PERGAMON

Pattern Recognition 34 (2001) 1741–1750

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Field data extraction for form document processing using a gravitation-based algorithm

Jiun-Lin Chen, Hsi-Jian Lee*

Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu, Taiwan 30050, ROC

Received 6 June 1999; received in revised form 24 April 2000; accepted 20 June 2000

Abstract

This paper presents a novel approach to grouping Chinese handwritten field data filled in form documents using a gravitation-based algorithm. An algorithm is developed to extract handwritten field data which may be written out of form fields. First, form lines are extracted and removed from input form images. Connected-components are then detected from remaining data, and the gravitation for each connected-component is computed by using the black pixel counts as their mass. Next, we move connected-components according to their gravitation. As generally known, filled-in data have the locality property, i.e., data of the same field are normally written in a local area consecutively. Therefore, the relationship of these connected-components can be determined by this property. Repeatedly moving these connected-components according to their neighbor components allows us to determine which connected-components should be extracted for a particular field. Experimental results demonstrate the effectiveness of the proposed method in grouping field data. © 2001 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Form document processing; Field-data grouping; Gravitation-based algorithm; Connected-component; Locality property

1. Introduction

The main purpose of a form document processing system is to collect information filled in by individuals for further applications. Retrieval and recognition of filled-in data are the essential functions of a form document processing system. However, automatically extracting filled-in data for each field is difficult since handwritten data may be written out of their fields, as shown in Fig. 1. According to the office and medical form documents collected in our experiments, this is the situation 17.63 and 30.04% of data fields and filled data fields, respectively. To overcome this problem, we present a novel approach for extracting handwritten filled-in data of form documents in this paper.

Several form document processing systems have been developed in recent years [1–7]. Casey et al. [1] developed an intelligent form-processing system with filled-in data acquisition from a form image. Taylor et al. [2] proposed a system for extracting data from the preprinted forms. They located pixels within the interior of the field and searched slightly above and below the field for characters which extend outside the field. Yu et al. [3] proposed a system for form dropout. Although their system can separate input characters and form frames, they did not extract any information between input characters and fields. Thus, their method cannot determine the fields that characters are filled into. Fan et al. [4,5] presented a clustering-based method for extracting characters from form documents. In their method, feature points were clustered to distinguish characters from form documents. Fletcher et al. [7] presented a connected-component based method for separating text strings from mixed text/graphics images. To extract text strings, their algorithm groups strings into words and phrases logically, groups collinear components together and then performs area/ratio filtering.

* Corresponding author.: Tel.: + 886-3-5731802; fax: + 886-3-5721500.

E-mail addresses: chenjl@csie.nctu.edu.tw (J.-L. Chen), hjlee@hjlee.csie.nctu.edu.tw (H.-J. Lee).

PII: S0031-3203(00)00115-1

Extracting filled-in field data is an important aspect of a complete form document processing system. Several techniques have been proposed including connected-component analysis [7], searching outside the field [2] and feature point clustering [4,5]. Connected-component analysis can be used to distinguish characters from mixed text/graphics images; however, it cannot identify the field that the characters belong to. Searching outside the field [2], although a preferred means of handling characters that extend out of the field, has two problems: (1) determining the extent to search is difficult and (2) determining which field should be extended is difficult as well. Feature point clustering, although efficiently distinguish characters from form structure, cannot determine which field characters belong to, either. The field data shown in Fig. 1(a) can be extracted correctly by using connected-component analysis, but not feature point clustering. However, neither field data in Fig. 1(b) nor in Fig. 1(c) can be extracted successfully using these techniques.

In this paper, we present a novel gravitation-based algorithm for grouping and extracting filled-in field data of form documents. According to the form documents collected herein, we found that filled-in data have the locality property, meaning that filled-in data of the same field are usually written in a local area consecutively. By considering each connected component of the filled-in data as a planet in the universe, the gravitation among them causes the consecutive data to gravitate towards each another. Thus, such a gravitation model can be applied to group field data since data of the same field are usually closer than those of different fields. Fig. 2 presents a form document with filled-in data, the telephone number, written out of fields.

The proposed method contains three major parts: preprocessing, gravitation processing and postprocessing. In the following sections, we discuss the details of the proposed method.

