Application of Bi-gram Driven Chinese Handwritten Character Segmentation for an Address Reading System

Yan Jiang[1], Xiaoqing Ding[1], Qiang Fu[1], and Zheng Ren[2]

[1] Department of Electronic Engineering, Tsinghua University, Beijing, China, 100084
{jyan, dxq, fuq}@ocrserv.ee.tsinghua.edu.cn
[2] Siemens AG, D-78467 Konstanz, Germany
zheng.ren@siemens.com

Abstract. In this paper, we describe a bi-gram driven method for automatic reading of Chinese handwritten mails. In destination address block (DAB) location, text lines are first extracted by connected components analysis. Each candidate line is segmented and recognized by our holistic method, which incorporates mail layout features, recognition confidence and context cost. All these are also taken into consideration to identify the DABs from the candidate text lines. Based on them, street address line and organization name line are determined. At last step, edit distance based string matching is performed against given databases. We also discuss the pretreatment to deal with Chinese address databases consisted of a large amount of vocabularies in order to generate keywords for fast indexing during matching. Detailed experiment results on handwritten mail samples are given in the last section.

1 Introduction

OCR (optical character recognition) has been applied in postal automation since 1970s. The first generation of systems is based on postcode recognition, which could only recognize the digits in a specific region on an envelope. Nowadays, it is becoming more and more deficient in satisfying the requirements for faster reading and more detailed treatment of rapidly increasing mails. With the development of pattern recognition, it is now feasible to automatically process all the information on an envelope to boost the recognition performance. There have been various literatures in this area, covering different languages, such as English ([1]), Japanese ([4]) and so on.

Generally speaking, there is no obvious gap between two adjacent characters in a Chinese address line, so a text line should be first segmented into characters for recognition, which has long been an obstacle in Chinese address reading. Although there has been much work considering layout and recognition information in segmentation, it still remains insufficient for some special real world applications like mail reading where high performance in segmentation is needed, since segmentation errors are always incorrigible for post-processing and address

H. Bunke and A.L. Spitz (Eds.): DAS 2006, LNCS 3872, pp. 220–231, 2006.

match. In our proposed method, we incorporate layout cost, recognition cost and contextual cost together based on bi-gram model to achieve a good performance both in segmentation and recognition. Furthermore, we could see that recognition cost and contextual cost are also very important in DAB location, comparing with layout features, they are more discriminating.

Up to now, there are still few efforts considering the work to search and match the Chinese address items in a very large database containing hundreds of thousands of items and our work also focus on this area. We propose a pretreatment method for address database to generate keywords for both searching and matching. The samples which cannot be uniquely identified would be rejected in the keywords searching stage for the consideration of efficiency.

The remainder of this paper is organized as follows: we briefly review bi-gram model based OCR post-processing in Sect. 2; in Sect. 3, we have a short review on Xue's method in extracting text lines from envelopes; in Sect. 4, we introduce our holistic method for address segmentation and apply the evaluated score in DAB location; in Sect. 5, we introduce our basic idea for the pretreatment of the address database and review the definition of edit distance which is used in our string comparison; experiments results are given in Sect. 6.

2 Bi-gram Model

N-gram model has been introduced to natural language processing (NLP) for a long time and it is applied in speech recognition and OCR post-processing. In OCR, a typical character classifier generates several hypotheses for an input image, and the first candidate is not ensured to be correct. Post-processing techniques are studied to select the most likely recognized strings from the candidate characters, the process is formularised as follows:

$$\mathrm{c}_{1,k_1^*}, \mathrm{c}_{2,k_2^*}, \ldots, \mathrm{c}_{T,k_T^*} = \underset{1\le k_t\le M, 1\le t\le T}{\arg\max} P(\mathrm{c}_{1,k_1}, \mathrm{c}_{2,k_2}, \ldots, \mathrm{c}_{T,k_T} | \mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_T) \quad (1)$$

Where $\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_T$ are a series of character images, x_i denotes the i-th character image. The classifier gives M candidate characters for each input character image, which are denoted by $\mathrm{c}_{i,k_i} (1 \le k_i \le M)$. The contextual post-processing method select characters from candidate sets to form the most likely string $\mathrm{c}_{1,k_1^*}, \mathrm{c}_{2,k_2^*}, \ldots, \mathrm{c}_{T,k_T^*}$ by maximizing the posterior probability.

