DOCUMENT IMAGE MATCHING BASED ON COMPONENT BLOCKS

Hanchuan Peng [†‡], *Fuhui Long* [†], *Wan-Chi Siu* [†], *Zheru Chi* [†], *and David Dagan Feng* [†]

[†] Center for Multimedia Signal Processing, Department of Electronic & Information Engineering,
The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong.
Email: phc@eie.polyu.edu.hk, fhlong@eie.polyu.edu.hk, enzheru@polyu.edu.hk

[‡] Chien-Shiung Wu Laboratory, Department of Biomedical Engineering,
Southeast University, Nanjing 210096, China. Email: phc@seu.edu.cn

ABSTRACT

Document image matching is the key technique for document registration and retrieval. In this paper, a new matching algorithm based on document component block list and component block tree is proposed. Our method can effectively make use of the local information of each page block and the global information of page layout, while it is also robust to image distortion, filled-in text, and noises. This algorithm is then refined and applied to automatic data extraction of column forms. A demonstrating software package has been developed.

1. INTRODUCTION

Document image processing systems with high quality are greatly required in office automation, digital libraries, document databases, *etc*. Two important document processing functions are document image registration and retrieval, where the key technique is image matching. In recent years, many document image matching methods have been proposed for specific types of documents. For example, Cesarini *et al.* proposed a form-reader system, INFORMys, which used attributed relational graphs to automatically register data forms [1]. Shimotsuji and Asano presented a cell structure based 2-dimensional hash table to identify different forms [2]. Watanabe *et al.* proposed the description of blank form structure that includes the repetitions and positions of cells [3]. Tseng and Chen presented a form registration method based on three types of line segments [4]. Fan and Chang calculated the line crossing relationship matrix to perform form registration [5]. Luo, Watanabe and Nakayama proposed an experimental method for identifying content page of documents [6]. Watanabe and Huang utilized a predefined logical structure to acquire the layout knowledge of business cards [7]. Safari *et al.* proposed a projective geometry method to map an input document to the master document [8].

It is notable that most of the above methods are based on line segments or other local features in the image. Due to the distortion, noises, and the irregular filled-in information on document page images, it is often difficult to find out these local features. Obviously successful document image matching should combine both global page layout and local features to produce a reasonable "representation" of the document image. In this paper, we propose a new matching strategy based on component blocks of document page image. This method can effectively make use of the local information of each page block and the global information of page layout, while it is also robust to image distortion, filled-in text, and noises. Section 2 explains data structures in organizing blocks. Section 3 proposes the page matching algorithm for general page layout. Section 4 gives some primary experimental results of the page layout matching algorithm. Section 5 refines the algorithm and applies it to extracting data in column forms. Section 6 is the conclusion.

2. DATA STRUCTURES

Through our previous work in [9], a document image directly acquired from scanners or cameras is deskewed and preprocessed to remove unknown distortions and noises. The resulted image is binarized to produce the major foreground. Then all straight lines in the image are marked and erased, followed by a block-growing algorithm that finds out all rectangular component blocks. An example of the blocked image is shown in Fig.1(b), which is the result of a grayscale image with large skewness in Fig.1(a).

After image blocking, the document image can be represented by the Component Block List (CBL), which is the one-dimensional data structure of image blocks and is sorted in image blocking order. The CBL of Fig.1(b) is given in Fig.1(c), where partial attributes of the first component block are shown. These attributes include block order, block position and boundary, block type (types of text or graphics), language index (for text block), *etc*. Two major rules are defined for sorting CBL. One is sorting in location (defined as block center). The other is sorting in size.

CBL is further arranged into Component Block Tree (CBT), which can immediately produce the two-dimensional layout of all blocks. For document images, CBT is organized so that all nodes (one block corresponds to one tree node) in one tree branch are approximately in the same row (in the preprocessed image). CBT is very important for document layout control and offers flexible document image matching algorithms. Fig.1(d) is an example of the CBT.

3. IMAGE MATCHING ALGORITHM

Given a set of master images, the major difficulty in document image matching is how to decide the correct Template Block List (TBL) of master images despite the instability of blocks' positions due to arbitrariness of the input information. We

0-7803-6297-7/00/$10.00 © 2000 IEEE

propose the following image matching algorithm for general document pages.

[Algorithm-1]:

(Step 1) All blocks in CBL, as well as all template blocks in TBL, are sequenced from small to large.

(Step 2) For each block in TBL, find a most similar block in CBL according to size.

(Step 3) In CBL, Search the neighbors (within a given neighbor-threshold) of the found CBL blocks for the most similar page blocks in location.

(Step 4) Calculate distances between the current document image and all masters and select the master with the minimum distance as the correct template.