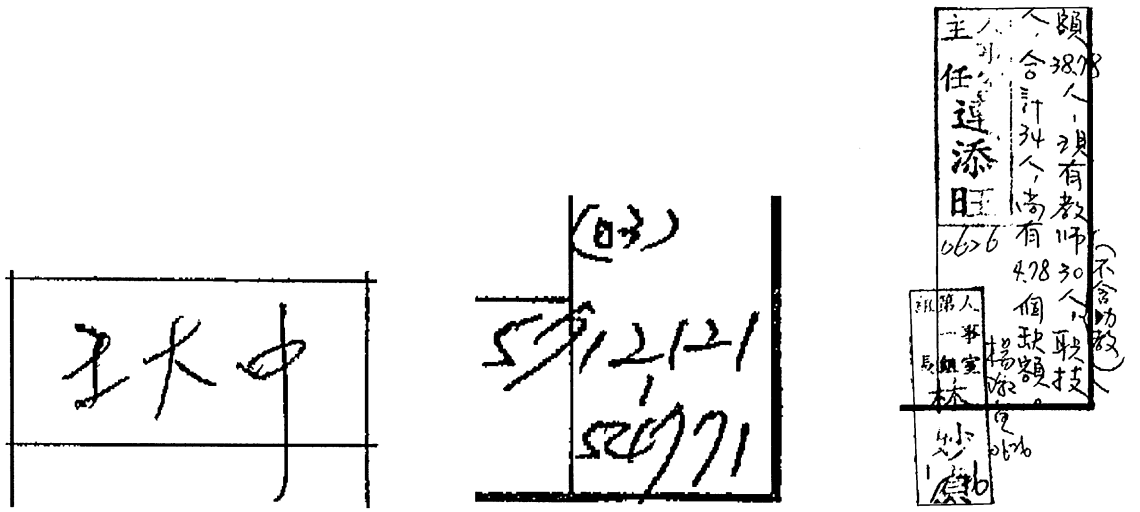
2. Preprocessing

In our proposed method, an input form document is first scanned with 300 dpi (dots per inch) as a gray-scale image. Next, noise removal and binarization are performed by using the method of Niblack [8] for input form images. Then, the method of Chen and Lee [9] is used to extract form fields and remove all extracted form lines from the form image. Finally, connected components detection is performed to locate the filled-in data.

2.1. Field extraction and form line removing

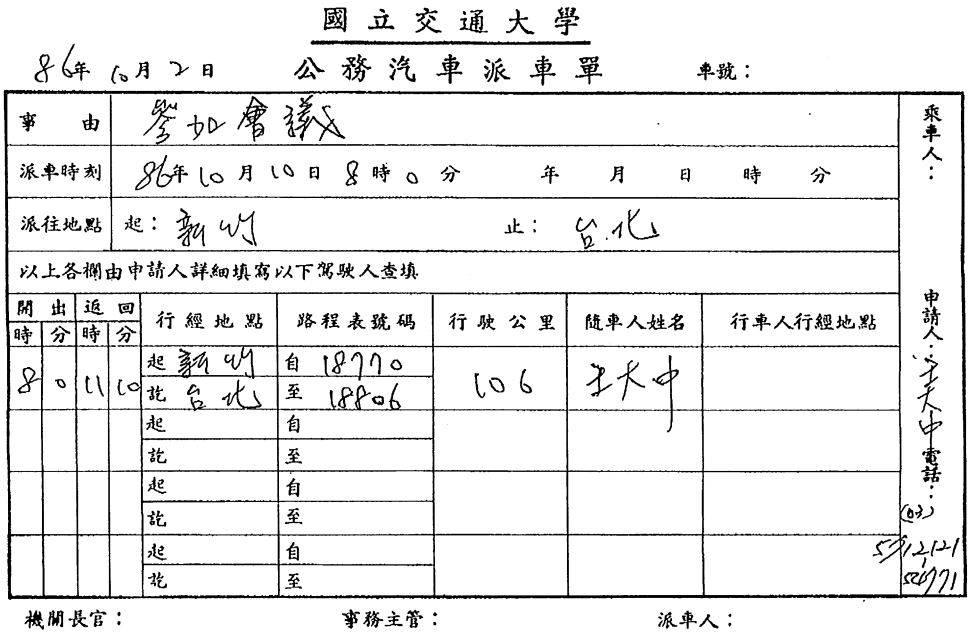
To extract form lines and fields, this work adopts the strip projection method [9], which is efficient in terms of extracting the form structure. Owing to that meaningful filled-in data that usually appeared around a form, we add four additional virtual fields (Fig. 3) into the extracted fields list. After locating the form lines with their two end-points, all form lines can be removed to focus on the form data.

According to the line extraction method employed herein, we obtain two end-points for each extracted line on the



(a) (b) (c)

Fig. 1. Three different cases of data written out of a field. (a) A stroke of a character extends outside the field. (b) An entire character is written outside the field. (c) Several characters are written outside the field.


國立交通大學

86年 10月 7日 公務汽車派車單 車號：

事由	參加會議						乘車人：
派車時刻	86年10月10日 8時0分 年 月 日 時 分						
派往地點	起：新竹 止：台北						
以上各欄由申請人詳細填寫以下駕駛人查填							
開出 時	開出 分	返回 時	返回 分	行經地點	路程表號碼	行駛公里	隨車人姓名	行車人行經地點	申請人：王大中 電話：(03) 5712121 ext 54771
8	0	11	10	起 新竹	自 18770	106	王大中		
				訖 台北	至 18806				
				起	自				
				訖	至				
				起	自				
				訖	至				
				起	自				
				訖	至				

機關長官： 事務主管： 派車人：

Fig. 2. A form document with filled-in data, the phone number, written outside the field.