By Bayesian formula,

$$\begin{aligned} &P(\mathrm{c}_{1,k_1}, \mathrm{c}_{2,k_2}, \ldots, \mathrm{c}_{T,k_T} | \mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_T) \\ &= \frac{P(\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_T | \mathrm{c}_{1,k_1}, \mathrm{c}_{2,k_2}, \ldots, \mathrm{c}_{T,k_T}) P(\mathrm{c}_{1,k_1}, \mathrm{c}_{2,k_2}, \ldots, \mathrm{c}_{T,k_T})}{P(\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_T)} \end{aligned} \quad (2)$$

Bi-gram model only considers the transition probability between two characters, so the probability $P(\mathrm{c}_{1,k_1}, \mathrm{c}_{2,k_2}, \ldots, \mathrm{c}_{T,k_T})$ is simplified as Eq.3.

$$P(\mathrm{c}_{1,k_1}, \mathrm{c}_{2,k_2}, \ldots, \mathrm{c}_{T,k_T}) = P(\mathrm{c}_{1,k_1}) \times \prod_{i=1}^{T-1} P(\mathrm{c}_{i+1,k_{i+1}} | \mathrm{c}_{i,k_i}) \quad (3)$$

Assuming that the current recognition behavior is independent of the previous decisions in the classifier ([3]),

$$\begin{aligned} P(\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_T | \mathrm{c}_{1,k_1}, \mathrm{c}_{2,k_2}, \ldots, \mathrm{c}_{T,k_T}) &= \prod_{i=1}^{T} P(\mathrm{x}_i | \mathrm{c}_{i,k_i}) \\ &= \prod_{i=1}^{T} P(\mathrm{x}_i) \times \prod_{i=1}^{T} \frac{P(\mathrm{c}_{i,k_i} | \mathrm{x}_i)}{P(\mathrm{c}_{i,k_i})} \end{aligned} \tag{4}$$

$P(\mathrm{c}_{i,k_i})$ is the prior probability determined by the classifier, which could be seen as uniformly distributed ([3]). $P(\mathrm{c}_{i,k_i}|\mathrm{x}_i)$ could be estimated from recognition distances given by the classifier ([5]).

Combining the above equations, we could simplify the maximization process of posterior probability to the following, in which, Viterbi algorithm is applied ([3]).

$$\mathrm{c}_{1,k_1^*}, \mathrm{c}_{2,k_2^*}, \ldots, \mathrm{c}_{T,k_T^*} = \underset{1 \le k_t \le M, 1 \le t \le T}{\arg\max} P(\mathrm{c}_{1,k_1}) \prod_{i=2}^{T} P(\mathrm{c}_{i,k_i} | \mathrm{c}_{i-1,k_{i-1}}) \prod_{i=1}^{T} P(\mathrm{c}_{i,k_i} | \mathrm{x}_i) \tag{5}$$

We will see the maximum of the right side of Eq.5 is important for both character segmentation and DAB location.

3 Text Line Extraction

The writing style of Chinese envelopes are much different from the western mails. Receiver's information is written in the upper part of the envelope, while sender's information is often written in the lower part. The receiver's name is almost in the center of the image. (Fig. 1) The basic framework of our work is illustrated in

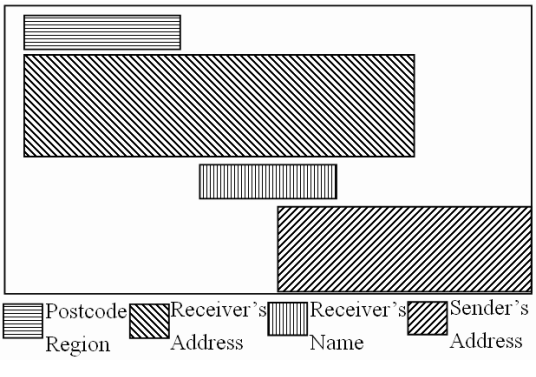


Fig. 1. A typical Chinese envelope layout

Fig.2. In text lines extraction module, connected components are extracted from binary image after preprocessing steps. Each connected component is regarded as a block, which is then categorized into four kinds: noise, text, graphic and image. This classification process is mainly based on the layout features of the connected component, detailed algorithm could be found in [7]. Only text blocks are reserved for the step of text lines extraction. Text lines are generated by

