2009 10th International Conference on Document Analysis and Recognition

A Tool for Ground-Truthing Text Lines and Characters in Off-Line Handwritten Chinese Documents

Fei Yin，Qiu-Feng Wang, Cheng-Lin Liu
National Laboratory of Pattern Recognition (NLPR),
Institute of Automation, Chinese Academy of Sciences
95 Zhongguancun East Road, Beijing 100190, P.R. China
E-Mail: {fyin, wangqf, liucl}@nlpr.ia.ac.cn

Abstract

Annotating the regions, text lines and characters of document images is an important, but tedious and expensive task. A ground-truthing tool may largely alleviate the human burden in this process. This paper describes an automated recognition-based tool GTLC for finding the best alignment between the text transcript and the connected components of unconstrained handwritten document image. The alignment process is formulated as an optimization problem involving candidate character segmentation and recognition. We have validated the effectiveness of this tool and have used it for annotating a large number of handwritten Chinese documents.

1. Introduction

Ground-truthing document images, i.e., annotating the regions, text lines, words and characters, is important for document analysis research, particularly, for algorithm design and performance evaluation. Often, document images were ground-truthed by humans since automatic tools were not available for giving desired accuracy. Automatic ground-truthing was available only for restricted cases such as documents with characters written in boxes or with large inter-line and inter-character spaces.

In recent years, some ground-truthing tools have been developed for annotating document images with less restriction [1-8]. For printed document images synthesized by text editing, printing and scanning, the text description (transcription) can be matched with the scanned image to generate ground-truth data [1][2]. Handwritten document images can be similarly matched with the transcript, but the matching process is much more complicated due to the irregularity of document layout and written word/character shapes. Usually, a dynamic time warping (DTW) [3] or linear hidden Markov model (HMM) [4] optimizing a match score between a text line image and its transcript is used to align the sequence of image segments and the words/characters. Some works have utilized word/character recognizers in scoring the alignment [5-8]. The word recognizer in [5] uses HMMs while the one in [6][7] uses character prototype-based word models. The work in [8] provides character-level alignment between online handwritten text lines and their transcription, which is converted to writer-dependent handwritten form. In these works, the tolerance to shape variability of the word or character models largely affects the result of alignment.

This paper describes a practical annotation tool GTLC (Ground-Truthing Text Lines and Characters) for unconstrained off-line handwritten Chinese documents. Unlike most previous works that align word boundaries, our aim is to align characters in text lines without word segmentation because Chinese texts have no extra space between words. We use a character recognizer trained with character samples of multiple writers for better tolerance to shape variability. On annotation, a document image is represented as a sequence of text lines, each line as an ordered sequence of connected components, which are partitioned into characters. We have validated the effectiveness of the tool and have applied it to annotate the document images in the HIT-HW database [9].

2. Overview of Annotation System

The overall approach of our annotation system is depicted in Fig. 1. The system loads a handwritten document image and the corresponding transcription, and outputs the ground-truth data (annotated image).

The system first extracts the connected components (CCs) from the input document image. The CCs are grouped into text lines. Each text line (as an ordered sequence of CCs) is over-segmented into primitive segments, which are aligned with the corresponding transcript.

The segmentation of text lines in handwritten documents is not a trivial task because the text lines are often un-uniformly skewed and curved, and the inter-

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.93

line space is not prominent. We have designed a text line segmentation algorithm based on minimal spanning tree (MST) clustering with distance metric learning [10], which performs fairly well but not perfectly. A post-processing procedure is thus needed to correct mis-segmented (split or merged) text lines manually. For example, Fig. 2(a) shows a text line image mis-split into two sublines. To correct, we first draw a box embracing the CCs of the left-side subline to be merged. Then, we double click in the region of the right-side subline for merging the CCs of the left-side subline (merged line in Fig. 2(b)). Similarly, for correcting mis-merged text lines, we first draw a box embracing the CCs of the merged text line, and then double click the place to separate.