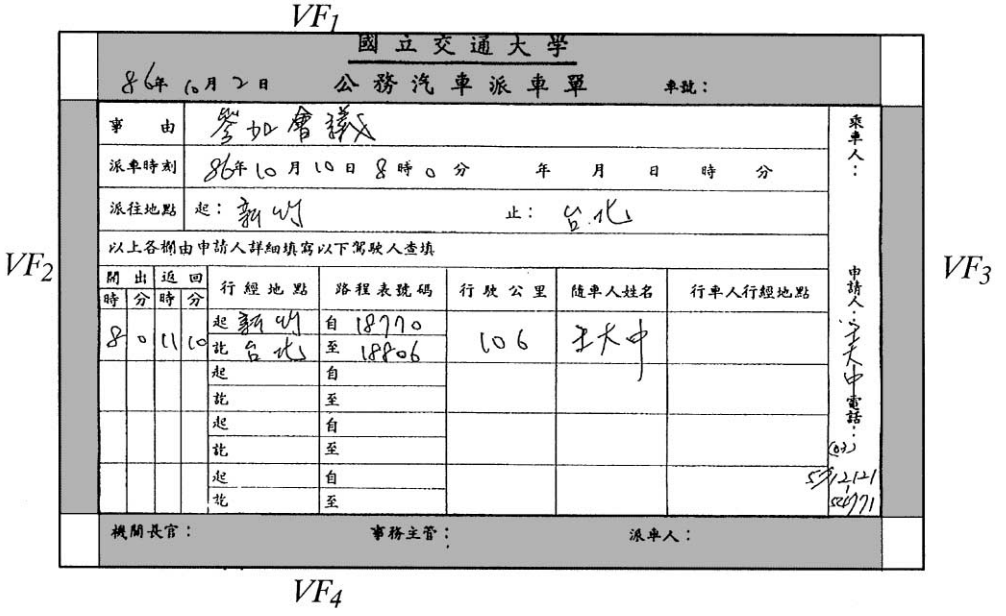


Fig. 3. Four additional virtual fields, labeled as gray regions, for a form document.

original form image. Thus, a 3×3 window shown in Fig. 4 can be used to trace through a given form line to remove it. The steps to trace a horizontal form line are as follows:

Step 1. Set P as the left end-point.

Step 2. If P_1 is a black pixel, move P to P_1. Repeat step 2.

Step 3. If P_2 is a black pixel, move P to P_2, and go to step 2.

Step 4. If P_8 is a black pixel, move P to P_8, and go to step 2.

Step 5. If there is a black pixel P_i on the right side of P and also in the same row as P, and *Distance* $(P_i, P) <$

20 *pixels*. Then, move P to P_i and go to step 2.

Step 6. Stop. {We meet the right end of this line.}

Removing a horizontal form line, we go through the line twice. The first pass is used to estimate the line width. At each point, we trace up and down to locate the top and bottom end points by a procedure similar to line tracing. The distance between the top and bottom end points is recorded as the line width at that point. After running through the whole line, the width of this line is defined as the medium value of all width values recorded so far. The estimated line width is used as the line width threshold in the next pass.

Next, the horizontal line is removed in the second pass. If the width at a point P is larger than the width threshold, the vertical line segment at P is preserved since it is assumed herein to be on a long vertical line or on a character. Otherwise, the vertical line segment at P is removed. Preserving the vertical line segments of these points that have larger widths than the line width threshold allows us to make other data remain complete if they touch with this horizontal line which is being removed.

The procedure to remove vertical form lines resembles the above procedure.

P_6	P_7	P_8
P_5	P	P_1
P_4	P_3	P_2

Fig. 4. The 3×3 windows used in line removal.

2.2. Connected-component detection

After removing all form lines, connected-component detection is performed. The bounding box, black pixels and the number of black pixels of each component are recorded. Overlapped connected components are merged together, and components that are too small are eliminated.

In the gravitation-based algorithm proposed herein, we operate on connected-components rather than on characters or black pixels. Fig. 5 presents the result of form lines removal and connected-component detection from Fig. 2. In the next section, we explain the method of grouping connected-components of the same field together.