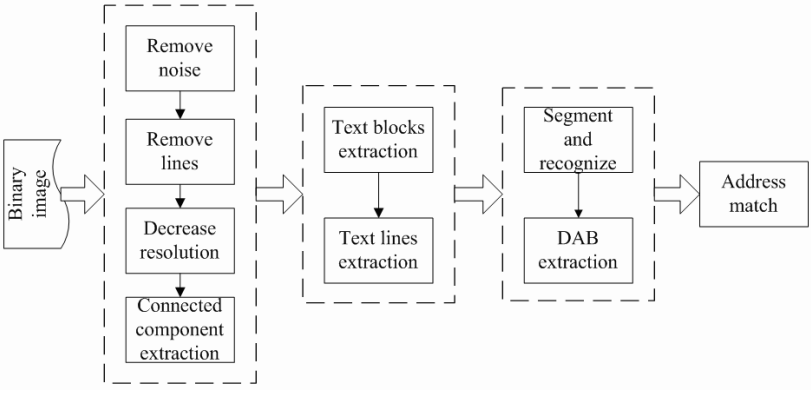


Fig. 2. Framework of our proposed system

merging the text blocks. Layout constraints are taken into account to identify which line a text block belongs to. Then, we merge all the text blocks belonging to the same line together as our extracted candidates, but there maybe more text lines than we expected. We must identify which text line is useful for us from the text line candidates.

For example, six text lines are extracted in Fig.3 (in rectangles). The first text line contains postcode, the fourth text line contains receiver's name and the fifth and six text lines contain sender's information. They are useless for us since we don't intend to process such information. Only the second and the third text line should be identified as DAB. In Xue's work, he presents a bottom-up strategy considering some special characteristics of handwritten Chinese envelopes. But his method may cause some errors simply by considering layout features. In the proposed method, DAB location is not performed directly after text line extraction, instead, all the extracted text lines are sent to the segmentation and recognition module. We will evaluate each line not only by its arrangement features but also by its recognition and contextual cost.

4 Character Segmentation and DAB Location

Chinese characters have very complex structures, and addresses are often written in cursive style. Conventional segmentation methods are based on structural

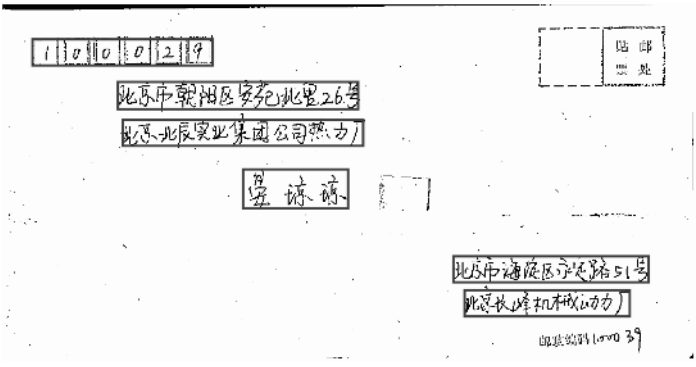


Fig. 3. Extracted text lines by Xue's method

and layout information, but they are not always stable in dealing with Chinese scripts. Some people introduced recognition cost in Chinese segmentation, but it is not enough. Left-right structured characters occupy a large part in Chinese. Both parts of them are valid characters and will own high recognition confidence respectively.

4.1 Over-Segmentation

The general process of character segmentation for Chinese involves an over-segmentation stage at first. Each line image is over-segmented into a series of radicals by structural method. A radical is one part of a character and all radicals are reunited to form character images. An efficient over-segmentation technique could remove overlapping from scripts. We adopt the algorithm proposed by Xue, which cited the work of Tseng and Gao. Only layout information is considered in over-segmentation. Fig.4(b) shows the result of Fig. 4(a) after over-segmentation, each radical is bounded in a rectangle.