The mechanism of Algorithm-1 is seen more clearly in Fig.2: the block template B^T will first find block A as a matching and the matching is then adjusted to be block B. CBL based Algorithm-1 can be easily extended to the case of CBT, which will be more effective in handling problems of the document layout matching. This block based matching strategy can be widely applied to problems of document image retrieval, page layout management, data form reading and database construction, because both the local information of blocks and the global information of document layout can be employed.

4. EXPERIMENTAL RESULTS ON PAGE IMAGE MATCHING

Due to the space-limitation, in this short summary we only report a small part of the experimental results with a simplified configuration of Algorithm-1. The sequencing operation in step 1 is implemented according to component block sizes. The size matching in step 2 is implemented as block area matching. The location matching in step 3 is implemented as block center matching. The distance in step 4 is calculated as the sum of block center distance to the template block.

4.1 Performance for block deformation

A document image database of 50 master images (each image is normalized to be size of 1024×768 and the corresponding TBL contains about 50 component blocks) is used in the experiments. Test images, each of which should be categorized (registered) to one of the 50 masters, are generated as deformation of these master images. These test images are produced through two stages of deformation. The first stage is made up of four independent types of deformation, which are block misdetection, block addition, block displacement, and block size variation (width and height). Parameters of the corresponding parameters are: the block misdetection (the block does not appear in the CBL) probability to be 0.10, block addition (additional blocks are detected due to noises or other factors) probability to be 0.10, the block center displacement probability to be 0.10 and the displacement scale rate to be 0.10, the block width deformation probability and deformation scale rate to be 0.20 and 0.20 separately, the block height deformation probability and deformation scale rate to be 0.20 and 0.20. The second stage of deformation is about the block rotation, which has two parameters, *i.e.* rotation angle D_r and rotation probability P_r. For each pair of parameters (P_r,D_r), 200 test images are generated to examine the influence of the P_r and D_r to the correct classification rate r_c of Algorithm-1. One example of the test image is shown in Fig.3(b), which is significantly different from its master image in Fig.3(a).

As shown in Table 1, for both cases $\{D_r=15°, P_r$ varies from 0.2 to 1.0$\}$ and $\{D_r$ varies from 5° to 45°, $P_r=0.5\}$, our algorithm can produce satisfying classification, even when the test images contain strong deformation, *e.g.* 50% component blocks having at most 45° rotation, or all components blocks having at most 15° rotation. Notice that block rotation will directly lead to the significant change of block sizes. However, all above local and global deformations do not result in an important reduction of Algorithm-1's performance. In fact, for Table 1, totally there are 3600 test images (200×9×2) are generated, on which an average classification rate 91% is obtained with Algorithm-1. For the more general cases of deformation (under the approximate conditions as the second to the fourth columns in Table 1), the classification accuracy can be improved to be above 96%.

4.2 Performance for master image database size

Algorithm-1 is also tested for different master image database size under rigorous block deformations. The block deformation parameters used are: block misdetection probability to be 0.2, block addition probability to be 0.2, block center displacement probability to be 0.5 and the displacement scale rate to be 0.5, the width and height deformation probability to be 0.2 and the corresponding scale rate to be 0.2, the block rotation rate to be 0.5 and the rotation angle to be 15°. Five document image databases are used. They contain 50, 100, 200, 500 and 1350 master images, respectively. For each master image database, at least 2000 test images are generated with the block deformation parameters and then used to examine the classification rates. The results are shown in Table 2, from which we see clearly Algorithm-1 can produce good results even when the master image set is large.

5. APPLICATION: AUTOMATIC DATA EXTRACTION IN COLUMN FORMS

Column form, where data are listed in columns, is widely used in business. A typical column form has a known number of columns and uncertain number of data rows. Data items in each data row often have different lengths and different heights of centers. Hence a column form may have large physical translation between different rows and small (but not trivial!) translation between different columns. In addition, due to the irregular shapes and sizes of blocks (printed or filled-in), data items in each row may not align regularly even after page skew correction. Therefore, the automatic data extraction of column forms requires accurate page matching and data item identification techniques.

It is noticed that in real environments the border frame lines of column forms vary greatly. Closed frame, non-closed frame, single line, double lines, *etc*, can appear in column forms. Thus it is difficult to make use of the line segments based approaches to identify the form structure and to extract data. We apply Algorithm-1 to this problem because data fields in a column form are fixed, independent with types of border frame lines. The following algorithm for automatic data extraction of column forms is proposed.

[Algorithm-2]:

(Step 1) CBL matching based form registration: Call Algorithm-1 to find out the basic format of the current form. The format includes the information of column number and the accurate meaning of each column. The meaning of each column can be obtained by applying character recognition to each block in the top row.