3. Gravitation-based algorithm

Having represented form data as connected-components, the connected-components can then be moved to their corresponding field with a gravitation-based

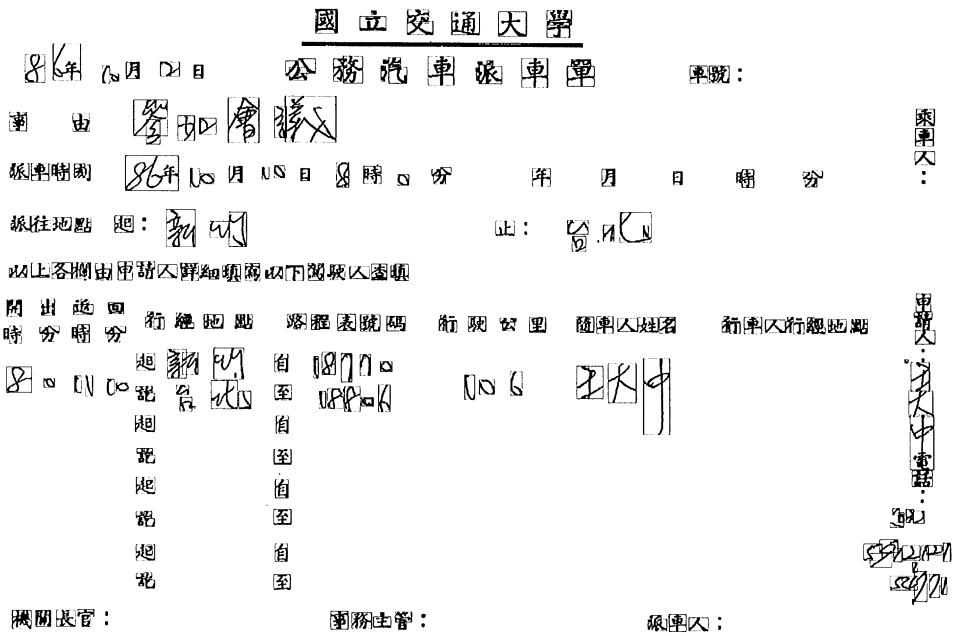


Fig. 5. Result of line removing and connected-component detection.

algorithm. According to physics theory, the gravitation between objects depends on the masses of objects and their distance between each other. Herein, the black pixel count of each connected component is used as its mass. Calculation is also performed of the gravitation for all connected-components as they impact one another according to their masses and the distances among them. To reduce the processing time, only those components whose center of mass are not located inside the center area of a field are moved. The gray area shown in Fig. 6 denotes a partial center area of a field, and the components '1' and ')' located in it. The rest of the components are moved according to their gravitation. Once a connected-component is moved into the center area of a form field, we can determine this component belongs to that field. Repeating the above steps, we can move those connected-components which are near field boundaries into the fields they belong to. When a stop criterion is satisfied, the iteration of the gravitation process is stopped. Fig. 7 shows the flow chart of the gravitation-based algorithm, and the details are as follows.

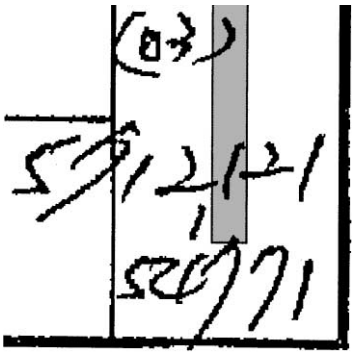


Fig. 6. Partial center area of a field.

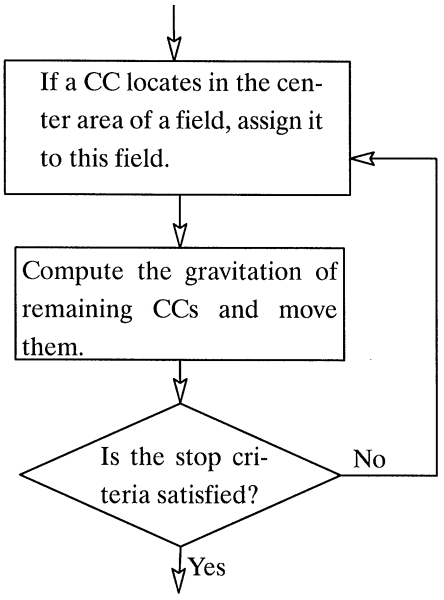


Fig. 7. Flow chart of gravitation-based algorithm.

First, a boundary checking process is applied for each connected-component. If the center of mass of a component is in the center area of a field, this component is assigned to that field and denoted as "ASSIGNED," indicating that processing this component is unnecessary. The height and width of the center area of a field are determined as follows.

$$B = \min(\textit{field height}, \textit{field width})0.4,$$

$$\textit{height} = \textit{field height} - 2B,$$

$$\textit{width} = \textit{field width} - 2B.$$

After performing the field boundary checking on each connected-component, the gravitation of the components that are not set as "ASSIGNED" is computed by