4.2 Merge Radicals

For over-segmented radicals, we establish a segmentation graph (Fig.4(c)), the edge from one node to another represents a combination image of some certain radicals and the edge is assigned a cost for the combination. For example, if the over-segmented radicals are denoted by $s_1, s_2, \ldots, s_l$, then the cost of the edge from Node_i to Node_{i+k} is assigned as the cost to combine $s_i, s_{i+1}, \ldots, s_{i+k-1}$.

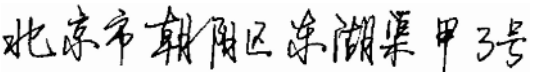
The widely adopted method to merge radicals is to find the shortest path in the segmentation graph from Node_1 to Node_{l+1}. Different strategies have been proposed to evaluate the edges' costs, which can be summarized into three main kinds: structural analysis based evaluation, recognition based evaluation and contextual evaluation.

In our method, we first evaluate each edge by Xue's cost function ([7]). Then we apply K-shortest algorithm ([2]) in the segmentation graph according to the

layout cost and evaluate each path by our proposed function L. The path with highest score will be selected as our segmentation decision. At the same time, the OCR result of this line image will be given. Our cost function L is formalized as follows, which incorporates layout cost, recognition cost and contextual cost together.

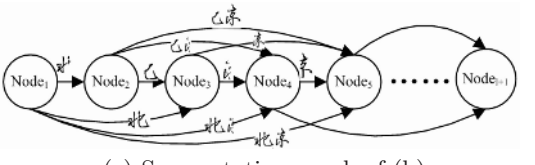
$$L(path) = \frac{1}{n}[\sum_{i=1}^{n} \log P(\mathrm{c}_{i,k_i}|\mathrm{x}_i) + \log P(\mathrm{c}_{1,k_i}) + \sum_{i=2}^{n} \log P(\mathrm{c}_{i,k_i}|\mathrm{c}_{i-1,k_{i-1}})] \\ + \frac{\lambda}{n}[\log P(\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_n|\mathrm{s}_1, \mathrm{s}_2, \ldots, \mathrm{s}_l)] \tag{6}$$

Where $\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_n$ are the merged images according to *path*, and the third item on the right hand $P(\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_n|\mathrm{s}_1, \mathrm{s}_2, \ldots, \mathrm{s}_l)$ is the confidence to merge $\mathrm{s}_1, \mathrm{s}_2, \ldots, \mathrm{s}_l$ into $\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_n$ according to their layout distribution, the item is estimated from the evaluated layout cost of *path*. λ is a weight factor that is estimated by experiments. The first and the second items are maximized by Viterbi algorithm.



(a) A Chinese address string image

(b) After over-segmentation (each radical is bounded in a rectangle)



(c) Segmentation graph of (b)

Fig. 4. Over-segmentation and merge of radicals

We compare our bi-gram based segmentation result with a previous method proposed by Xue ([7]). Table 1 shows the result containing two content rows which are corresponding to two different methods. The first and the second columns give the number of correctly segmented characters and lines respectively; the third and the fourth columns tell the percentage of the correctly segmented characters and lines. Experiments is carried out on 233 line images containing 3,014 characters which are extracted from Chinese handwritten envelopes.

Table 1. Bi-gram driven segmentation results

	Correct segmentations of characters	Correct segmentations of lines	Ratio of correct segmentations of characters (%)	Ratio of correct segmentations of lines (%)
Xue's method	2,492	55	82.7	23.6
Our method	2,787	144	92.4	61.8

4.3 Recognition and Contextual Information Based Address Line Determination

In Xue's bottom-up method, he uses an empirical evaluation function and identifies the text line with the highest score to be the name line. Then the lines between the first line and the name line are selected as DAB. But this method is not stable in practice, since there maybe some unexpected errors to locate the name line simply by layout analysis. Furthermore, we cannot tell which line contains the geographic location and which line contains organization name by layout feature, since text lines of geographic location and text lines of organization name have no obvious differences.

