A Handwritten Chinese Address Recognition Method Using Recursive Holistic Word Matching and Edit Distance Based Verification Strategies

Kaizhu Huang, Yoshinobu Hotta, Satoshi Naoi

A Handwritten Chinese Address Recognition Method Using Recursive Holistic Word Matching and Edit Distance Based Verification Strategies

Kaizhu Huang
kzhuang@frdc.fujitsu.com
Fujitsu Research and Development Center Ltd
Beijing, P.R.China

Yoshinobu Hotta, *Satoshi Naoi*
{y.hotta, naoi.satoshi}@jp.fujitsu.com
Fujitsu Laboratories Ltd
Kawasaki, Japan

Abstract

With extensive applications in many domains, e.g., in postal address recognition, bank cheque recognition etc, Handwritten Chinese Address Recognition (HCAR) has attracted much interest recently. However, due to the large number in Chinese characters categories, irregular distortion and touching in handwriting, and high variations in writing styles, this task presents a big challenge in the community of pattern recognition. In this paper, we propose a novel three-stage approach to attack the Handwritten Chinese Address Recognition problem. More specifically, we propose the framework of coarse recognition, fine recognition, and verification as well as some new techiqniues tailored to each stage for HCAR. Coarse recognition is designed for speeding up the system; fine recognition exploits a novel recursive and enhanced segmentation-free holistical word recognition method in order to overcome the difficulties in HCAR; verification uses an important modified edit distance definition for further lifting the system accuracy. A series of experiments are conducted to evaluate the proposed novel system. The results demonstrate that our system outperforms other traditional methods in three real data sets.

1 Introduction

Handwritten Chinese Address Recognition (HCAR), an important and challenging pattern recognition problem, has attracted much interest recently. This technology can be applied in many domains including postal address recognition and bank cheque recognition. The basic task of this problem is to recognize the actual address directly from an input address image, which can be obtained by either a scanner or a digital camera.

To deal with this problem, one intuitive method is the plain recognition method. This method first segments each character from the input image, and then recognizes the isolated characters one by one. However, there are many difficulties for this approach. For example, free style in handwriting will lead to inevitable segmentation errors; the large number of categories in Chinese characters[1] will cause high misclassification rate for isolated character recognition. Moreover, this bottom-up recognition strategy (e.g., from single characters recognition to the address) makes it difficult to utilize the address knowledge. Recently some researchers propose some advanced segmentation method for HCAR, e.g., Hidden-Markov-Model based segmentation [1], stroke cross number analysis [2]. But these methods are still founded in plain recognition. Therefore their performance will be restricted. Different from the above approach, in this paper, we engage the framework of segmentation-free holistic word recognition [3][4]. We first extract the key characters, which are defined as the basic administration units, such as 省 (province), 市 (city), 区 (district), 路 (road), and so on. We then base the address knowledge to recognize the words holistically, which are defined as a sequence of characters (the place-name) between two key characters in an image address.

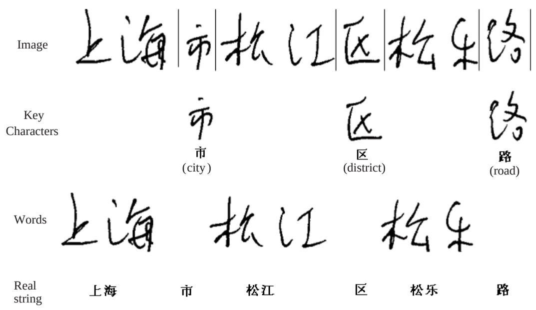


Figure 1: A typical example of key characters and words

Figure 1 illustrates a typical example. The key characters 市 (city), 区 (district), and 路 (road) are firstly extracted from the image. Then the word images between each pair of key characters or before the first key character (the image of the word 上海) are segmented. These images are then holistically matched with the synthesized features of the place names (stored in the reference dictionary). Finally, the real string 上海-市-松江-区-松乐-路 is output.

[1]Typically, there are 6763 categories for the first and second level simplified Chinese characters.

Extracting and recognizing key characters from the address image elegantly transforms the large-category pattern recognition task, i.e., the common Chinese character recognition task, to the small-category key characters recognition (with only 22 key characters). This therefore provides potentials to improve the accuracy significantly. Holistically recognizing the words between each pair of key characters enables the system the segmentation-free merit. This further overcomes the difficulties caused by segmentation.