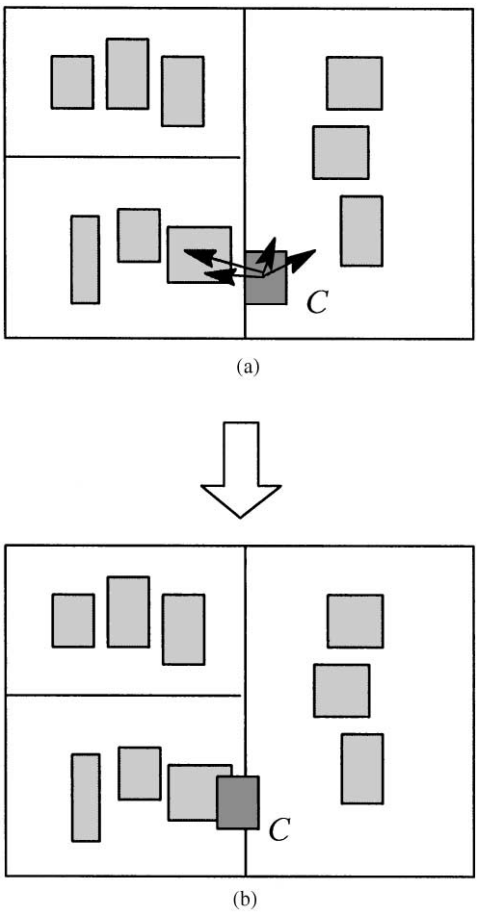


Fig. 8. (a) The gravitation of a connected-component C. (b) The gravitating result of C.

applying the gravitation function below:

$$Gf(C_a, C_b) = \frac{\overrightarrow{\mathbf{u}_{ab}}\, M_a M_b}{Distance(W_a, W_b)},$$

where W_i *is the centre of mass of* C_i, $\overrightarrow{\mathbf{u}_{ab}}$ *is the unit vector of* $(\overrightarrow{W_a - W_b})$, *and* M_i *is the mass of* C_i.

$$G(C_x) = \frac{k \sum_i Gf(C_x, C_i)}{M_x}, \quad \text{for all } C_i \neq C_x.$$

The constant k in the above expression is used to translate the gravitation into moving distance in pixels. Experimental results indicate that setting $k = 5$ can yield a better result.

After computing the gravitation for all connected-components that are not set as "ASSIGNED," connected-components can be moved according to their gravitation. Fig. 8 illustrates the movement of a component.

Repeating these steps allows us to assign most data components to the field they should be. When any of the following two conditions are satisfied, the iteration is stopped:

(1) The $G(C_x)$ of each connected-components is less than three pixels.
(2) The iteration is repeated for n times.

Although criteria 1 can be satisfied after a certain number of iteration n, a larger n increases the computational time. Experimental results indicate that setting n as 10 can yield an excellent performance.

Details of the gravitation-based algorithm are as follows:

```
Repeat {
        for a component C_i in all connected components
          if C_i is inside the center area of a field, then
            set C_i as "ASSIGNED."
        for C_i which is not set as "ASSIGNED,"
          compute the gravitation of C_i.
        move all connected component according to
        their graviation.
}until (a stop criterion is met.)
```

Fig. 9 shows the gravitating progress of connected-components at the right bottom area of Fig. 2. We can see the digit '5' gravitating towards the field it should be and all components are gravitating together.

4. Postprocessing

After the gravitation process, some connected-components not set as "ASSIGNED" will remain because their

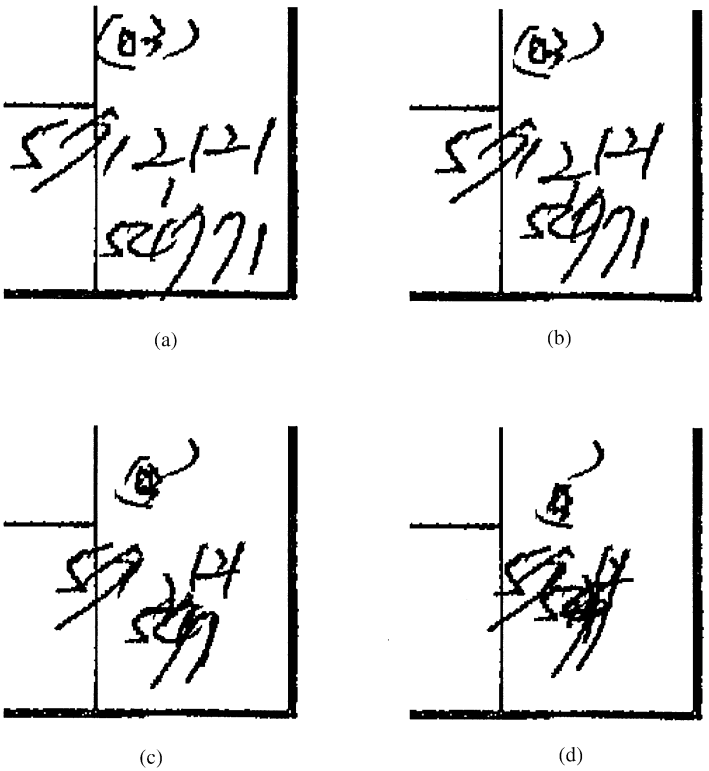


Fig. 9. (a) The gravitating progress of connected-components in sequence of (a)–(d). (a) The right bottom area of Fig. 2.

gravitation is not large enough to gravitate into the center area of any field. However, these components may have already gravitated back into the fields they belong to. For these connected-components, two postprocessing operations are performed in this study to determine the fields they belong to.