A more effective method should take both character recognition result and contextual information into account. The proposed method take the context-recognition cost H as the criterion, H is defined as the maximum of the sum of the first item and the second item in Eq.6.

$$H = \max_{1 \le k_1, k_2, \ldots, k_n \le M} \frac{1}{n} [\sum_{i=1}^{n} \log P(\mathrm{c}_{i,k_i} | \mathrm{x}_i) + \log P(\mathrm{c}_{1,k_1}) + \sum_{i=2}^{n} \log P(\mathrm{c}_{i,k_i} | \mathrm{c}_{i-1,k_{i-1}})] \tag{7}$$

In fact, bi-gram model trained on different corpus could reflect different contextual characteristics. Inspired by this idea, we trained our bi-gram model on geographic addresses and organization names respectively. Each segmented line image is evaluated by the above two bi-gram models and get H_{g} (geographic address based bi-gram model) and H_{o} (organization name based bi-gram model) respectively. If a sample is consistent with the domain of the bi-gram model, the value of H will be higher than that is calculated on the corpus which doesn't belong to this domain. Noticing this fact, we could summarize the following rules: (1) if $H_{\mathrm{g}} > H_{\mathrm{o}}$ and $H_{\mathrm{g}} > T$, then we could judge this line image to be a geographic address line; (2) if $H_{\mathrm{o}} > H_{\mathrm{g}}$ and $H_{\mathrm{o}} > T$, then we could judge this line image to be a organization name line. T is a predefined threshold to control the lines to be selected as DAB, since postcode line and name line always have distinctly low H value.

5 Address Matching

Address interpretation is often considered as a string match process. We choose the most suitable address item from a large vocabulary dictionary according to

our OCR results. Moreover, we are always required to search and match with acceptable expenditure of time. So the key problem is how to effectively select as small address candidates as possible to include the correct one in a large database.

In our method, keywords are generated for each address item in the pretreatment stage and we establish a lookup table for fast keywords index. For a line image, after the recognition process, we look for the keywords in the recognized string first and select the corresponding address items as candidates (Figure 5).

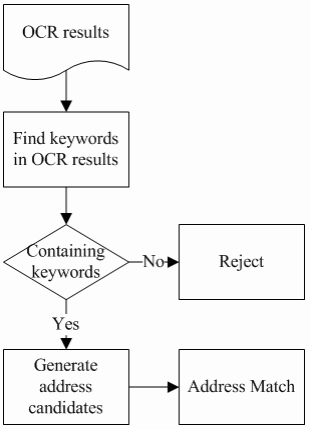


Fig. 5. Flowchart of address searching and matching

5.1 Keyword Generation

For geographic location items, there are twenty items with the same street name averagely, that is, these items are only different in street number. For organization names, it is also a common situation that some organization names are only different in one or two characters. Ambiguous items make it tough for us to identify an item against many similar ones. Furthermore, recognized characters cannot be assured to be entirely correct because some unforeseen errors may be brought by DAB location, character segmentation or recognition. Two principles are essential for keyword generation: (1) the extracted keywords should be easy for searching; (2) OCR result should be rejected if no keywords could be found.

For a Chinese address item, words are piled without gaps between each other. Word segmentation is first applied to split an item into words first. For example, a organization name "北京希盛技术开发有限公司" is segmented into five words: "北京" "希盛" "技术" "开发" "有限公司". "北京" is the city name, "希盛" is the name of the company, "技术" means "technology", "开发" means "development" and "有限公司" means "limited company". It is easy to see that the words "北

京", "技术", "开发" and "有限公司" are helpless to uniquely identify the item. Only the word "希盛" is extracted as the keyword for this name. We give a detailed description for this idea. For an input item, we exam all the segmented words. If a Chinese word contains more than two characters, we exhaust all the two characters combinations, that is, for the word "建国门", we decompose it into "建国" and "国门". For a given string, after word segmentation, we extract a series of two-character words $w_1, w_2, \ldots, w_k$ and their corresponding number of occurrence in address data $N_1, N_2, \ldots, N_k$. The weight for each word is calculated as $\text{Weight}(w_i) = \frac{N_i}{\sum_{j=1}^{j=k} N_j}$ and only the words whose weights are less than a predefined threshold are extracted as keywords for this item.

