On-line Chinese Character Recognition System for Overlapping Samples

Xiang Wan, Changsong Liu
Department of Electronic Engineering, Tsinghua University
State Key Laboratory of Intelligent Technology and Systems
Beijing, 100084, P. R. China
Email: {wanxiang,lcs}@ocrserv.ee.tsinghua.edu.cn

Yanming Zou
Nokia Research Center
Beijing, BDA, 100176, P. R. China
Email: yanming.zou@nokia.com

***Abstract*—We proposed a new process strategy for on-line handwriting Chinese Character recognition and applied it to overlapping samples. On one hand, those samples are evaluated on stroke level by support vector machine; on the other hand, we do character level evaluation basing on a character pair search model. Then a merging strategy was proposed to filter out correct segmentation positions. We test our strategy on samples from real context, verifying that our strategy performs better than traditional over-segmentation and merging method.**

Keywords-OLCCR; Support Vector Machine; Character Pair Search.

I. INTRODUCTION

The On-line Chinese Character Recognition (OLCCR) system has been developing in recent years that the accuracy of single character recognition is higher than 98% [1]. Although some methods have been tried, it is very difficult to get better performance. With the in fashion of writing equipment, PDA, smart phone and so on, it is necessary for the system to recognize many characters at one time based on single character recognition technic. It requires the system be able to separate the character automatically. Practically, the research of OLCCR emphasize on the segmentation of characters more than the character recognition for the reason that the performance ascribe greatly to segmentation.

There are many researches in the literature about continuous writing words and sentence recognition. Compared with OLCCR, continuous writing offline Chinese Character Recognition system has been studied more. Generally, the recognition systems can be divided into two categories according to their strategies: segmentation-based systems and segmentation free systems [2]. Whether online or offline, the recognition systems usually adopt the former one. Tseng [3] uses stroke bounding boxes and knowledge-based merging operation to get candidate box, then a dynamic programming method is applied to find the best segmentation boundaries. Zhao [4] uses vertical projection and background skeleton as coarse segmentation. In the consecutive segmentation step, connected characters are found and segmentation positions are identified. FUKUSHIMA [5] uses a multi-layer perception(MLP) to segment characters.

Due to the prevalence of touch screen on PCs and Telephones, improving input efficiency becomes more and more important. We cannot write many characters as we can do on papers on a touch screen, and the overlapped writing is a way to improve character input speed.

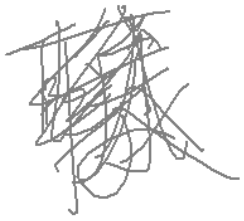

Fig. 1(a) overlapping sample

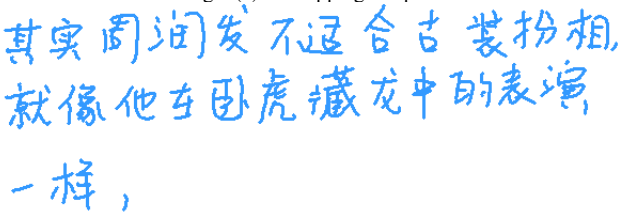

Fig. 1(b) traditional sample

Fig. 1(c) standard result

In this paper, we study the problem of how to segment the so-called overlapping characters as correct as possible.

Fig. 1(a) is an overlapping sample. It is recorded in a fixed box without moving the pen to a new place when writing a new character. Conventional writing style is shown in Fig. 1(b). The ideal final results should be characters that are separated correctly with corresponding recognition results, as shown in Fig. 1(c).

In this paper, we propose a support vector machine based segmentation position evaluation on stroke level and an adjacent segmentation positions search method on character level. All those information would be merged together by a

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.164

merging algorithm to generate the final segmentation path. This paper is organized as follows: Section 2 give an overview of our system, Section 3 presents our approach for evaluation of strokes and characters, Section 4 describes our merging strategy, section 5 provides results and perspectives, and concluding remarks would be given in Section 6.