Density merging: Remaining connected-components are merged with components already located according to the black pixel density of merged results. For the connected-component C_i that is not set as "ASSIGNED", if the maximum density of the additional area introduced by adding C_i into a given field is larger than 60%, we add C_i into this field and set C_i as "ASSIGNED".

This operation is repeated until no more connected-components can be merged.

Direct assignment: Directly assign the remaining connected-components to the field that they are located in.

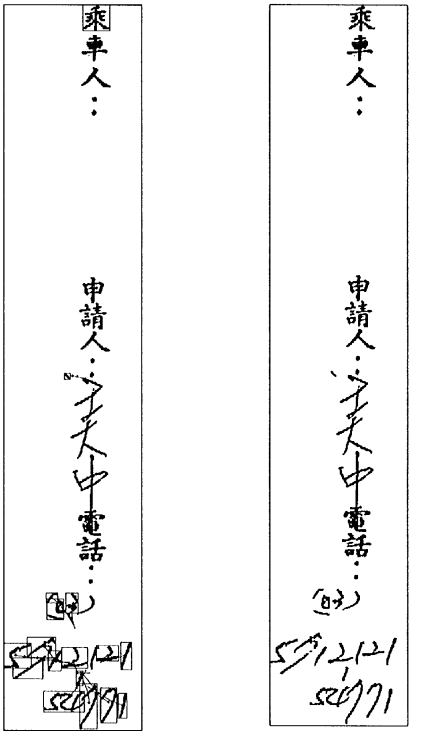
5. Experimental results

In this section, some experimental results are presented to demonstrate the validity of our proposed method. The proposed system was implemented in C language on a Pentium-300 personal computer running Linux with 96 MB RAM. The input forms used here were typical office forms and the scanning resolution was 300 dpi. Twelve form documents are tested in this experiment. Filled-in data in 95.42% of fields were correctly extracted. 72.86% of ambiguous connected-components that were filled out of fields were extracted correctly.

To demonstrate the effectiveness of our approach, we test the same data set with the algorithm proposed by Taylor et al. [2], since other approaches [4,5,7] do not group data in fields. With Taylor's method, filled-in data in 94.86% of fields were correctly extracted; 41.24% of ambiguous connected-components that were filled out of fields were extracted correctly.

Fig. 10 shows a field extracted from Fig. 2. According to this figure, the digit '5' that is written outside its field is correctly extracted. Fig. 11 shows another result. The borders shown in Figs. 10 and 11(d) are added to emphasize the extracted field data. According to Figs. 11(c) and (d), most field data are extracted correctly.

Some field data cannot be extracted correctly with the proposed approach. Some errors are introduced by the four surrounding additional fields as shown in the circled area in Fig. 11(c). We observed that some form documents require filled-in data in these surrounding areas. Meanwhile, some form documents have printed characters in these areas but no filled-in data. In addition, the data filled in these areas are usually written besides the boundary form lines. Thus, determining which field the filled-in data belong to is quite difficult. The left circled area in Fig. 12(c) shows another kind of error



(a) (b)

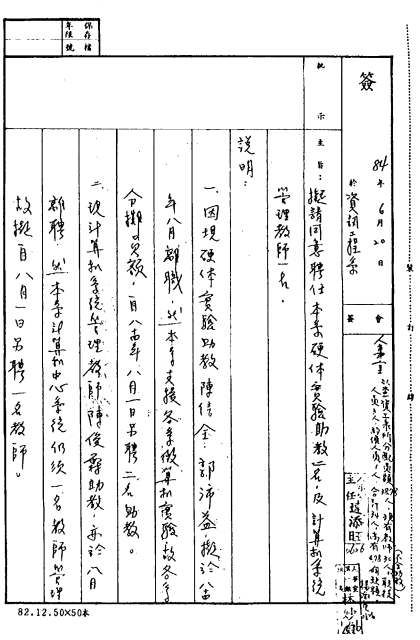
Fig. 10. (a) Moving status of a field data extracted from Fig. 2. (b) Extracted results of (a).

which is caused by the preprinted characters along the field boundaries. Preprinted characters along field boundaries can be mis-extracted if there are a lot of filled data in the field beside. The other circled area shown in Fig. 12(c) illustrates that the input data is written out of its original field (the field on top) and is close to the data in the other field. This situation makes it mis-grouped with data of the field below.