5.2 Edit Distance

Edit distance is proposed by Levenshtein ([6]) to compare two strings, which is defined as the minimum number of the basic edit operations to transform one string into another. Levenshtein concluded three basic edit operations: deletion, insertion and substitution. Edit distance is calculated by dynamic programming.

For string $a_1a_2, \ldots, a_P$ and $b_1b_2, \ldots, b_Q$, we build a $(P+1) \times (Q+1)$ matrix $\{L_{p,q}\}_{0 \le p \le P, 0 \le q \le Q}$, and set $L_{p,0} = p$ for $0 \le p \le P$, $L_{0,q} = q$ for $0 \le q \le Q$. Then $L_{p,q}$ is calculated by Equation (8), and $L_{P,Q}$ is the edit distance between $a_1, a_2, \ldots, a_P$ and $b_1, b_2, \ldots, b_Q$.

$$L_{p,q} = \begin{cases} L_{p-1,q-1} & \text{if} \quad a_p = b_q \ ; \\ 1 + \min\left(L_{p-1,q-1}, L_{p-1,q}, L_{p,q-1}\right) & \text{otherwise} \ . \end{cases} \tag{8}$$

5.3 Geographic Location Items Match

In fact, there are many valid variations for geographic location items. Some parts in a geographic address are not necessary and could be omitted. We try to reflect this feature according to the geographic address model shown below. A common

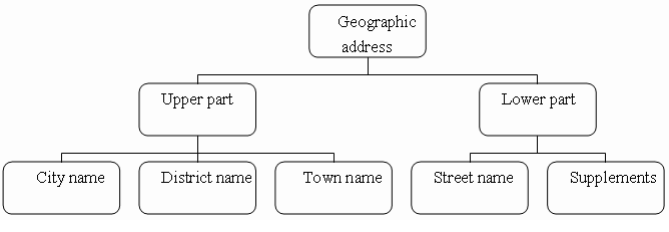


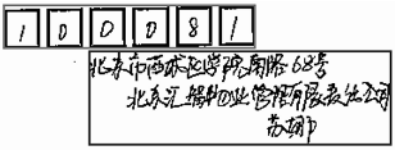
Fig. 6. General structure of Chinese geographic address

Chinese geographic address is composed of five parts and the district name part and town name part are often omitted in general. In addition, there are some suffixes which are written directly behind the address components, such as "市", "县", "区", "镇","乡" and so on, could be omitted too.

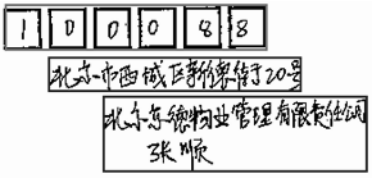
6 Experiments

We collect about 1000 handwritten samples written in different styles. The handwritten character classifier is developed by the department of electronic engineering, Tsinghua University. We collected two address databases: one is full address database, which contains more than 180,000 items in Beijing city (Database I) and the other is street name database which contains more than 7,000 street names in Beijing city (Database II). Each item in Database I is composed of three parts: postcode, geographic location and organization name.

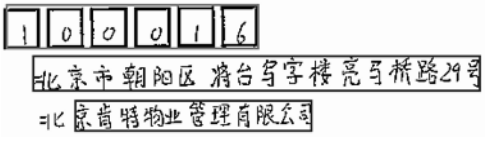
We test DAB location and line discrimination performance. 315 envelopes are taken for this experiment, which contains 630 DAB lines that should be extracted. We will divide this process into two steps: one is text line location and the other is DAB location. The method for the first step is reviewed in Section 3. DAB location step tries to find the text lines of interest. Our contextual-recognition method is compared with Xue's bottom-up strategy.



(a) Fail to extract the text lines containing address information



(b) Some address text lines are missing



(c) Some characters in a text line are missing

Fig. 7. Text line location error classification

Table 2. Text line location errors

Error of Type (a)	Error of Type (b)	Error of Type (c)
16	7	34

In fact, some errors occur at text line location step, we classify those errors into three basic types as shown in Fig.7 and conclude these errors in Table 2.