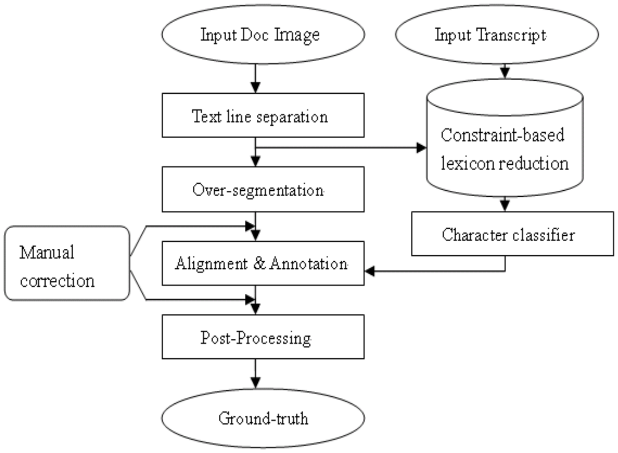
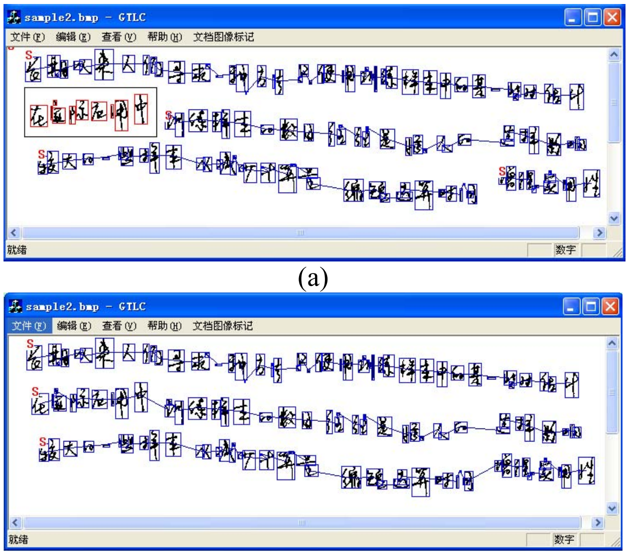


Fig. 1. Block diagram of the annotation system.


sample2.bmp - GTLC
文件(F) 编辑(E) 查看(V) 帮助(H) 文档图像标记
就绪
数字
sample2.bmp - GTLC
文件(F) 编辑(E) 查看(V) 帮助(H) 文档图像标记
就绪
数字

(a)

(b)

Fig. 2. (a) A box is drawn to embrace the connected components of a mis-split subline; (b) Merged text line.

After segmenting the document image into the same number of text lines as the text transcription, each text line image is then aligned with its transcript. For aligning character boundaries, the text line image is over-segmented into primitive segments with the hope that each primitive segment forms a character or a part of character. We use the over-segmentation technique of [11], where the CCs that are wide enough or have a large width/height ratio are examined to split.

After over-segmentation, each text line is represented as an ordered sequence of primitive segments. The text line alignment procedure can be viewed as a word (character string) recognition problem with a one-word lexicon. The optimal match can be found by dynamic programming (DTW). During alignment, candidate character patterns are generated by concatenating primitive segments and are assigned distance scores to characters in the transcript. After alignment, mis-segmentation and mis-labeling of characters (such errors are inevitable) are corrected manually.

3. Text Line Alignment

Each text line image is aligned with its corresponding text transcript. In the following, we consider a text line image and its text transcript and aim to find their optimal alignment.

3.1 Problem Formulation

After over-segmentation, a text line image is represented as a sequence of l primitive segments ordered from left to right: $I_m = \{I_{m1}, I_{m2}, \ldots, I_{ml}\}$. A primitive segment I_{mj} may contain a character or a part of a character. The corresponding transcript is a string of n characters: $T_m = \{C_{m1}, C_{m2}, \cdots C_{mn}\}$ ($n \le l$). An example of text line image and its text transcript is shown in Fig. 3.

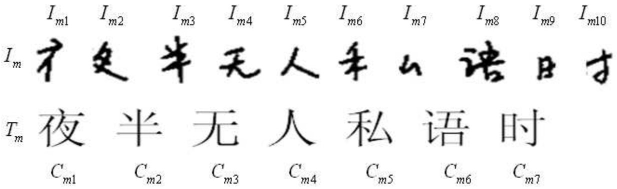


Fig. 3. An example of text line image and its text transcript.

A mapping Ψ of a text line image I_m and its transcript T_m is defined as:

$$\Psi = \{(C_{m1}, I_{m1\cdots mr}), (C_{m2}, I_{m(r+1)\cdots mp}), \cdots, (C_{mn}, I_{mn\cdots ml})\},$$

where $I_{m(r+1)\cdots mp} =< I_{m(r+1)} \cdots I_{mp} >$ is sub-sequence of primitive segments, which are concatenated into a character pattern. The problem of text line alignment is to find a best mapping $\Phi \subset \Psi$ that partitions the primitive segments to character patterns minimizing a distance measure (cost):

$$\Phi = \arg\min_{\Psi} \left\{ \sum_{i=1}^{n} d(C_{mi}, I_{mr\cdots mp}) \right\}, \qquad (1)$$

where $d(\cdot,\cdot)$ denotes the distance of matching a character pattern $I_{m(r+1)\cdots mp}$ with a character class C_{mi}. In our system, $d(\cdot,\cdot)$ is given by a character classifier inputting the feature representation of the character image.

