pixel value 0 may be associated with a white pixel and the pixel value 1 may be associated with a black pixel.

The preprocessing may further include removing noise such as, but not limited to, salt and pepper noise and Gaussian noise from the digitized image using one or more noise removal techniques known in the art. In addition, the preprocessing may also include making the width of all the characters in the received image uniform by normalizing the width of each portion of the text to a predetermined value of width.

Normalization

This may include either reducing width of some portions of the handwritten text or increasing their width to the predetermined value of width. The width of a portion may be reduced by converting any undesired black pixels to white if the handwritten text is represented by black pixels.

Similarly, the width may be increased by converting some white pixels in the vicinity of the handwritten text to black if the text is represented by black pixels. The normalizing of the width of the text may result in a uniform width across each portion of a character and across all the characters in the received image.

It should be understood, however, that preprocessing of the received image may include additional techniques such as, but not limited to, cropping, resizing, applying one or more filters and transformation of an image from colored to a grayscale image.

Once the received image is preprocessed, the received image is segmented into one or more first strokes by the processor.

Segmentation

On preprocessing the image, the processor of the electronic device may segment the handwritten text in the received image into characters. The processor may distinguish one component of the text from another component based on spacing between the components.

A component may be a set of continuous pixels that includes one or more characters. A first component including a character may be distinguished from a second component including one or more characters based on whether the spacing between the first and the second components is above a predetermined first threshold. Thus, the processor may distinguish each character from other characters. Once the characters are distinguished from each other, the processor may segment the text into individual characters.

Strokes preparation

Language	Character	Lang ID	Char ID	Strokes per style				
English	A	1	1	A1	A2	A3		
	B	1	2	B1	B2			
	C	1	3	C1	C2			
	D	1	4	D1	D2			
	E	1	5	E1	E2	E3	E4	
	F	1	6	F1	F2	F3		
	G	1	7	G1	G2	G3	G4	G5

Character set Index

Stroke set Index

On segmenting the characters, the processor may represent each character in a matrix format that includes a plurality of cells. Below represents an exemplary matrix representation of an English alphabet 'A' as a character 202 on a matrix 204 that includes a plurality of cells 206.

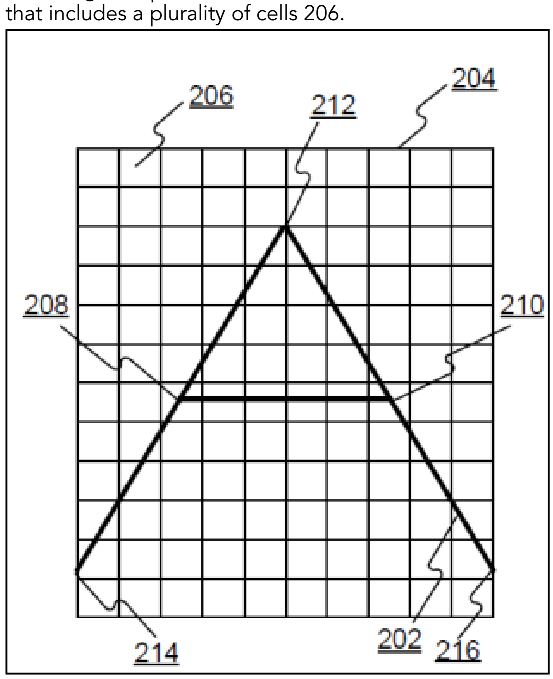

Fig 1: Stroke representation

Further, the processor may scan matrix 204 to identify one or points on character 202 that represent a sudden change in angle.

A sudden change in angle may be considered at a point on a character when two linear or non-linear line segments form an angle at that point that is below a predetermined threshold angle.

Here, point 208, point 210, and point 212 on character 202 may represent a sudden change in angle along the representation of character 202 because at each of these points, two line segments form an angle that may be considered to be below a predetermined threshold angle.