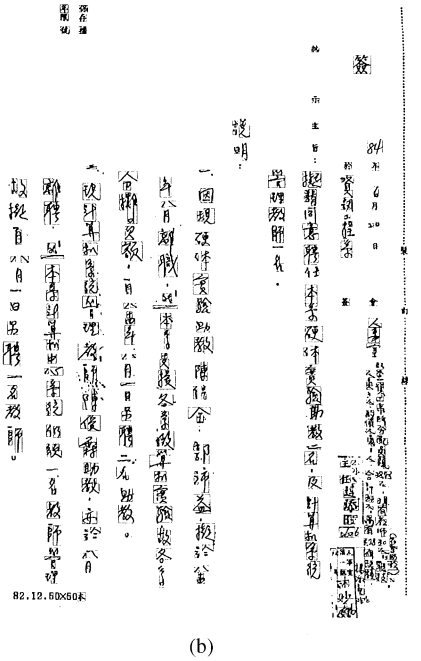
Since the proposed gravitation-based algorithm works on the connected-components located near field boundaries, the computation time of our method does not increase with the resolution of input images. The execution time used in our test images is shown in Table 1.

6. Conclusion

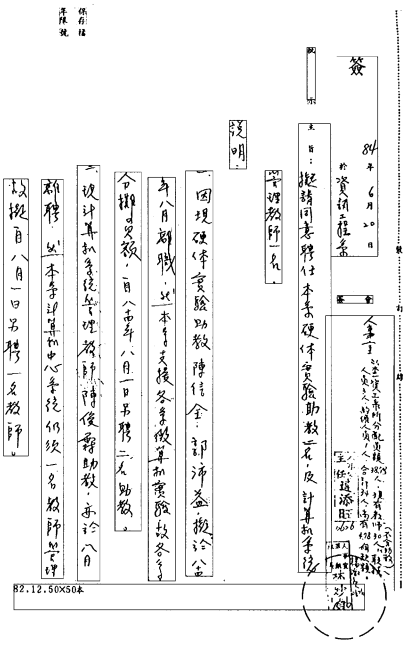
This paper presents a gravitation-based algorithm for grouping field data of form documents based on the locality property of filled-in data. The filled-in data of form documents are closely related to their fields. Understanding a form document requires exact knowledge of which field the handwritten data belong to. By adopting the locality feature of filled-in data, the proposed method can effectively extract filled-in data with their fields.



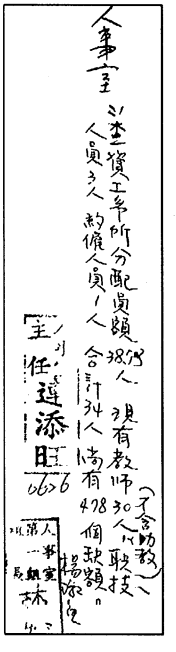
(a)



(b)



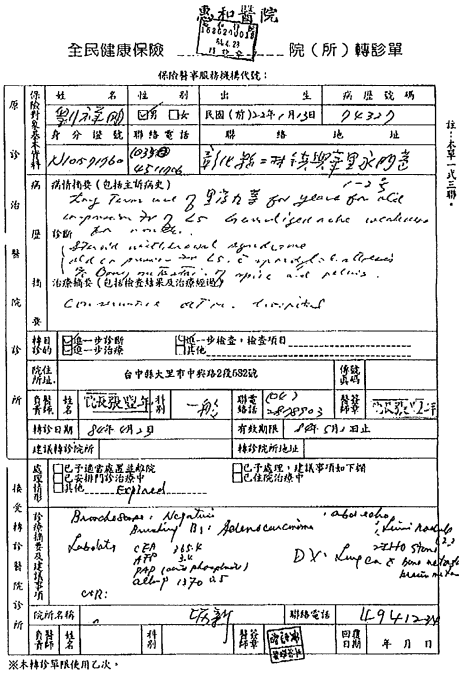
(c)



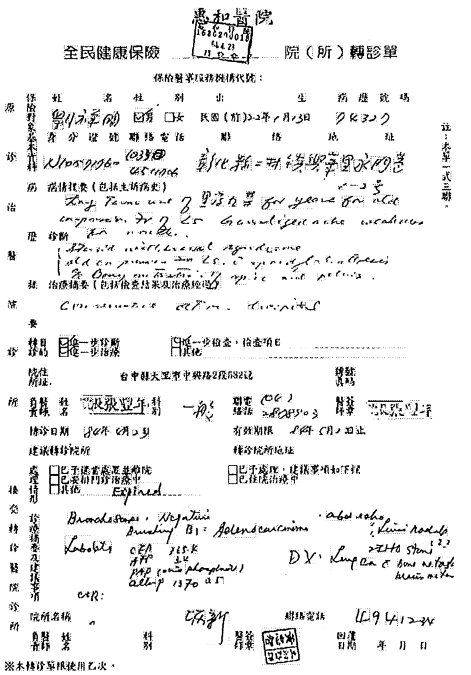
(d)

Fig. 11. (a) Original binary form image. (b) Results of line removal and connected-component detection. (c) Data extracted from (a). (d) An extracted field which contains data written out of it.