3.2 String Alignment with DTW

The optimization problem (1) can be solved by dynamic programming (DP), also called dynamic time warping (DTW) in this context.

Without loss of generality, we drop off the index of text line and denote text line image $I =< I_1 \cdots I_l >$ and text transcript $T =< C_1 \cdots C_n >$. During dynamic alignment, a character pattern $< I_{j-k+1} \cdots I_j >$ is hypothesized to match character class C_i. We allow at most four primitive segments for a character pattern, namely, $1 \le k \le 4$.

To formulate the DTW alignment procedure, we define $D(i, j)$ as the accumulated cost of optimal alignment between a partial text line image $<I_1 \cdots I_j>$ and partial text $<C_1 \cdots C_i>$. $D(i, j)$ can be updated from the preceding partial alignments by DP:

$$D(i, j) = \min_k \begin{cases} D(i, j-1) + penalty(I_j) \\ D(i-1, j) + penalty(C_i) \\ D(i-1, j-k) + d_c(C_i, < I_{j-k+1} \cdots I_j >) \end{cases}, \quad (2)$$

where $d_c(C_i, < I_{j-k+1} \cdots I_j >)$ is the matching distance between character class C_i and pattern image $< I_{j-k+1} \cdots I_j >$, and is given by a character classifier, $penalty(I_j)$ is the cost of deleting primitive segment I_j, and $penalty(C_i)$ is the cost of skipping character C_i.

The DTW procedure starts with $D(0,0) = 0$, $D(i,0) = \infty$ and $D(0, j) = \infty$. Then for $i = 1, \cdots n$ and $j = 1 \cdots l$, $D(i, j)$ are updated according to (2). At each updating step, the optimal number k (number of primitive segments concatenated into a character) is stored for (i, j). Finally, $D(n,l)$ gives the total cost of optimal alignment, and the partition of primitive segments can be retrieved by backtracking the optimal numbers k from the (n,l) entry. Fig. 4 shows an example of text line alignment.

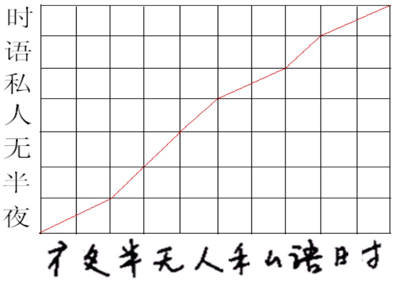


Fig. 4. An example of text line alignment.

3.3 Character Classifier

The matching distance $d_c(C_i, < I_{j-k+1} \cdots I_j >)$ between pattern image $< I_{j-k+1} \cdots I_j >$ and character class C_i is given by a character classifier, which is desired to have high classification accuracy and resistance to non-character patterns [12].

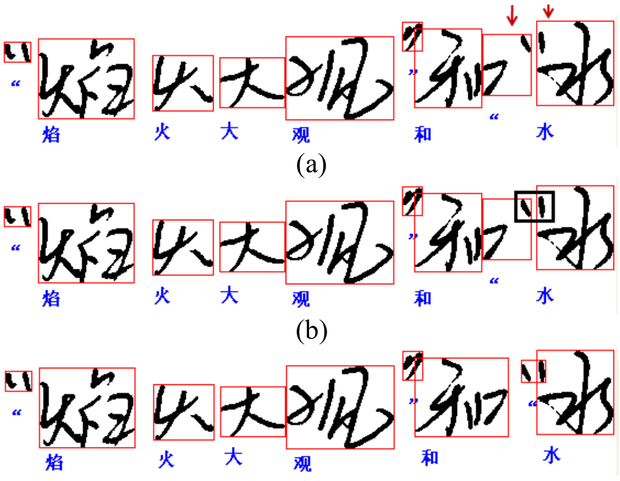
For large character set recognition, the modified quadratic discriminant function (MQDF) [13] is widely used for classification. For feature representation of character image, we extract 8-direction contour direction histogram feature using continuous NCFE (normalization-cooperated feature extraction) method combined with the MCBA (modified centroid boundary alignment) normalization method [14]. The resulting 512D feature vector is projected onto a 160-dimensional subspace learned by Fisher linear discriminant analysis (FLDA). The 160D projected vector is then fed to the MQDF classifier (7,356 classes: 7,185 Chinese characters, 10 Arabic numerals, 52 English letters and 109 other symbols). Given a character pattern and a class C_i, the classifier outputs the matching distance.

