World Applied Sciences Journal 13 (6): 1459-1464, 2011
ISSN 1818-4952

Broken and Touching Characters Recognition in Persian Text Documents

[1]*Abdol Hamid Pilevar and* [2]*Mohammad Taher Pilevar*

[1]Department of Computer Engineering, Language Engineering Lab.
Bu Ali Sina University, Hamedan, Iran
[2]Department of Electrical and Computer Eng,
Natural Language Processing Lab, University of Tehran, Tehran, Iran

Abstract: A technique named Eleven Direction Method (EDM) is used to detect the broken and touching characters in the printed text documents. In this technique the shape of the vertical projection curves are considered. The behavior of the edges of vertical projection curve is selected for creating the feature vectors of the characters. The edges of the vertical projection curve traced and the direction of the movement in the edges has been mapped by EDM method .The direction codes have been extracted and saved as features vectors of the characters. The method is tested on the Persian printed text documents. The testing data are collected from various legal documents. The test documents contain alphabet, special, broken and touching characters. The effectiveness and performance of the proposed algorithm have been tested with two separate sample data. Our method is compared with three other similar methods, the results shows a significant improvement. The experiments shows that more than 97% of the total segmented characters and over 93% of broken and touching characters are recognized correctly therefore, the proposed algorithm and the selected features perform satisfactorily.

Key words: Touching characters · Character recognition · Broken characters · Natural language processing · Text documents

INTRODUCTION

In Persian language the letters can be attached or separated freely, therefore touching characters problem in English text documents is not the same in Persian text documents. The similar problem can be raised when we are looking for broken and touching characters instead of touching characters. Segmentation of merged characters is one of the main causes for errors in the recognition. As segmentation errors induce recognition errors, the performance of segmentation is crucial for the whole OCR process. For segmenting the merged characters in a work which is called cut-off point or an inflexion, the point with the smallest interior angle is detected and the whole stroke is split into two adjacent curves by this point, [1]. A segment extraction algorithm based on polygonal approximation for On-Line Chinese characters recognition [2]. Directional and positional features for on-line one stroke cursive character recognition based on dynamic time warping algorithm [3]. Character recognition based on pixel distribution probability of character image [4]. A character is defined by a multivariate random variable over the components and its probability distribution is learned from a training data set [5]. A segmentation algorithm for character recognition on a license plate [6]. A character recognition method which executes segmentation and recognition simultaneously [7]. A method for segmentation of touching italic characters [8]. A segmentation technique for touching Thai type written [9]. A segmentation of machine printed Gurmukhi text [10]. A segmentation technique far recognizing the touching Thai type written [11]. A lossy/lossless compression method for printed typeset bi-level text images is proposed for archiving purposes [12]. A recognition method of line-touching characters without line removal [13]. Links between landscape aesthetic theory and visual indicators [14]. A technique for identification and segmentation of Bengali printed characters [15]. Vertical and horizontal text lines are detected without prior assumption. The touching characters belonging to different lines are detected [16]. A merged character segmentation and recognition method based on forepart

Corresponding Author: Abdol Hamid Pilevar, Department of Computer Engineering, Language Engineering Lab. Bu Ali Sina University Hamedan, Iran. E-mail: pilevar@basu.ac.ir.

prediction, necessity-sufficiency matching and character-adaptive masking [17]. A method for segmentation of touching italic offered [18]. A two pass algorithm for the segmentation and decomposition of Devanagari composite characters-symbols into their constituent symbols [19]. A recursive segmentation algorithm for segmenting touching characters [20]. A morphological based method for recognition of middle age Persian characters [21].

Furthermore, since segmentation errors often raise chain effects, the performance of segmentation is crucial for the whole OCR process. In this paper, we address this issue by proposing a robust method for detecting the touching characters in Persian text document images. We also present a technique for recognition of isolated Persian character images using simple features.