(Step 2) Constructing CBT: Self-cluster all blocks to find out the total number of rows and each row's center in height.

(Step 3) Using CBT to annotate data fields: Reorganize blocks in each found row and set the item meaning to the data fields.

An example of applying Algorithm-2 is given in Fig.4. The column form in Fig.4(a) is firstly decomposed in blocks of Fig.4(b). Then these blocks are arranged into a CBL, which is used to find out the column number being 5. Then a CBT is constructed and each row is found out. The first row is then recognized to produce meanings of data fields. Blocks in the rest rows are then automatically annotated.

These algorithms have been implemented in a software package for demonstration. Data fields in real column forms from industry can be successfully annotated with this software. The error has only been found when there are large errors in image blocking. For this reason we have incorporated powerful image editing functions in this software.

6. CONCLUSION

In this paper a new strategy of document image matching is proposed based on the component block list and the component block tree. Two algorithms are presented and applied to a realistic industry problem of column form data extraction. The experimental results indicate the effectiveness of the page matching algorithms. A demonstration software package has been developed and applied to column data form auto-reading.

REFERENCES

[1] F., Cesarini, M., Gori, S., Marinai, and G., Soda, "INFORMys: a flexible invoice-like form-reader system," IEEE Trans on PARMI, 20(7), pp.710-745, 1998.

[2] S., Shimotsuji, and M., Asano, "Form identification based on cell structure," Proc of 13th Int. Conf on Pattern Recognition, 3, pp.793-797, 1996.

[3] T., Watanabe, Q., Luo, and N., Sugie, "Layout recognition of multi-kinds of table form documents," IEEE Trans on PARMI, 17(4), pp.432-445, 1995.

[4] L., Tseng, and R., Chen, "The recognition of form documents based on three types of line segments," Proc of 4th Int Conf on Document Analysis and Recognition, 1, pp.71-75, 1997.

[5] K., Fan, and M., Chang, "Form document identification using line structure based features," Proc of 14th Int Conf on Pattern Recognition, 2, pp.1098-1100, 1998.

[6] Q., Luo, T., Watanabe, and T., Makayama, "Identifying contents page of documents," Proc of 13th Int. Conf on Pattern Recognition, 3, pp.696-700, 1996.

[7] T., Watanabe, and X., Huang, "Automatic acquisition of layout knowledge for understanding business cards," Proc of 4th Int Conf on Document Analysis and Recognition, 1, pp.216-220, 1997.

[8] R., Safari, N., Narasimhamurthi, M., Shridhar, and M., Ahmadi, "Document registration using projective geometry," IEEE Trans on Image Processing, 6(9), pp.1337-1341, 1997.

[9] H., Peng, Z., Chi, W., Siu, and D., Feng, "PageX: an integrated document processing software for digital libraries," Proc of 2000 Int Workshop on Multimedia Data Storage, Retrieval, Integration, and Applications, Hong Kong, pp.203-207, Jan. 2000.

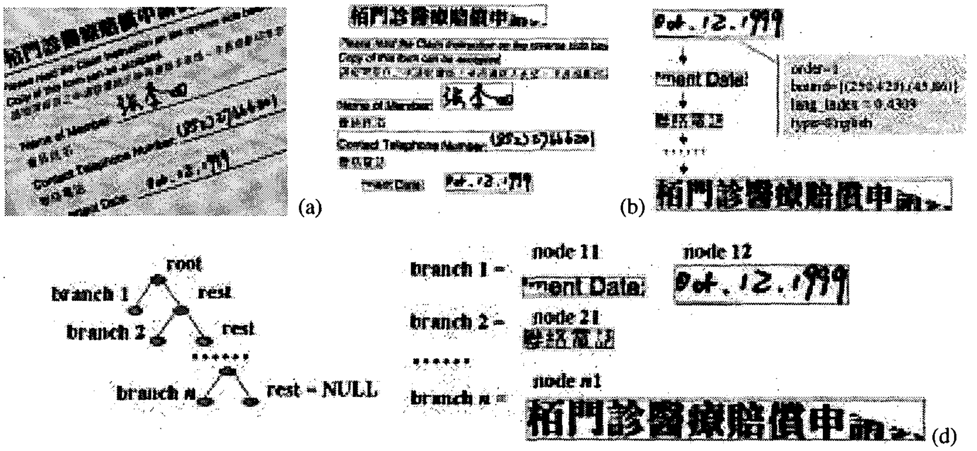


Fig.1 Partial results in document image blocking and data organizing. (a) The input grayscale image with unknown skewness; (b) Results of image blocking after binarization and skew correction; (c) CBL of the decomposed image and partial attributes of the first block in this CBL; (d) Architecture of CBT.