3.4 Post-processing

Due to the imperfection of pre-segmentation and imprecision of character recognition, some errors of character segmentation and labeling may remain, which are mainly of three types: miss error, alignment error and insertion error. The dominant error, alignment error, includes mis-split of a character into multiple ones and mis-merge of multiple characters into one. To correct such errors, we draw a box to embrace the primitive segments corresponding to a character, then the boxed character as well as the neighboring characters are adjusted automatically. Fig. 5 shows an example.

A miss error refers to the case that a character in the transcript has no corresponding image segments, due to missed writing or mis-merging the segment of the character with other characters (Fig. 6). In the latter case, the mis-merged primitive segments are drawn a box to correspond to the missed transcript character.

An insertion error refers to the case that a primitive segment has no corresponding transcript character, i.e., it is aligned with a non-character (denoted as “#”). This implies an extra image segment is inserted into the transcript text. Fig. 7 shows two examples of insertion errors. In Fig. 7(a), the segment labeled as “#” can be re-merged into the correct character “国” by drawing a box embracing the segments. In Fig. 7(b), the segments labeled as “#” are redundant, so no correction is needed.

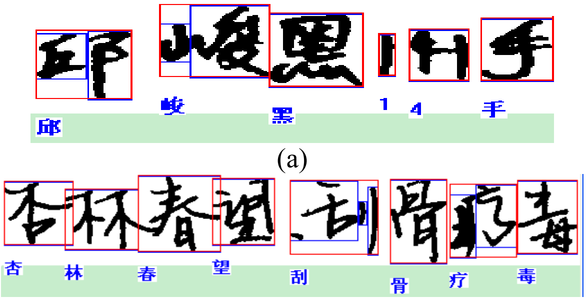


(a)

(b)

(c)

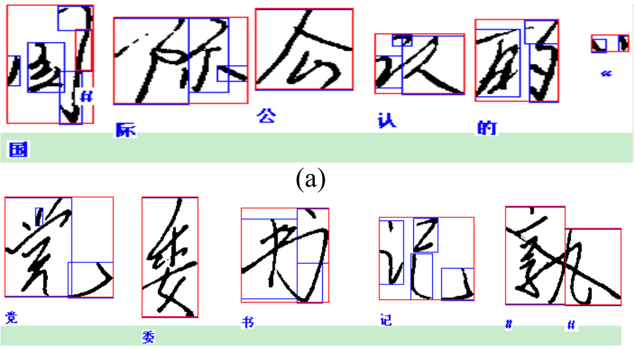
Fig. 5. Correction of alignment error. (a) An example of alignment error; (b) Draw a black box to embrace the primitive segments; (c) Corrected alignment.



(a)

(b)

Fig. 6. Two types of miss errors. (a) A “1” after “4” is missed due to splitting failure in pre-segmentation; (b) The segment of “、” is mis-merged into “刮”.



(a)

(b)

Fig. 7. Insertion errors. (a) A part of character “国” is aligned with “#” (non-character); (b) The last, redundant character is aligned with two “#”s.

4. Experiments

Our algorithm of text line segmentation has been evaluated in [10], and hereof we focus on the evaluation of text line alignment. The test set is a small subset chosen from the HIT-HW database [15], containing 383 text lines (8,424 characters in total).

In our experiments, after text line alignment, a match between a transcript character and the primitive segments, $(C_i, < I_k \cdots I_j >)$, is judged as correct if the bounding box of the primitive segments and the bounding box of the true character image overlap sufficiently. Overlapping implies that the difference of top, bottom, left and right bounds between two bounding boxes does not exceed a threshold (which is taken as the estimated stroke width [12]).

We report the recall rate (R) and precision (P) of characters (before post-processing is applied) for evaluating the performance of text line alignment. On 383 text line images, we observed a recall rate of 89.47% over 8,424 transcript characters and a precision of 89.13% over 8,453 aligned characters.

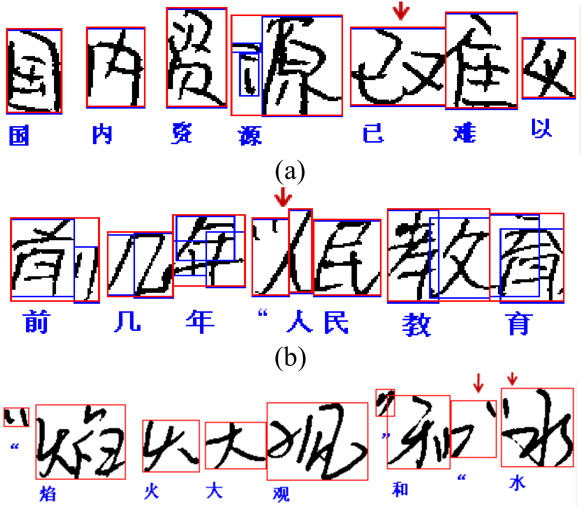
The rates of three types of errors (miss error, alignment error, insertion error) are shown in Table 1. It turns out that alignment error is dominant.