Preprocessing: In the present system, character images have been obtained by optical scanning of the character images on plain paper. The input data obtained by scanning of printed text is contaminated with noise and contains redundant information. Preprocessing includes noise removal, elimination of redundant information as far as possible, segmentation and scaling. The segmentation started by scanning the page images then continued by horizontally detecting the Text lines in each scanned page. Frequency of black pixels in each row is counted in order to construct the row histogram. The position between two consecutive lines, where the number of pixels in a row is zero denotes a boundary between the lines. After a line has been detected, it is scanned vertically. In order to find the column histogram, the number of black pixels in each column is counted. If there are *n* consecutive vertical scans that find no black pixel, we denote those columns to be a marker between two words. The value of *n* is decided experimentally. To segment the individual character in a word, the column histogram is found; number of black pixels in each column is calculated. If there are more than one consecutive scans that has no black pixels, then the region is decided to be a marker between characters and again if it is more than *n*, then considered as marker between two words. The isolated characters have been normalized so that size invariant recognition is possible. Though the recognition is size invariant, better result is obtained when the size of the characters is assumed to be within a specific range. The performance of our algorithm has been tested, by using the fixed size characters in creating our characters training sets. For the developed system, 6pt - 36pt character sizes have been selected. The characters are scaled to a standard size (40 x 40) using an efficient scaling algorithm [22]. Scanning commonly introduces noisy cavities in the character images. These distortions detrimentally affect the shape of the characters. Single pixel components of noise are removed from the character images, before feature extraction.

Feature Selection and Extraction: The features are extracted and vectors are selected in the follows stages:

Vertical Projection Count (VPC): The vertical projection of all the characters (letters ا,ب ... ی, numbers 9-0 and special characters *,?, >, @, …) were found and the black pixels for each projected column are count.

The variations of the Vpc values from one column to the next column (right side) are found and the direction of the variation is respectively registered as directional vectors.

Angle Calculation of Directional Vectors: The angle of directional vectors are calculated with,

$$\alpha_i = \text{atan}(y_{i+1} - y_i) * \frac{360}{2\pi}$$

where: i = 1 …n-1, π= 3.1416… n is the number of features in the feature vector

For Example: The graphical depict of a directional vector is shown in Figure 1.

Eleven Connected Chain (ECC)-Codes: In Freeman's coding method characters are usually encoded either with 8-connected or 4-connected chain codes. For the work described in this paper, 11-connected chain codes were used (Figure 2). The technique is named Eleven Direction Method and abbreviated as EDM coding system.

Discrete Directional Angle (DDA): In our recognition system, templates are presented by strings of normalized discrete directional angle values. The angle values are classified as demonstrated in Table 1.

The number of the classes is selected to be eleven classes for being compatible with ECC-codes

EDM-Code: If the calculated discrete directional angles (DDA) string for a given directional vector in stage (d) are: -88, -36, 14, 53 and 86 then its EDM-code by referring respectively to Table 1 is 1479B

The EDM-codes for all of the characters (letters ا,ب ... ی, numbers 9-0 and special characters) are calculated and saved in EDM-Table.

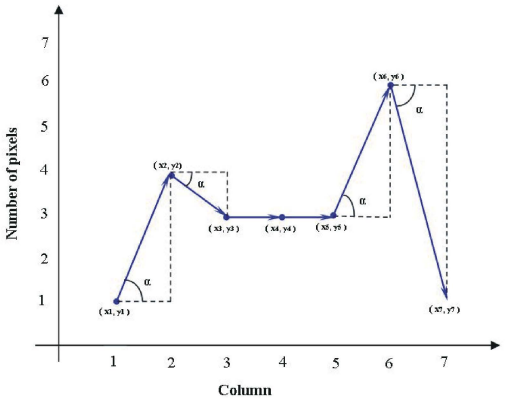

Fig. 1: The graphical representation of a directional vector

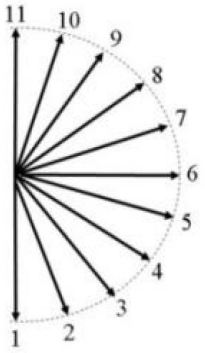

Fig. 2: Eleven connected chain coding

Table 1: normalized discrete directional angle values

	Angle	Angle
Direction	>	<=
1	-99	-81
2	-81	-63
3	-63	-45
4	-45	-27
5	-27	-9
6	-9	9
7	9	27
8	27	45
9	45	63
A	63	81
B	81	99

Table 2: Characters feature vectors table

	F1	F2	F3	F8	F9	F20	F21	F22
ـد	7	8	6	0	0	0	0	0
ـه	7	7	7	6	5	0	0	0
ـه	7	6	7	7	7	0	0	0
ـا	6	6	6	9	10	0	0	0
ت	5	6	5	7	6	0	0	0
پ	8	5	5	7	7	0	0	0
ب	7	5	5	6	4	0	0	0
ا	10	5	0	0	0	0	0	0

Feature Vector: The left most 22 characters of the EDM-codes are selected as characters feature vectors and saved in the feature vector table (Table 2).