II. SYSTEM ARCHITECTURE

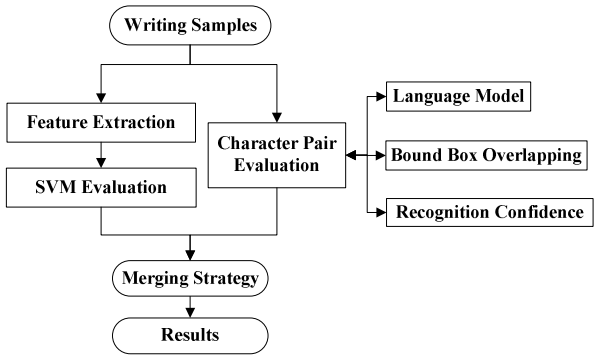

Fig. 2 block diagram of whole system

This system evaluates information of samples on two levels. On the left side of the processing flow, stroke features are extracted from handwritten samples to determine whether it's a segmentation position (class one) here or not (class two). We trained the support vector machine (SVM) with some writing samples randomly selected from the dataset and then it is used to classify the rest samples. On the final stage of classification, rather than classifying those samples into a specific class, a linear mapping is done to the predicting values to get the stroke level evaluation scores. On the right side of the processing flow, we define the concept of "Character Pair", which consists of three candidate segmentation positions and two strokes groups. This combination would be evaluated from three aspects: Language Model, Bound Box Overlapping and Recognition Confidence. Semantic Relation score can be computed through Language Model. Bound Box Overlapping reflects how the two characters overlap with each other geometrically. Recognition Confidence describes the probability that a specific candidate is really the character which is being recognized. After all those scores are calculated, we design a merging algorithm to generate the segmentation path and the final recognition result.

III. EVALUATION

The essence of this step is to utilize the information which embeds in the stroke flows of the samples. The more we get the correct information, whether geometrical or recognition-based or time sequential, the better the final results would be. Evaluation processing would be detailed as follow:

A. *Stroke level evaluation*

Our overlapping samples consist of consecutive strokes. Logically, each "gap" between two strokes is a segmentation position candidate (SPC). We define the concept of imaginary stroke: those virtual pen moving trajectories between two consecutive strokes. It's a straight line from the end point of a stroke to the start point of the next stroke. As shown in Fig. 3. It is the first two strokes when we write Chinese character "文". We evaluated each imaginary stroke to determine whether it is a segmentation position between two characters here. Support vector machine (SVM) is introduced to complete this process automatically. Features used are listed in Table 1. They are also depicted in Fig. 3.

TABLE 1 FEATURES OF IMAGINARY STROKE

Features	Description
X1	X-coordinate of end point in current stroke
Y1	Y-coordinate of end point in current stroke
X2	X-coordinate of start point in next stroke
Y2	Y-coordinate of start point in next stroke
Length	The length of imaginary stroke
LeftRight	Varible marking relative position of two strokes horizontally
UpDown	Varible marking relative position of two strokes vertically

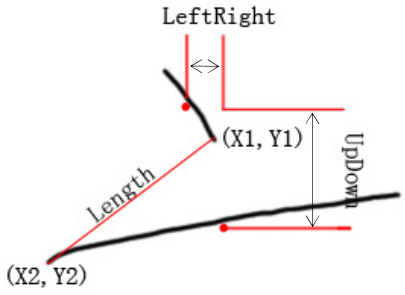

Fig. 3 imaginary stroke and its features

Traditionally, after training the SVM, classification results (whether it's segmentation position or not) are obtained basing on the sign of output values. The segmentation results are not good enough if we just adopt SVM classification. To make the information on stroke level can be merged with other information; we do a linear mapping to output values. The histogram of output values are shown in Fig. 4:

The output value varies between a minimum value and a maximum value, denoted as min_value and max_value respectively.