Furthermore and more importantly, we propose a three-stage framework, i.e., coarse recognition, fine recognition, and verification for lifting the system's efficacy. Coarse recognition delivers speed-up for the system; fine recognition utilizes a recursive and holistic word matching approach that improves the accuracy significantly; verification further corrects the recognition results by a novel modified edit distance definition. With all these techniques, the system presents a comprehensive approach for the HCAR problem. Our system is significantly different from our previous method [6] in that various techniques have been proposed in the new system: coarse recognition, recursive and enhanced holistic matching, and verification are newly designed for the system.

This paper is organized as follows. In the next section, we present our system in detail. In Section 3, we present a series of experiments to evaluate our proposed system. Moreover, some analysis will be conducted in this section. Finally, we set out the conclusion in Section 4.

2 Novel HCAR

In this section, we first describe the overall framework of the system. We then introduce coarse recognition, fine recognition, and verification sequentially.

2.1 Overall Framework

2.1.1 Notations

We define several notations for clarity.

Definition 1 *A **Key Character** - is defined as one of the* 22 *basic administration units which are* 市,省,区,弄,路,街,村,乡,镇,港,湾,县,道,里,同,巷,楼,州,旗,胡,庄,坊.

Definition 2 *A **Word** - is a place name. In an address image, it comes either in the beginning of the address or between a pair of key characters.*

In Fig. 1, 上海, 松江,松乐 are *words*.

Definition 3 *A **Key Character Path** - is a set of key characters that satisfies the hierarchy structure relationship in Chinese Address.*

For example, 市 (City)-街 (Street) is a legal key character path, while 街 (Street)-市 (City) is not a key character path because a street can locate in a city but a city never locates in a street. Part of key character hierarchy structures is illustrated in Fig. 2.

```
市----区
  ----区----路
  ----区----路----{村，里，同，楼，街，道，巷，庄}
  ----区----街
  ----区----街----{村，里，同，楼，路，道，巷}
  ----区----{村，巷，里，同，镇，乡，弄，湾，楼，港，庄}
  ----县
  ----县----乡
  ----县----乡----{村，路，庄，湾，街，里，巷}
```

Figure 2: Part of the Chinese key character hierarchy structure

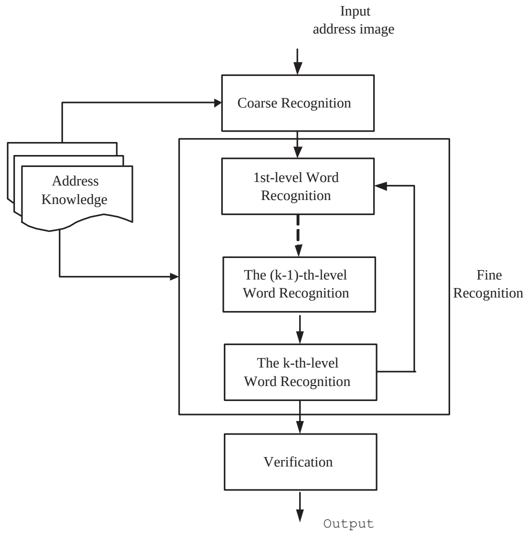


Figure 3: The overall framework of the system

Definition 4 *An **Address Level** - is defined as a word and a key chracter which is associated with the word.*

For example, in Fig. 1, 上海市, 松江区, 松乐路, are address levels. Moreover, 上海市 is the *upper level* of 松江区, and 松江区 is the *lower level* of 上海市. Similar relationship can be also defined in 松江区 and 松乐路. Typically, in Chinese address, the number of the address levels in one address will not exceed 4.

Definition 5 *A **Legal Address** in an address level is defined as a place name which locates within the place specified by the previous address level (if exists the previous level) and is associated with the key character in this address level.*

For example, the *legal addresses* in the third level of Fig. 1 are all the place names of 路 (Road) which locate within the place 松江区.

2.1.2 Framework

In this section, we describe the overall framework of the system. When an input address image is input for recognition, the coarse recognition module is called to select the first P legal key character paths according to address knowledge and the matching distance. In this module, key characters will be first extracted. Following that, the fine recognition module will recursively evaluate each legal key character path and recognize the place name in each address level,

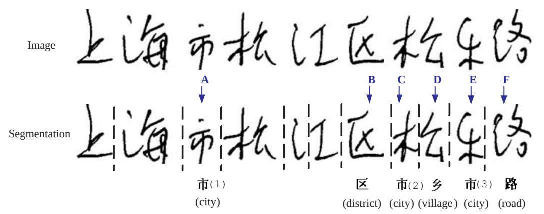


Figure 4: Illustration of the key character extraction

i.e., the word between each pair of key characters holistically. Next, the verification will be applied to verify and correct the recognition results. Finally, the recognition result will be output. In the following, we will explain each module in detail.