Our experiments show that, with selecting length of 22 features for each vector, the best results can be achieved.

Broken and Touching Characters Recognition: Template-matching algorithms can recognize touching versions of character templates without major difficulty. The reason for this phenomenon is simple: The difference between a broken image and the ideal one is relatively small. For template-matching, if a gap is introduced in an image and it does not change the interpretation of the image, it increases the distance from the image to all templates so the recognition is not affected. Statistical classifiers trained to recognize the most common broken and touching characters and in this context a broken and touching character is better seen as the result of a feature extractor anomaly.

In contrast, most structural approaches to classification are confused by broken characters, since a broken piece may drastically change the representation of the character structure.

Our approach allows recognition of broken characters by a template matching technique: the representations of a character with and without a gap look alike modulo the graph matching. In other words, the representation does not decide if the gap corresponds to a missing part or to a real separation of strokes, but allows both possibilities to coexist until the matching with models decides which the best interpretation of the gap is.

This property for gap representation is just an extension of our paradigm that all singularities on the image should appear on the input graph, but only in the context of the models is the interpretation of the singularities given.

Thus, gaps should be represented as character parts that may be missing or not from the ideal image. Gaps are now positive features with ambiguous interpretations. This brings up two problems: first, how to identify all gaps, in other words, how to define a gap and, second, how to consider only possible useful ones, since we want to increase the number of edges in the input graph as little as possible.

Figure 3 illustrates the segmentation procedure of the character 'مد'. The optimal cutting points (CP) can be found after several forward and backward cutting iterations. The segmented patterns are best matched to the prototypes. The touching characters may contain

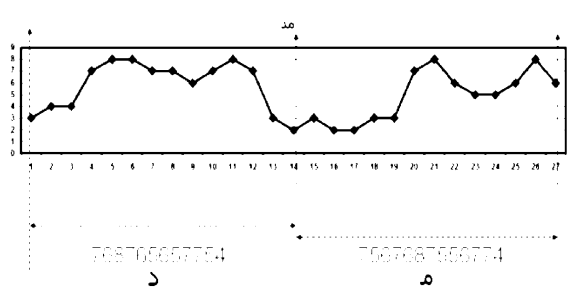

Figure 3: Graphic representation of Persian word "نم"

three or more characters while broken characters normally happen in the sequences of the recognized characters. The broken characters will be rejoined in the merge process using layout context information.

Dynamic Recursive Segmentation Algorithm: The approach presented in this section, the dynamic recursive segmentation algorithm, executes a forward segmentation or a backward merge process dynamically, based on the recognition result of the current input array and the neighboring broken character arrays, until the connected components are accepted by the classifier as a valid character. Instead of registering all the cutting points whose discrimination function values exceed the specific threshold, only the cutting points determined by the recursive segmentation algorithm are recorded for additional processing with context information and spelling tools. To prevent from misclassifying the broken and touching characters whose shape is similar to the specific characters, the minimum distance classifier utilizes multiple rejection thresholds to control the recursive segmentation process. R_1, R_2 and R_3 are respectively used for the initial input patterns (IP), the residue input pattern (RP) and the forward and backward input patterns (FP- BP). By properly adjusting R_1, R_2 and R_3 ($R_1 < R_2 < R_3$), the algorithm achieves the optimal results.

Matching the Feature Vectors: Given the feature vector for a word, our method finds vectors and sub-vectors that are homeomorphic to some prototype. A distance function between vectors measures the amount of distortion between vector, sub-vector and prototype. This distance function represents the minimum transformations that a vector will undergo so that the matching is possible. Thus, it is used as a measure of the quality of the matching. The cost of matching two vectors is the distance between the vectors.