We calculate SVM score via, which has a range $0 \sim 100$:

$$C_{SVM} = \frac{value - min\_value}{max\_value - min\_value} * 100$$

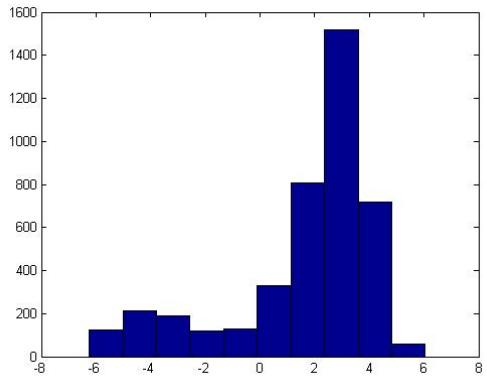

Fig. 4 histogram of output values

B. Character pair search and evaluation

Previously, common strategy for character level evaluation is to find a global optimum path. Dynamic programming, Beam search, multilayer-perception has been used. Those strategies have a concrete objective function and corresponding processing procedures. But their weakness is that once a wrong segmentation has been done during processing, segmentation for adjacent strokes would probably be affected during this procedure.

Our main concern is to utilizing consistency constraint of the character pair to simplify the global search to part evaluation, without losing information.

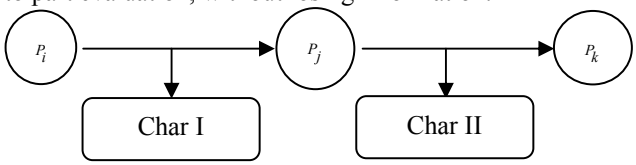

Fig. 5 character pair search

Fig. 5 is the so-called Character Pair (CP) for character level evaluation. Suppose that $S=\{s_1,\cdots,s_N\}$ is the input strokes set, and $P=\{p_1,\cdots,p_{N+1}\}$ is the segmentation position candidates (SPC) set. Two consecutive strokes contain a segmentation position candidate lying between them. The pattern for strokes and segmentation position candidates is $\{p_1,s_1,p_2,\cdots,p_N,s_N,p_{N+1}\}$. Suppose that p_j is the segmentation position candidate we are currently evaluating. We set a pre-candidate p_i and pro-candidate p_k to generate a CP with three SPCs: $p_i p_j p_k$, where $0<j-i<T$, $0<k-j<T$ (T is a threshold set by the system). Strokes between p_i and p_j compose character I. Similarly, strokes between p_j and p_k compose character II. Then we utilize this model to evaluate each segmentation position candidate through three aspects: Language Model, Bound Box Overlapping and Recognition Confidence, detailed as follows:

1) Language Model

The same as off-line handwritten Chinese character recognition, online characters also have semantic relations. We train character-based bigram language model to get the transition probability $p(c_2|c_1)$. That is the occurrence probability of char II when the previous character is char I. Semantic Relation Score would be:

$$C_{LM}=\left[ln(p(c_2|c_1))+25\right]*4$$

The language Model is built basing 50MB linguistic data from People's Daily, which is an official newspaper in China. Good-Turing Estimation is adopted to get the transition probability matrix [6]. Character set is GB2312, containing 6763 characters in all from level one and level two character set. Probability less than a threshold would be set as e^{-25}, so the Semantic Relation Score would be in the range of $0\sim100$.

2) Bound Box Overlapping

Samples we are going to tackle with are overlapping handwriting characters. The more the two characters overlap with each other, the more likely the character pair is correct for the sample. We calculate this score via the ratio of overlapping area and the area which the sample occupies as follow:

$$C_{BBO}=\sqrt{\frac{O\_area}{area}}*100$$

It is also in the range $0\sim100$.

3) Recognition Confidence

For each character, we give ten recognition candidates for evaluation of the character pair search. It is necessary for us to give each candidate a corresponding score to access how reliable the candidate is truly the character.

Lin[7] proposes the concept of "general recognition confidence(GRC)" and a mapping relation so that the recognition confidence(RC) of the first candidate could be calculated. The mapping relation is illustrated in Fig. 6:

We calculate GRC for the first candidate through $c_1=1-\frac{d_1}{d_2}$, then it is mapped to $p(w_1|x)$. For the rest candidates, we adopt a recursive method to get their scores.