2.2 Coarse Recognition

Coarse recognition is used to extract the key characters and select those highly possible legal key character paths. In the following, we will describe the steps in detail.

2.2.1 Key Character Extraction

The module of key character extraction attempts to find key characters from the input address image. Fig. 4 illustrates an example of key character extraction. Vertical projection is first used to segment the images into small parts. Each part is then recognized whether it is a candidate key character. Missing a correct key character definitely results in the failure of the whole system. Therefore, we accept a small part as a key character if only its matching distance with a key character is smaller than a given threshold. In Fig. 4, 6 small parts ($A \sim F$) are recognized as the candidate key characters. Note that although extracting key characters will also first segment the address image, the recognition objective here is to find only 22 key characters. This is much easier than the plain recognition which will conduct a large-category (in particular 6763) recognition task.

2.2.2 Key Character Path Filtering

After the key characters are extracted from the address image, key character paths will be formed. Back to Fig. 4, 6 key characters will be combined as key character paths. Excluding those illegal key character paths (e.g., D-E-F because 乡 (village) never cover 市 (city) as its administrative unit), the key character paths can be $A - B - F$ (市(1)-区-路), $A - D - F$ (市(1)-乡-路), $A - F$ (市(1)-路), $B - C - F$ (区-市(2)-路) etc.

To reduce the time complexity, those first P key characters with the smaller average matching distances are selected as the key character paths. The purpose of the coarse classification is to include those possible addresses as precise as possible while excluding those illegal and highly impossible addresses simultaneously. Typically, we use a relatively larger threshold to include more candidate key characters. Then the average distance ranking and the address knowledge are exploited to exclude less possible address paths.

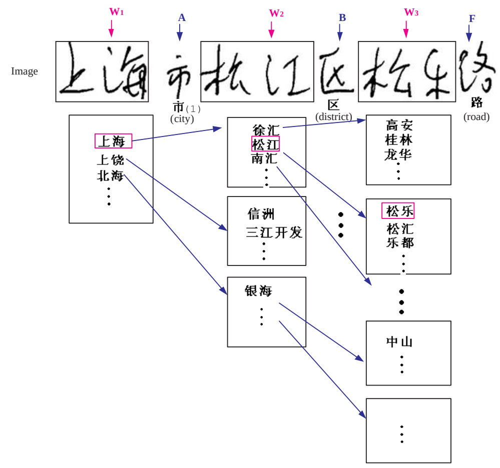


Figure 5: Recursive and holistic word recognition

2.3 Fine Recognition

After coarse recognition, fine recognition will evaluate each address path and then choose the best one as the recognition result. More specifically, for each key character path, the recursive and holistic word recognition recognizes the words (place name) between each pair of key characters.

2.3.1 Holistic Word Recognition

Different from the plain address recognition method, the images between each pair of key characters are recognized as a whole. Beginning from the first address level, the word image is segmented and recognized as a place name. In the next address level, the word image is cut out and the features are extracted from it. These features are then compared with the synthesized features of those place names; these place names must be those of the administrative units specified by the key character in this level and must locate in the place specified by the recognition result in the last level. Similar process is conducted until all the address levels are recognized. This scheme avoids the difficult single character segmentation problem and therefore increases the accuracy of the system. Detailed information about the feature synthesis and holistic matching can be seen in [3].

2.3.2 Recursive Holistic Word Recognition

As seen in [3][6], the above holistic word recognition only adopts the first candidate in each level. However, handwriting is of free styles. Moreover, sometimes, two words (place names) contain very similar shapes. Only choosing the first candidate may generate many errors. Therefore, we propose an enhanced holistic word recognition approach. Our approach utilizes multiple candidates and recursively performs holistic word recognition in each address level. For solving the speed problem, first, we have applied coarse recognition. Second, as seen in Section 2.3.3, we propose a trimming-down strategy. Both strategies are demonstrated to speed up the whole system greatly.