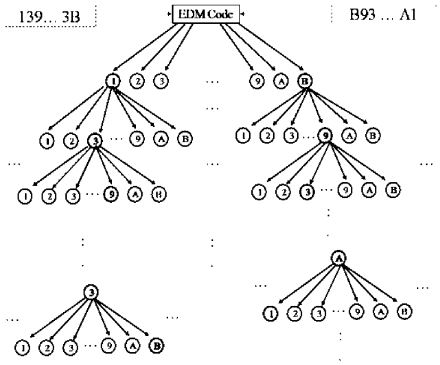

Fig. 4: The search tree: each state represents a character candidate and the remaining pattern. A path from the root to a node constitutes a segmentation attempt. Search pattern for feature vectors 139…3B and B93…A1 are marked

The cost function and the algorithm that is used to find sub-vectors are basically the same used for recognition of segmented words. In order to find sub-vectors of the entire word vector that match prototypes, the recognition process is initiated on all features of the vector. The prototypes will guide the best recognition that contains the initial feature. The cost of the sub-vector matching does not include features that are not matched in the candidate, since they may belong to a different character.

Search Algorithm for Feature Vector: A special tree search technique named EDM-Codes method is applied, the feature vectors are looked for using a tree based searching technique as displayed in Figure 4. The length of the feature vectors are 22 characters, but the EDM-coding system recognizing eleven codes, therefore the feature vectors are divided into two equal sections, which are named upper and lower half.

Let Us for Example: A query image is given and its feature vector is divided into two eleven character codes and extracted by the system, then the searching process will be followed through the nodes of the search tree as marked in Figure 4.

Distance Metric: The characters are classified based on the above mentioned features vectors and for similarity measurement the weight method is used [23]. The similarity degree S_i, between the i-th elements of the feature vectors of character images $f(x)$ and $g(x)$ is defined as:

if ($f_i = g_j$), then

$$S_{i=1}$$

else

$$S_i = \frac{\min\left(f_i, g_j\right)}{\max\left(f_i, g_j\right)}$$

Where:

f_i: is the feature vector of the i-th character in the mean set of the training sets

g_j: is the feature vector of the j-th character in the document.

The similarity degree between character images $f(x)$ and $g(x)$ is the sum of the similarity degrees between the corresponding n elements of the feature vectors derived from the two images and defined as:

$$S = \sum_{i=1}^{n} \delta_i S_i$$

Where:

$0 \le S_i \le 1$ n is the number of features, n=22

$$\sum_{i=1}^{n} \delta_i = 1$$

In simplest case, the value of weights δ_i can be set $\delta_t = \frac{1}{22}$

Experimental Results and Discussion: The effectiveness and performance of our algorithm have been tested on samples collected from various images of legal documents belonging to one city. For testing our method, around 300 printed Persian test documents are scanned at 300 dpi and binarized using the two-stage method described in [24]. About 900 broken and touching characters are inserted artificially and randomly in different rows of the texts in the documents.

The textual lines and words segments are determined from valley points in the horizontal and vertical projection profiles. A one-pixel margin is kept while detecting zone boundaries of the characters. However, it is assumed that all the characters of a text line are of the same font size. The extracted characters are normalized to the size of 40×40 pixels. After the character boxes are extracted, before starting the character recognition and recognizing the broken and touching characters process, the documents are checked for their skew [25]. In the ultimate experiment, two sets of separate documents given to the

Table 3: Recognition rates after using different segmentation methods

Segmentation method	Recognition rate
Vertical projection	78.2%
Contour analysis	83.2%
Combination of vertical projection and contour analysis	87.8%
EDM (proposed method)	93.1%

EDM software system, the documents are segmented into 130000 characters, more than 840 touching and broken characters are recognized, it means: more than 93% of broken and touching characters are recognized correctly.

The proposed EDM method is compared with three other segmentation methods to show our methods improvement on segmentation. Table 3 shows recognition results obtained by EDM method and three other segmentation approaches.

As we can see in Table 3, the contour based method (83.2%) works better than vertical projection method (78.2 %) and the hybrid method which combines the vertical projection and contour analysis is more efficient than those two methods (87.8%). But the EDM method works even much better than combination method 93.1%. Therefore, the proposed algorithm and the selected features perform satisfactorily.