We abandon the text lines, which cannot be correctly extracted on the envelopes and test our DAB location and line discrimination method on the remain-

ing text lines. There remains 591 text lines that should be identified as either geographic location or organization name from all the extracted text lines. We compare the number of two kinds of errors in DAB location in Table 3. Class A error indicates that a text line which should be identified as DAB is missed; Class B error indicates that a text line which should not be identified as DAB is wrongly accepted. Our proposed line discrimination method is tested in the fourth column of Table 3.

Table 3. DAB location and line discrimination

	Class A errors	Class B errors	Correct rate of discrimination
Xue's method	39	37	—
Proposed method	7	4	97.3%

We have two experiments on Database I and II respectively. In fact, after DAB location, we could tell whether a text line contains geographic information or organization information, so we could compare a recognized string with the corresponding items in Database I. The text line of geographic address and the text line of organization name on the same envelope are matched independently and the results are shown in the first and second column of Table 4. If we combine the matching result of geographic address and organization name on an envelope together, it will reduce the matching errors as shown in the third column of Table 4.

We can find that there are more errors in geographic address match in Database I. Most of these errors are only caused by street number comparison. We may summarize the following problems: (1) many geographic addresses are only different in their street number; (2) there are many confusing characters between Arabic digits, English alphabets and Chinese characters, such as: "13" vs. "B", "3" vs. "ℨ" and so on, it is even difficult for a people to tell whether a character image is a Chinese character, an English character or an Arabic digit without a look at the whole text line; (3) there are no strict contextual constraints to the digits. In another experiment, we compare the extracted geographic address line with the items in Database II. All the items in this database are varied according to the address model and street number comparision is neglected (result is shown in the fourth column of Table 4).

Table 4. Experiments of address match

	Geographic address match (Database I)	Organization name match (Database I)	After fusion (Database I)	Geographic address match (Database II)
Recognition rate	72.8%	84.8%	90.3%	89.8%

7 Conclusions

In this paper, contextual information is taken into consideration for character segmentation by bi-gram model trained on the address corpus which gives better result for cursive handwritten character segmentation. We also see that contextual-recognition cost is also important in DAB location, which makes it possible for precise address line selection and discrimination. We also have discussed two main rules for keywords generation for the purpose of faster search and match.

There still remains some challenging work. First of all, text line location needs to be adapted to diverse writing styles. In matching stage, edit distance with uniform weight for each operation is not suitable for real word application, since the characters in an item are not of same importance. If we could involve address match in character segmentation and recognition stage, it is sure that we could improve our matching of Arabic digits. Additionally, postcode information may help to boost the speed for searching address candidates and improve the matching accuracy to a certain extent. Furthermore, recognition confidence, bi-gram model and string matching could be somewhat combined to find a global optimal solution in the framework of probability theory, which would put our algorithm onto a more stable theoretical foundation.

Acknowledgements. This work has been funded by Siemens AG under contract number 20030829 - 24022SI202.

References

1. El Yacoubi A., Bertille, J.M., Gilloux, M.: Conjoined location and recognition of street names within a postal address delivery line. Proc. 5th International Conference on Document Analysis and Recognition (1995) 1024–1027
2. Jimenez, V.M., Marzal, A.: Computing the K shortest paths: A new algorithm and an experimental comparison. Proc. of the Third International Workshop on Algorithm Engineering, London, July, 1999. LNCS vol. 1668. Springer 15–29
3. Li, Y.X., Ding, X.Q., Tan, C.L., Liu, C.S.: Contextual Post-processing based on the confusion matrix in offline handwritten Chinese script recognition. Pattern Recognition **37**(9) (2004) 1901–1912
4. Liu, C.L., Koga, M., Fujisawa, H.: Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading. IEEE Trans. PAMI **24**(11), (2002) 1425–1437
5. Liu, C.L., Masaki, N.: Precise Candidate Selection for large character set recognition by confidence evaluation. IEEE Trans. PAMI **22**(6), (2000) 36–642
6. Levenshtein, V.I.: Binary codes capable of correcting insertions and reversals. Soviet Physics Doklady **10**(8), (1966) 707–710
7. Xue, J.L., Ding, X.Q.: Location and interpretation of destination addresses on handwritten Chinese envelopes. Pattern Recognition Letters **22**(6), (2001) 639–656