In addition, the proposed method is integrated into our form document processing system. Our upcoming work will add a blank form drop-out procedure to our system to obtain better results of filled-in data extraction. In the future, we will also attempt to optimize the source codes of this method to further enhance its performance. Character segmentation and recognition we have developed will also be



(a)



(b)

(c)

Fig. 12. (a) Original binary form image. (b) Results of line removal and connected-component detection. (c) Extracted field data.

added to the system based on the results of this work. Furthermore, a comprehensive form document processing system will be constructed on the basis of this approach.

7. Summary

In this paper, we present a gravitation-based algorithm for grouping field data of form documents based on the

Table 1
Execution time for the gravitation-based algorithm

	Resolution	Time (s)
Fig. 2	2088×1356 pixels	0.16
Fig. 11	1965×2976 pixels	1.64
Fig. 12	1000×1404 pixels	0.99

locality property of filled-in data. The filled-in data of form documents are closely related to thier fields. Understanding a form document requires exact knowledge of which field the handwritten data belong to. By adopting the locality feature of filled-in data, the proposed method can extract filled-in data with their fields correctly.

To utilize the locality property, we extract and remove form lines and apply connecte-component detection to locate form data. We compute the gravitation for each connected-component by using the black pixel counts as thier mass. Repeatedly moving these data by their gravitation according to their neighbor components, we can determine which connected-components should be extracted for a particular field.

Experimental results show that the proposed method can group data which are filled out of fields. Filled-in data in 95.42% of fields were correctly extracted. 72.86% of ambiguous connected-components that were filled out of fields were extracted correctly.

References

[1] R.G. Casey, D.R. Ferguson, K. Mohiuddin, E. Walach, Intelligent forms processing system, Machine Vision Appl. 5 (1992) 143–155.
[2] S.L. Taylor, R. Fritzson, J.A. Pastor, Extraction of data from preprinted forms, Mach. Vision Appl. 5 (1992) 211–222.
[3] B. Yu, A.K. Jain, A generic system for form dropout, IEEE Trans. Pattern Anal. Mach. Intell. 18 (11) (1996) 1127–1134.
[4] K.C. Fan, J.M. Lu, L.S. Wang, H.Y. Liao, Extraction of characters from form documents by feature point clustering, Pattern Recognition Lett. 16 (1995) 963–970.
[5] L.H. Chen, J.Y. Wang, H.Y. Liao, K.C. Fan, A robust algorithm for separation of Chinese characters from line drawings, Image Vision Comput. 14 (1996) 753–761.
[6] Y.Y. Tang, S.W. Lee, C.Y. Suen, Automatic document processing: a survey, Pattern Recognition 29 (12) (1996) 1931–1952.
[7] L.A. Fletcher, R. Kasturi, A robust algorithm for text string separation from mixed Ttxt/graphics images, IEEE Trans. Pattern Anal. Mach. Intell. 10 (6) (1988) 910–918.
[8] W. Niblack, An Introduction to Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1986, pp. 115–116.
[9] J.L. Chen, H.J. Lee, An efficient algorithm for form structure extraction using strip projection, Pattern Recognition 31 (9) (1998) 1353–1368.

About the Author—JIUN-LIN CHEN received his B.S. degree from National Chiao Tung University in Computer Science and Information Engineering in 1992. He is currently a Ph.D. student in Computer Science and Information Engineering at National Chiao Tung University. His research interest is document processing and image processing.

About the Author—HSI-JIAN LEE received the B.S., M.S. and Ph.D. degrees in Computer Engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1976, 1980, and 1984, respectively. From 1981 to 1984, he was a lecturer in the Department of Computer Engineering, National Chiao Tung University, and from 1984 to 1989 an associate professor in the same department. Since August 1989, he has been with National Chiao Tung University as a professor. He was the chairman of the department of Computer Science and Information Engineering from August 1991 to July 1997. From January 1997 to July 1998, he was a deputy director of Microelectronic and Information Research Center (MIRC). Since August 1998, he has been the general secrectory to the president of National Chiao Tung University. He was the president of the Oriental Language Computer Society (OLCS), the editor-in-chief of the International Journal of Computer Processing of Oriental Languages (CPOL), and has been an associate editor of the International Journal of Pattern Recognition and Artificial Intelligence, and Pattern Analysis and Applications. He has been a member of the executive committee of the Chinese Society on Image Processing and Pattern Recognition. He was responsible for the 1992 ROC Computational Linguistic Workshop and 1993 ROC Conference on Computer Vision, Graphics, and Image Processing. He was the program chair of the 1994 International Computer Symposium and the Fourth International Workshop on Frontiers in Handwriting Recognition (IWFHR). In 1997, he was a winner of the ten outstanding information persons of ROC. In 1992–94, he was a winner of outstanding researchers of the National Science Council, ROC. He was the general Chair of the fourth Asian Conference of Computer Vision (ACCV), January 2000. His current research interests include document analysis, optical character recognition, image processing, pattern recognition, digital library and artificial intelligence. He is a member of Phi Tau Phi.