REFERENCES

1. Lu, X., X. Liu, G. Xiao, E. Song, P. Li and Q. Luo, 2008. "A Segment Extraction Algorithm Based on Polygonal Approximation for On-Line Chinese Character Recognition," Japan-China Joint Workshop on Frontier of Computer Science and Technol., pp: 204-207.
2. Chen, J., X. Lu, Q. Luo, P. Li and X. Liu, 2008. "A Segment Extraction-Combination Algorithm Based on Polygonal Approximation and Finite State Machines for On-Line Chinese Character Recognition," The 9th International Conference for Young Computer Scientists, pp: 1789-1794.
3. Long, T., L.W. Jin, L.X. Zhen J.C. Huang, 2005. "One stroke cursive character recognition using combination of directional and positional features," Interface, pp: 449-452.
4. Wang, N., 2008. "Printed Chinese Character Recognition Based on Pixel Distribution Probability of Character Image," 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp: 1403-1407.

5. Kang, K. and J.H. Kim, 2004. "Utilization of Hierarchical, Stochastic Relationship Modeling for Hangul Character Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, 26: 1185-1196.
6. He, X., L. Zheng, Q. Wu, W. Jia, B. Samali and M. Palaniswami, "Segmentation of Characters on Car License Plates," 2008. Electronic Engineering, pp: 399-402.
7. Iwamura, M., K. Negishi, S. Omachi and H. ASO, 2005. "Isolated Character Recognition by Searching Feature Points,", Eight International Conference on Document Analysis and Recognition,
8. Li, Y., S. Naoi and M. Cheriet, 2004. "A Segmentation Method for Touching Italic Characters," Pattern Recognition, pp: 2-5.
9. Watcharabutsarakham, S., 2004. "Segmentation for touching thai typewrittens," Sci., pp: 199-202.
10. Davessar, N.M., S. Madan and H. Singh, 2003. "A Hybrid Approach to Character Segmentation of Gurmukhi Script Characters," Pattern Recognition, pp: 4-8.
11. Electronics, N., C.T. Center and K. Luang, 2004. "Using Projection and Loop for Segmentation of Touching Thai Typewritten," Analysis, pp: 504-508.
12. Grailu, H., M. Lotfizad and H. Sadoghi-Yazdi, 2009. "A lossy/lossless compression method for printed typeset bi-level text images based on improved pattern matching", International Journal on Document Analysis and Recognition, pp: 1-24.
13. Hotta, Y. and K. Fujimoto, 2008. "Line-touching character recognition based on dynamic reference feature synthesis", Proceedings of SPIE-The International Society for Optical Engineering. pp: 6815.
14. Ode, Ã., M. Tveit and G. Fry, 2008. "Capturing landscape visual character using indicators: Touching base with landscape aesthetic theory", Landscape Res., 33(1): 89-117.
15. Sattar, M.D.A., K. Mahmud, H. Arafat and A.F.M. Noor Uz Zaman, 2007. "Segmenting Bangla text for optical recognition, 10th International Conference on Computer and Information Technology", ICCIT,
16. Faure, C. and N. Vincent, "Simultaneous detection of vertical and horizontal text lines based on perceptual organization", Proceedings of SPIE- The International Society for Optical Engineering, pp: 7247.
17. Song., J., Z. Li, Michael R. Lyu and S. Cai, 2005. "Recognition of Merged Characters Based on Forepart Prediction, Necessity-Sufficiency Matching and Character-Adaptive Masking", IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics, 35: 1.
18. Li, Y.S., M. Cheriet and Ching Y. Suen, 2004. "A Segmentation Method for Touching Italic Characters", Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04), pp: 1051-4651.
19. Bansal, V. and R. Sinha, 2002. "Segmentation of touching and fused Devanagari characters", Pattern Recognition, 35: 875-893.
20. Liang, S., M. Shridhar and M. Ahmadi, 1994. "Segmentation of touching characters in printed document recognition", Pattern Recognition, 27(6): 825 840, 1994.
21. Center, I.T., 2004. "An Efficient Selected Feature Set for the Middle Age Persian Character Recognition," Pattern Recognition, pp: 2-6.
22. Suman Kumar Nath and Muhammad Mashroor Ali, 1997. "An Efficient Object Scaling Algorithm for raster device", Graphics and Image Processing, NCCIS,
23. Pilevar, A.H., 2005. "Retrieval of signal from Biomedical Databases some new approaches", Ph.D thesis, University of Mysore,
24. Dhanya, D., 2001. "Bilingual OCR for Tamil and Roman scripts. Master's thesis, Department of Electrical Engineering", Indian Institute of Sci.,
25. Pilevar, A.H. and A.G. Ramakrishnan, 2006. "Inversion detection in text document images", 9th Joint Conference on Information Science, Taiwan,