For number N candidate, the sum confidence of candidates from number N to number ∞ (if exists) would be $\sum_{i=N}^{\infty}p(w_i|x)=1-\sum_{i=1}^{N-1}p(w_i|x)$. If we do not take the first

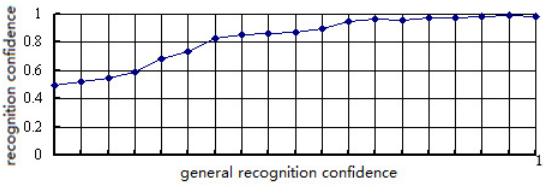

Fig. 6 mapping function(cited from [6])

N-1 candidates into account, number N candidate would be the "first candidate" of remaining candidates. Similarly, we can get its GRC through $c_N = 1 - \frac{d_N}{d_{N+1}}$, which is then mapped to $p_{temp}(w_N|x)$ by Fig. 6. Summing up the above, the RC for number N candidate is

$$p(w_N|x) = \left[1 - \sum_{i=1}^{N-1} p(w_i \mid x)\right] * p_{temp}(w_N|x)$$

Then the RC score is

$$C_{RC} = p(w_i|x) * 100$$

It's also has a range $0 \sim 100$.

IV. MERGING STRATEGY

After stroke level evaluation and character level evaluation as described above, four different scores have been generated for a specific character pair. Those scores would be merged to one score showing how reliable this combination is truly a splitting combination:

$$C_M = \sum_{TYPE} \lambda_{TYPE} C_{TYPE}$$

In which $TYPE$ refers to one of $\{SVM, LM, BBO, RC\}$.

Then we process combination information according to C_M. The algorithm is below:

1) Initialize. We denote the flag for all segmentation positions as F. $\forall f_j \in F, f_j = 0$
2) We select the best three CPs according to C_M for each SPC, denoted as $\{P_j\}_3$. All of $\{P_j\}_3$ constitute a set $A = (\{p_1\}_3, \{p_2\}_3, ..., \{p_{N+1}\}_3)$
3) Do a voting on the best N elements of A: if a SPC appears in one elements of A, add its count by one, if count for any SPC exceeds TH, set its flag f_j to be 1. Then the whole path is divided into several sections.
4) For each section, we evaluate the remaining SPC in this section according to their CPs and corresponding C_M by descending order of C_M. If one CP with fewer score violates with one having higher score, it would be discarded.
5) The algorithm ends. Final segmentation results would be presented.

N is set to the number of strokes when processing a specific sample, and TH is related with the average number of strokes per character in samples, usually we set it to one and a half times as the average number of strokes.

V. EXPERIMENT RESULTS

We implement the proposed approach by C++ language. The sample set is collected manually. There are 1000 samples with 6845 characters. Numbers of characters in a sample varies from 2 to 15. Those samples are short messages used in daily communication scenarios. The segmentation ground-truth is marked manually.

We adopt two strategies to segment samples mentioned above. Strategy one uses a SVM to do pre-segmentation and then use a Dynamic Programming (DP) algorithm basing on a lexicon, which as the traditional over-segmentation and merge strategy does[8]. Strategy two uses method this paper proposes. We can see the comparison of them in Table 2.

There are two kinds of correct rates: SCR is short for segmentation correct rate, which is the percentage of classifying SPC correctly to its type in ground-truth. CSCR is short for character segmentation correct rate, which is the percentage of characters with both left and right boundaries matched with the ground-truth boundaries:

There are two kinds of errors: touch refers to the situation that a SPC is classified to be a segmentation position while it is not one. Conversely, over-segmentation means that a SPC is classified to be a non-segmentation position while it is truly one.

TABLE 2 PERFORMANCES ON TEST SET

Index	SVM+DP	Evaluation-Merging
SCR	90.93%	96.91%
CSCR	25%	86.4%
Touch	8.28%	1.18%
Over-segmentation	0.79%	1.91%

We can see a great improvement when using our strategy instead of DP algorithm. Some segmentation-recognition results are shown in Fig. 7. Three Figures represents successful segmentation, segmentation with over-segmentations and segmentation with touch errors separately. The two characters circled in Fig. 7(b) should be one character "本", and the character circled in Fig. 7(c) should be two characters"备过".