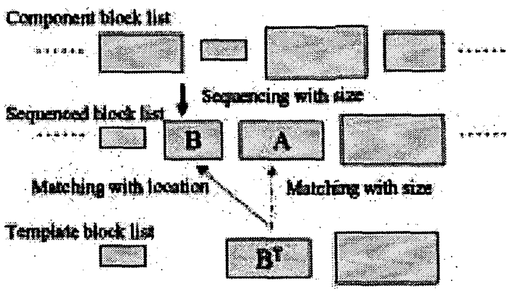


Fig.2 An algorithm of general page matching

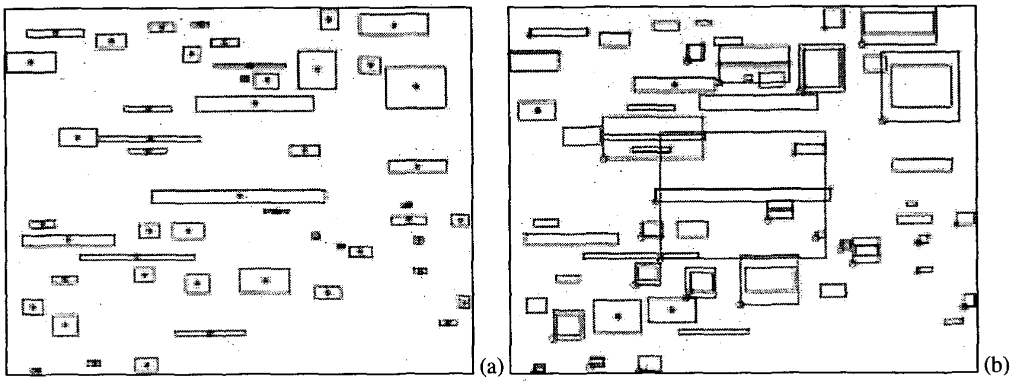


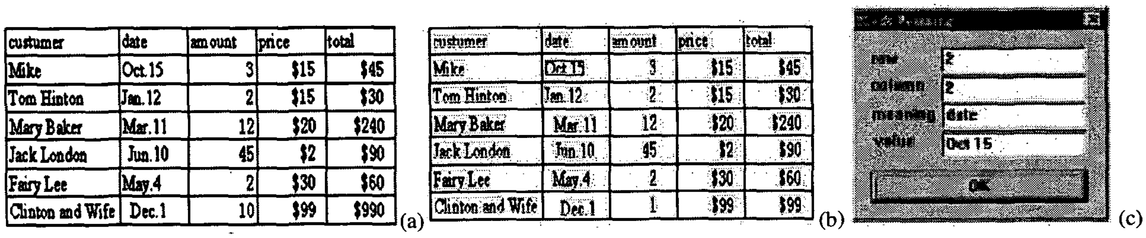
Fig.3 Example of the master image (a) and one of its deformation images (b).
(For visualization, the deformation image is superimposed on the master image in (b))

Table 1: Categorization performance of Algorithm-1 for block rotation deformation

P_r (D_r=15°)	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
r_c (%)	99.5	99.5	99.0	97.0	94.0	89.0	85.0	75.5	62.0
D_r (P_r=0.5)	5°	10°	15°	20°	25°	30°	35°	40°	45°
r_c (%)	99.5	97.5	97.0	96.5	94.0	91.5	90.0	88.5	83.0

Table 2: Categorization performance of Algorithm-1 for different master image database size

Master image number	50	100	200	500	1350
Test image numbers	2000	2000	2000	2000	2700
r_c (%)	92.30	91.45	87.40	83.60	70.00



custumer	date	amount	price	total
Mike	Oct.15	3	$15	$45
Tom Hinton	Jan.12	2	$15	$30
Mary Baker	Mar.11	12	$20	$240
Jack London	Jun.10	45	$2	$90
Fairy Lee	May.4	2	$30	$60
Clinton and Wife	Dec.1	10	$99	$990

(a)

custumer	date	amount	price	total
Mike	Oct.15	3	$15	$45
Tom Hinton	Jan.12	2	$15	$30
Mary Baker	Mar.11	12	$20	$240
Jack London	Jun.10	45	$2	$90
Fairy Lee	May.4	2	$30	$60
Clinton and Wife	Dec.1	1	$99	$99

(b)

row 2
column 2
meaning date
value Oct 15

(c)

Fig.4 Partial results in data extraction of a column form. (a) A column form sample; (b) Extracted blocks (yellow framed) from the sample image; (c) Extracted data of the block "Oct 15" (red framed in (b)).