Volume : 4 | Issue : 4 | Apr 2014 | ISSN - 2249-555X

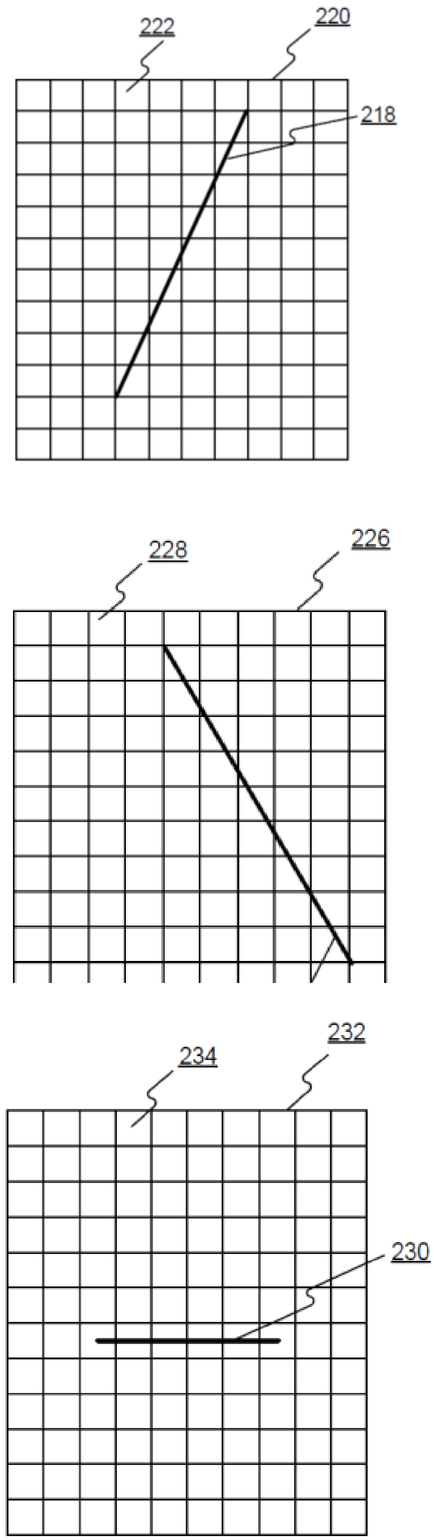

Fig 2: Segmentation and first stroke identification

For example, if an angle formed by two line segments at a point is below a predetermined threshold angle of 40°, it may be considered as a sudden change in angle. Once all the points representing a sudden change in angle have been identified, the processor may split character 202 at these points into different strokes as represented in below figures respectively.

Stroke recognition
Below figure illustrates a stroke 302 that may be represented on matrix 304 that includes plurality of cells such as cell 306, cell 308, cell 310, cell 312, cell 314, cell 316, cell 318, cell 320, cell 322, cell 324, cell 326, cell 328, cell 330 etc.

Stroke 302 may extend from cell 306 to cell 326 as shown in above figure, thereby, spanning multiple cells, each of which may represent a portion of stroke 302.

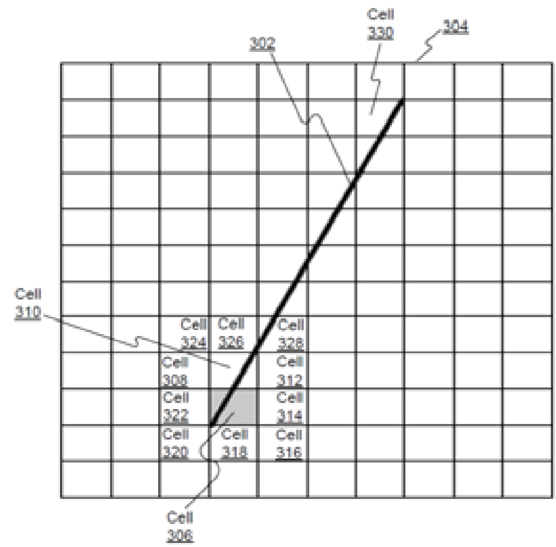

Fig 3: Cell representation for a given stroke

The processor may scan each of these cells that represent a portion of stroke 302 sequentially to determine one or more parameters associated with a portion of stroke 302 represented in that cell.

Before initiating scanning of the cells representing different portions of stroke 302 to determine one or more parameters, the processor may search for a first cell 306 to be scanned that represents a portion of stroke 302. In accordance with these embodiments, the processor may search for cell 306 by scanning each row of matrix 304 horizontally starting from a lower left corner of matrix 304.

Once the processor has found cell 306 as a result of its search, the processor may determine one or more parameters associated with a portion of stroke 302 represented in cell 306. These parameters may include length and width of the portion of stroke 302 represented in cell 306.

Self-training
For example, the angle of a curvilinear portion may range from 10° to 65° with respect to a horizontal axis. The parameters may also include coordinates of multiple points of the portion. The coordinates may be computed to determine a positioning of various points of the portion with respect to horizontal and vertical axis.

Once the processor determines the parameters associated with cell 306, the processor may scan the neighboring cells of cell 306 to search for a subsequent cell that represents a portion of stroke 302. In some embodiments, the neighboring cells of cell 306 may include a first layer of eight cells that are adjoining to cell 306 and enclose cell 306.

Here, the processor may be configured to scan cell 308, cell 310, cell 312, cell 314, cell 316, cell 318, cell 320, and cell 322 that are adjacent to cell 306. As a result of this scan, the processor may determine that cell 310 represents a second portion of stroke 302.

Volume : 4 | Issue : 4 | Apr 2014 | ISSN - 2249-555X

The processor then determines one or more of the discussed parameters associated with the second portion of stroke 302 that is represented in cell 310.

Result of evaluation:
The processor may also determine a set of first parameters for other strokes such as those shown in figure 2 into which the image of character 'A' was segmented by the processor.

This is how self-learning based training is conceptualized based on first stroke identification.