Fig. 5 illustrates the detailed procedure. In this figure, assuming the key character path $A-B-F$ is evaluated, the words are W_1, W_2, and W_3. Firstly, features are extracted from the image of W_1 and then they are matching with all the 市 (city) as indicated by A (e.g., 北京, 上海) in the reference dictionary. The candidate words are 上海，上饶，北海 which are sorted by matching distances. Each candidate will be recursively evaluated in matching the image W_2. For example, the image of W_2 is compared with all the place names which are 区 (as indicated by B) and locate in 上海市, 上饶市, 北海市 respectively. As a result, three candidate lists of W_2 are generated for 上海市, 上饶市, 北海市 respectively. Similar process will be conducted on these candidates to match the image of W_3. The recognition result is the path which has the smallest matching distance. In this example, it is 上海-市-松江-区-松乐-路.

Note that, the first recognition candidate of W_2 is not the correct word recognition result 松江. By using the recursive matching, it is selected as the recognition result. In such a way, the recognition accuracy will be lifted.

2.3.3 Trimming

In the above, we adopt the multiple-candidate strategy to increase the system's accuracy. However, the time complexity will be increased simultaneously. Assume there are k levels in an input address and N candidates are used in recognizing each word address. The total number of combinations in one key character path is hence N^k. This will be very time-consuming. To speed up the whole process, we design a trimming-down strategy as follows:

Rule 1: The maximum number of candidates should be less than a given number K.

Rule 2: Only the candidates that satisfy the condition (1) will be evaluated.

$$\frac{Dist(Cand_i)-Dist(Cand_1)}{Dist(Cand_1)} < Th_1 \ . \tag{1}$$

In the above, $Dist(Cand_i)$ represents the matching distance of the i-th candidate. Th_1 is a predefined threshold, which is set to 0.125 in our system. Rule 1 will restrict the maximum number of the candidates. Rule 2 will not search those candidates whose matching distance is far from the first candidate. In this way, not all the combinations will be evaluated. Hence, the processing time can be reduced greatly. This will be verified later in the experiments.

2.4 Verification

When the words in the upper levels are mis-recognized, the word images in the latter levels will be absolutely misrecognized. Hence the matching distances in the latter levels will be large. This means the recognition result of the latter levels can verify the recognition of upper levels. However, for the last level word recognition, there is no such verification. Moreover, as counted in practice, recognition errors occuring in the last level address accounts for 25% of misclassification. Therefore, for correcting the last level address, we propose to combine the plain character recognition result with the holistic word recognition result in the last level. We first provide a definition of the modified edit distance, which is different from its traditional definition.

Definition 6 - ***Modified Edit Distance*** *Assume that $S=\{S_1,S_2,\ldots,S_u\}$ is a place name, and W is a $C\times K$ array, where S_k $(1\le k\le u)$ represents the k-th character in the place name, W_{ij} ($1\le i\le C, 1\le j\le K$) represents the j-th candidate for the i-th segmentation part in a word image, W_i represents the candidate list of the single character recognition result for the i-th segmentation part. The modified edit distance between S and W is defined as the minimum cost at which W is changed to S by the operations of insertion, substitution, and deletion. The cost between S_k and W_i is defined as follows:*

$$Cost(S_k,W_i)=\begin{cases}1 & if\ \forall j \quad W_{ij}\neq S_k \\ \frac{j}{CK} & if\ \exists j \quad W_{ij}=S_k\end{cases} . \tag{2}$$

In the above, C can be considered as the total number of the segmentation parts for the word images in the last level, and K can be regarded as the maximum number of candidates for each segmentation part.

In this procedure, in order to utilize the candidates of the plain results for the last level address image, we modify the concept of the edit distance which is designed originally for comparing two strings to that for comparing a string (a place name or address) and the string array (the plain recognition result and its candidates).

The detailed verification steps are described in the following:

1. Base the Dynamic Programming [5] to calculate the modified edit distance between each legal address in the last level and the plain recognition result for the last level image. All these legal addresses are sorted according to their modified edit distances.
2. Get the word recognition results from fine recognition and rank each legal address according to the matching distance.
3. Output the legal address A_f as the final recognition result according to the following decision rule as defined in Definition 7.