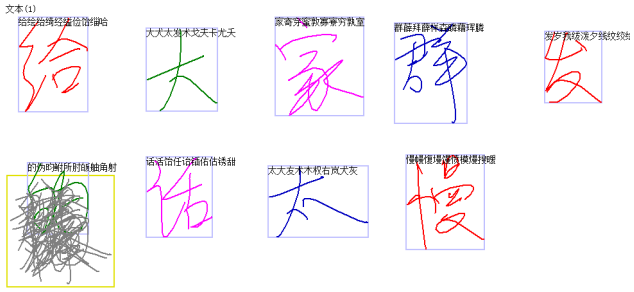

Fig. 7(a) sample successfully segmented

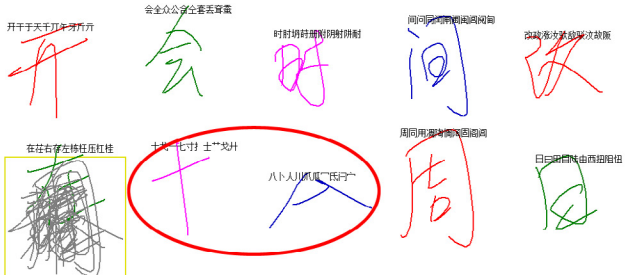

Fig. 7(b) sample with an over-segmentation

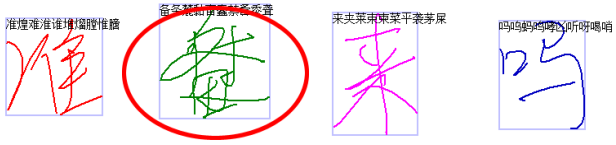

Fig 7(c) sample with a touch error

VI. CONCLUDING REMARKS

In this paper, we propose a method for overlapping samples in OLCCR. The method do both stroke level and character level evaluation on samples in real contexts, utilizing geometrical, linguistic, recognition-based information. The result of character-segmentation is better than using traditional methods. Future improvement relies on more effective features for SVM, recognition core with higher accuracy and more accurate language model. We can notice that touch error cannot be rectified once it occurs, while over-segmentation errors can be rectified by post-processing. To make it into practice, multi-segmentation path could be selected to decrease over-segmentation errors further. On the other hand, the merging algorithms can be improved to decrease touch errors.

ACKNOWLEDGMENTS

This work was supported by the National Basic Research Program of China (973 program) under Grant No. 2007CB311004 and the National Natural Science Foundation of China under Grant Nos. 60772049, 6093310.

REFERENCES

[1] Q. Fu, X. Ding, T. Li, C. Liu, "An Effective and Practical Classifier Fusion Strategy for Improving Handwritten Character Recognition," Document Analysis and Recognition, International Conference on, pp. 1038-1042, Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2, 2007

[2] T.-H. Su, T.-W. Zhang, D.-J. Guan and H.-J. Huang, Off-line recognition of realistic Chinese hand writing using segmentation-free strategy, Pattern Recognition **42** (2008), pp. 167–182.

[3] Tseng L Y, Chen R C. Segmenting handwritten Chinese characters based on heuristic merging of stroke bounding boxes and dynamic programming[J]. Pattern Recognition Letter

[4] Zhao S, Chi Z, Shi P, etc. Handwritten Chinese Character Segmentation Using A Two-stage Approach, Proc. 6th Int. Conf. Document Analysis and Recognition, Seattle, USA, Sep. 2001, IEEE Computer Society Press: 179-183.

[5] FUKUSHIMA Takahiro, NAKAGAWA Masaki. On-line Writing-box-free Recognition of Handwritten Japanese Text Considering Character Size Variations[A]. ICPR'00, 2000,2: 359-363.

[6] Chen S F, Goodman J. A empirical study of smoothing techniques for language modeling. Proceedings of the Thirty-Fourth Annual Meeting of he Association for Computational Linguistics, 1996:310-318.

[7] Xiaofan Lin, Xiaoqing Ding, Youbin Chen, Jinhui Liu and Youshou Wu, "Evaluation and Application of Recognition Confidence in OCR", *Proceedings of ACCV'98*, Jan. 1998, Hongkong.

[8] Zhengbin Yao, Xiaoqing Ding, Changsong Liu "On-line handwritten Chinese word recognition based on lexicon", Proc. 18th Int. Conf. Pattern Recognition, Hong Kong, China, Aug. 2006, pp. 320-323.