VI. Acknowledgement
This concept is a sub-section of US/India patent filed concept and acknowledged by the patent (Indian Patent Journal Issue 47/2013 dated 22/11/2013). All content and idea are copyrighted.

VII. Summary
The discussion of this topic is based on a character recognition device comprising:

- at least one processor; and
- a memory storing instructions executable by the at least one processor, wherein the instructions configure the at least one processor to:
- receive an image representing a character including one or more strokes;
- determine a set of first parameters associated with each of the one or more first strokes;

A character recognition method executed on an electronic device, and the method of processing comprising:

- Receiving, at the electronic device, an image representing a character including one or more first strokes;
- Determining a set of first parameters associated with each of the one or more first strokes;

A second level of processing will be engaged which will judge whether to assimilate the results of the first process, extend them and proceed to the next stage with a positive recognition, or to dismiss them and reinvoke the first level again while asking for modifications.

REFERENCE
[1] Wai Kin Kong, David Zhang, Wenxin Li, Palmprint feature extraction using 2-D Gabor flters, The Journal of the Pattern Recognition Society (Elsevier) Pattern Recognition 36 (2003) 2339 - 2347 | [2] Daming Shi, Robert I. Damper And Steve R. Gunn, Offline Handwritten Chinese Character Recognition by Radical Decomposition, ACM Transactions on Asian Language Information Processing, Vol. 2, No. 1, March 2003, Pages 27-48. | [3] Anita Pal, Dayashankar Singh, Handwritten English Character Recognition Using Neural Network, International Journal of Computer Science & CommunicationVol. 1, No. 2, July-December 2010, pp. 141-144 | [4] R.Jagadeesh Kannan And R.Prabhakar, Off-Line Cursive Handwritten Tamil Character Recognition, WSEAS Transactions On Signal Processing, Issue 6, Volume 4, June 2008, Issn: 1790-5052 Pages: 351-360 | [5] Lubna Badri, Development of Neural Networks for Noise Reduction, The International Arab Journal of Information Technology, Vol. 7, No. 3, July 2010 Pages: 289-294 | [6] Mansi Shah And Gordhan B Jethava, A Literature Review On Hand Written Character Recognition, Indian Streams Research Journal, Vol -3 , ISSUE 2, March.2013, ISSN:-2230-7850 | [7] Zhiyi Zhang, Lianwen Jin, Kai Ding, Xue Gao, Character-SIFT: a novel feature for offline handwritten Chinese character recognition, 10th International Conference on Document Analysis and Recognition, 2009 Pages: 763-767 | [8] Li Fuliang, Gao Shuangxi, Character Recognition System Based on Back-propagation Neural Network, 2010 International Conference on Machine Vision and Human-machine Interface Pages: 393-396 | [9] Dr.J.Venkatesh and C. Sureshkumar, Tamil Handwritten Character Recognition Using Kohonon's Self Organizing Map, IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.12, December 2009 Pages: 156-161 | [10] L. D. Jackel, C. E. Stenard, H. S. Baird, B. Boser, J. Bromley, C. J. C. Burges, J. S. Denker, H. P. Graf, D. Henderson, R. E. Howard,W. Hubbard, Y. leCun, 0. Matan, E. Pednault, W. Satteriield,E. Sickinger, and T. Thompson, A Neural Network Approach to Handprint Character Recognition, IEEE CH2961-1/91/0000/0472 2001 Pages: 472-475 | [11] Seong-Whan Lee, Young- Jaon Kim, A New Type of Recurrent Neural Network for Handwritten Character Recognition, IEEE 0-8186-7128-9/95 2005 Pages: 38-41 | [12] Ishwarya .M.V, R. Jagadeesh Kannan, An Improved Online Tamil Character Recognition Using Neural Networks, 2010 International Conference on Advances in Computer Engineering IEEE 978-0-7695-4058 Pages: 284-288 | [13] G. Tambouratzis, Applying Logic Neural Networks to Hand-written Character Recognition Tasks, IEEE 0-8186-7686-8/9 10996 Pages : 268-271 | [14] Anil K.Jain, Jianchang Mao, K.M.Mohiuddin, Artificial Neural Networs : A Tutorial, IEEE 0018-9162/96 March 1996 Pages: 31-44 | [15] Neural Networks, Fuzzy Logic and Genetic Algorithms – Sythethis and Applications by S.Rajasekaran and G.A.Vijayalakshmi Pai from Eastern Economy Edition Page-31-33 | [16] LI Guo-hong,SHI Peng-fei, An approach to offline handwritten Chinese character recognition based on segment evaluation of adaptive duration, Journal of Zhejiang University Science ISSN 1009-3095 2004 5(11):Pages: 1392-1397 | [17] Magesh Kasthuri, Dr. V.Shanthi, Noise Reduction and Pre-processing techniques in Handwritten Character Recognition using Neural Networks, TECHNIA International Journal of Computing Science and Communication Technologies, VOL.6 NO. 2, January. 2014 (ISSN 0974-3375) Pages: 940-947 |