Definition 7 - ***Decision Rule*** *(1) If a legal address contains the edit distance smaller than or equal to 1, output the address with the minimum modified edit distance as the recognition result. (2) If all the legal addresses contain the edit distance is greater than or equal to C, the number of the connected components, output the word recognition result as the final result. (3) If (1) and (2) are not satisfied, the weight is calculated as Eq. (3) and output the legal address with the smallest weight as the final result.*

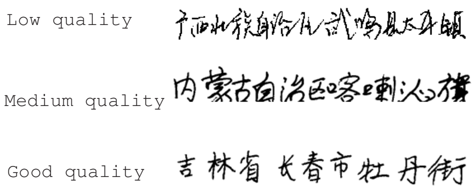


Figure 6: Three test samples in the low, medium, and good quality

$$Weight(A_i) = (1 - t_1)Rank_{ED}(A_i) + t_1 Rank_{WR}(A_i) \quad (3)$$

where, A_i is i-th legal address, $Rank_{ED}(A_i)$ means the rank of the modified edit distance between A_i and the plain recognition result W, $Rank_{WR}(A_i)$ means the rank of A_i by the word recognition. t_1 is defined as $t_1 = round(ed(A_i))/C$; $ed(A_i)$ represents the modified edit distance between A_i and the plain recognition result.

The decision rule is justified in the following. From the cost definition in Eq. (2), with Rule (1) satisfied for a legal address, each character in the legal address should occur in the candidate list of plain recognition. This actually implies a verification for this legal address. Therefore we should output the address with the minimum modified edit distance as the recognition result. On the other hand, with Rule (2) satisfied for all legal addresses, none of the characters in these addresses occurs in the candidate list of the plain recognition result. This actually implies that the plain recognition is highly unreliable. Therefore, the holistic word recognition result should be output.

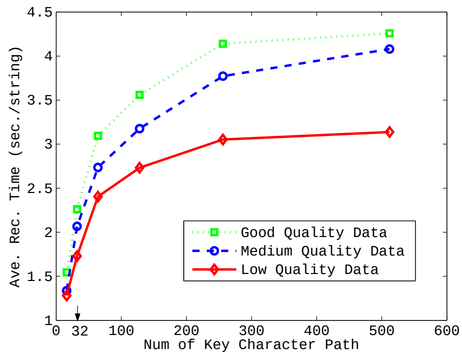
Rule (3) actually combines the plain recognition result with the holistical word recognition result. When the edit distance of A_i is very small, the plain recognition result appears highly reliable. Therefore we should give more weight to $Rank_{ED}(A_i)$; otherwise, $Rank_{WR}(A_i)$ should make more contributions. In particular, when the edit distance of A_i is less than or equal to 1, we should trust the result of plain recognition; when the edit distance of all the legal address is big enough, the plain recognition should be very unreliable, leading that we should output the word recognition result as the final result.

3 Experiments

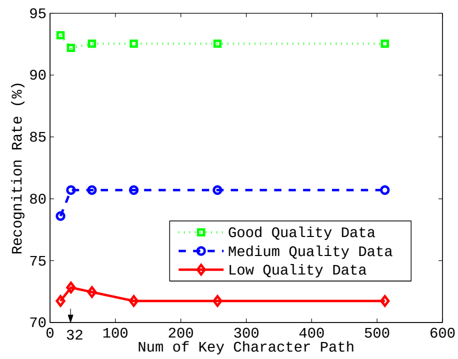
In this section, we will first introduce the experimental setup. We then report the experimental results.

3.1 Setup

Three data sets are used to evaluate the performance of the new system. These data sets, which are of the low, medium, and good quality respectively, consist of 854 images. There are 276, 285, 295 images in low, medium, and good quality data sets respectively. These images are collected from the real users. Note that we follow [6] and do not consider



(a)



(b)

Figure 7: Analysis of coarse recognition

the address part after the last key character (this part might be the building name, room number etc). Figure 3 provides one example for each data set. The experiments are conducted by using Microsoft Visual C++ 6.0 in the platform of Windows XP with the CPU of 3.2 Ghz.

3.2 Results

We compare our approach with the plain recognition approach (PRA), the previous holistic word matching approach (HWA) on the three data sets. In order to see the effect of verification, we present the accuracy of our system without vefification (Our Appr.) and with veification (Our Appr.+ Veri.). We use String Recognition Rate (SRR) as the metric. SRR is defined as

$$SRR = \frac{\text{The number of correctly recognized addresses}}{\text{The total number of addresses}}. \quad (4)$$

Dataset	Low Quality	Medium Quality	Good Quality
PRA(%)	0.72	3.86	29.05
HWA(%)	55.80	66.32	85.76
Our Appr.(%)	65.58	75.79	88.14
Our Appr.+Veri.(%)	**71.74**	**80.70**	**92.54**

Table 1: String Recognition Rate in three data sets

The results are reported in Table 1. As observed from the table, our novel three-stage approach (Our Appr.+Veri.) demonstrates the best performance. Due to the inevitable segmentation errors and the free styles in handwriting, PRA's performance is the worst in all the three data sets. HWA uses the holistic word matching method; but it lacks the recursive evaluation scheme and the verification. Even without applying verification, our approach demonstrates the best when compared with PRA and HWA. After the verification is applied, the accuracy is further lifted. Our system presents a comprehensive method. The final improvement is significant. Compared to HWA, an accuracy increase of 15.94%, 14.38%, 6.76% is respectively achieved for the low, medium, and good quality data sets.