Table 1. Rates of three types of errors.

	Miss error	Alignment error	Insertion error
Error rate	0.095%	10.529%	0.438%

Our experiments observed three sources of alignment errors: (1) Touching strokes between characters are failed to split in pre-segmentation

(Fig.8(a)); (2) Heavy overlap of bounding boxes (Fig.8(b)); (3) Segmentation error between characters (Fig.8(c)). To improve alignment for such cases, we need to improve the pre-segmentation module and consider the geometric context (size and position in text line) of characters.



(a)

(b)

(c)

Fig. 8. Three types of alignment errors.

5. Conclusion

We proposed a recognition-based ground-truthing approach for annotating Chinese handwritten document images. This approach provides high accuracy of text line segmentation and character segmentation and labeling, and the remaining errors can be corrected manually. We have used the tool to annotate a large number of Chinese handwritten document images in the HIT-HW database, and the ground-truth data has been used for design and evaluation of our works in text line segmentation and character string recognition.

The performance of the annotation system, particularly, the text line alignment module, can be further improved via elaborating the pre-segmentation module, increasing the accuracy of the character classifier, and incorporating the geometric context of characters.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC) under grants no.60775004 and no.60825301.

References

[1] J. D. Hobby, Matching document images with ground truth, *Int. J. Document Analysis and Recognition*, vol.1, pp. 52-61, 1998.

[2] T. Kanungo, R.M. Haralick, An automatic closed-loop methodology for generating character groundtruth for scanned documents, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, No. 2, pp. 179-183, 1999.

[3] E.M. Kornfield, R. Manmatha, J. Allan, Text alignment with handwritten document, *Proc. Int. Workshop on Document Image Analysis for Libraries (DIAL)*, pp. 195-209, 2004.

[4] J. Rothfeder, T.M. Rath, R. Manmatha, Aligning transcripts to automatically segmented handwritten manuscripts, *Proc. 7th Int. Workshop on Document Analysis Systems (DAS)*, pp. 84-95, 2006.

[5] M. Zimmermann, H. Bunke, Automatic segmentation of the IAM off-line database for handwritten English text, *Proc. 16th Int. Conf. on Pattern Recognition*, vol.4, pp.35-39, 2002.

[6] B. Zhang, C. Tomai, S. Srihari, V. Govindaraju, Construction of handwriting databases using transcript-based mapping, *Proc. First Int. Workshop on Document Image Analysis for Libraries (DIAL)*, pp. 288-298, 2004.

[7] C. Huang, S.N. Srihari, Mapping transcripts to handwritten text, *Proc. 10th Int. Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pp.15-20, 2006.

[8] A. Kumar, A. Balasubramanian, A. Namboodiri, C.V. Jawahar, Model-based annotation of online handwritten datasets, *Proc. 10th Int. Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pp.9-14, 2006.

[9] T. Su, T. Zhang, D. Guan, Corpus-based HIT-MW database for offline recognition of general-purpose Chinese handwritten text, *Int. J. Document Analysis and Recognition*, vol.10, pp. 27-38, 2007.

[10] F. Yin, C.-L. Liu, Handwritten text line segmentation by clustering with distance metric learning, *Proc. 11th Int. Conf. on Frontiers in Handwriting Recognition*, pp. 229-234, 2008.

[11] C.-L. Liu, M. Koga, H. Fujisawa, Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.24, no.11, pp. 1425-1437, 2002.

[12] C.-L. Liu, H. Sako, H. Fujisawa, Effects of classifier structures and training regimes on integrated segmentation and recognition of handwritten numeral strings, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.26, no.11, pp.1395-1407, 2004.

[13] F. Kimura, K. Takashina, S. Tsuruoka, Y. Miyake, Modified quadratic discriminant functions and the application to Chinese character recognition, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 1, pp. 149-153, 1987.

[14] C.-L. Liu, K. Marukawa, Global shape normalization for handwritten Chinese character recognition: A new method, *Proc 9th Int. Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pp. 300-305, 2004.

[15] T-H. Su, T-W. Zhang, D-J. Guan, H-J. Huang, Off-line recognition of realistic Chinese handwriting using segmentation-free strategy, *Pattern Recognition*, vol.42, no.1, pp.167-182, 2008.