3.3 Analysis

After obtaining the conclusion that our approach outperforms the previous methods, we turn to examining how the coarse classification and trimming-down strategy influence the system performance.

3.3.1 Analysis of Coarse Recognition

We examine how coarse recognition affects the system's performance in the following. Fig. 7 (a) plots the average recognition time vs the number of the key character paths; (b) plots the recognition accuracy vs the number of the key character paths. By controlling the key character paths (which are selected by the average matching distance of key characters), coarse recognition can reduce the processing time greatly while maintaining the accuracy nearly unchanged. In the system, we choose the typical point 32, i.e., the number of key character path is equal to 32. When coarse recognition is not applied (not restricting the number of the key character paths), the average processing time by using 32 is at least reduced by half, while the accuracy is nearly unchanged as seen in Fig. 7.

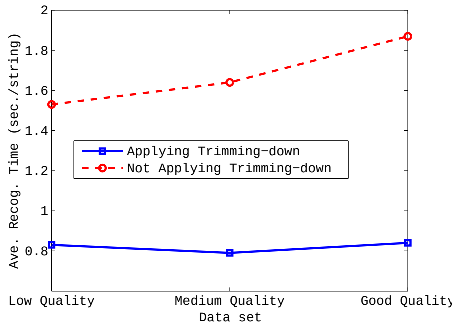
3.3.2 Analysis of Trimming-down Strategy

To further examine how the trimming-down strategy influences the system performance, we evaluate our performance of our new system with and without applying the trimming-down method.

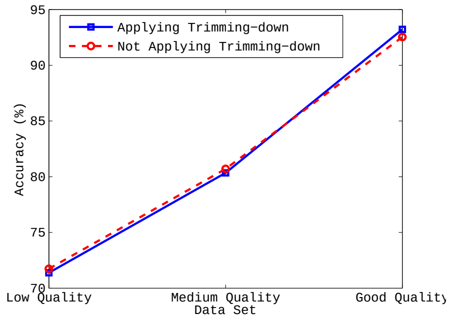
Figure 8 illustrates the performance with and without applying the trimming-down strategy. Observed from Fig. 8 (a), after applying the trimming-down strategy, the average recognition time is nearly reduced by half in the three data sets. Furthermore, the recognition rate is nearly unchanged as seen in Fig. 8 (b). This figure clearly demonstrates how the trimming-down strategy improves the performance of the system.

4 Conclusion

In this paper, we have reported a new Handwritten Chinese Address Recognition system. We propose a three-stage structure, i.e., a coarse recognition-fine recognition-verification framework to attack this difficult recognition task. Experimental results on three real world data sets demonstrate that our new system outperforms the previous methods.



(a)



(b)

Figure 8: Analysis with and without applying the trimming-down strategy

References

[1] Q. Fu, X. Ding, Y. Jiang, C. Liu, and X. Ren. A hidden markov model based segmentation and recognition algorithm for chinese handwritten address character strings. In *Proceedings of the Eigth International Conference on Document Analysis and Recognition (ICDAR-2005)*, pages 590–594, 2005.

[2] Z. Han, C. P. Liu, and X. C. Yin. A two-stage handwritten character segmentation approach in mail address recognition. In *Proceedings of the Eigth International Conference on Document Analysis and Recognition (ICDAR-2005)*, pages 111–115, 2005.

[3] Y. Hotta, H. Takebe, and S. Naoi. Holistic word recognition based on synthesis of character features. In *Proceedings of the Fourth IAPR International Workshop on Document Analysis Systems (DAS-2000)*, pages 313–324, 2000.

[4] S. Naoi, M. Suwa, and Y. Hotta. Recognition of handwritten japanese addresses based on key character extraction and holistic word matching. In *Proceedings of the Third IAPR International Workshop on Document Analysis Systems (DAS-1998)*, pages 149–152, 1998.

[5] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal Molec. Biol*, 48:443–453, 1970.

[6] C. Wang, Y. Hotta, M. Suwa, and S. Naoi. Handwritten chinese address recognition. In *Proceedings of the 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR-9)*, 2004.