International Journal of u- and e- Service, Science and Technology
Vol.8, No. 12 (2015), pp.389-402
http://dx.doi.org/10.14257/ijunesst.2015.8.12.39

Auto-reconstruction of Shredded Document based on Matching Models

Mengmeng Yu[1], Pinjie Ye[1], ShuopingWang[1,*] and Honghao Gao[2]

[1]*Zhejiang University City College, Hangzhou, China*
[2]*Computing Center, Shanghai University, Shanghai, China*
**wangsp@zucc.edu.cn*

Abstract

Shredding auto-reconstruction is a hot research topic in pattern recognition. The research progress can produce certain effect to various fields. The purpose of this paper is to study shredding auto-reconstruction based on regular shredded document from shredders, to obtain a practical and efficient splicing algorithm to auto-reconstruction of strip shaped shredded text documents and block shapedshredded text documents. For strips, this paper uses the pretreatment, the similarity matching model, combined with the optimalHamilton path algorithm, for which we get a good result with 100% correct rate and no human intervention. For blocks, first, this paper pretreats the fragments. And then uses the row cluster model to divide all debris to some rows, and then uses the similarity model with direct reverse matching model to achieve the shredding auto-restore in different rows. At last, we use line spacing matching model to get the result that has a high correct rate reaching to 90% with little human intervention. In this paper, the design of some algorithms is original. Combined with the present feasible algorithm, we get an ideal result.

Keywords: *document auto-recovery, row cluster, similarity matching model, line-spacing matching*

1. Introduction

Nowadays, paper document is the most popular office file applied to different fields, so the shredder plays a heavy role in the office. Sometimes many important files are damaged by accident, leading to a huge loss because of missing some significant files. So this paper focuses on building some model to restore shredded paper (strips and blocks) with little human intervention [1].

2. Reconstruction of Strip Shaped Shredded Documents

2.1 Image Preprocessing

2.1.1. Binaryzation: Considering the large-scale value of the gray image, which may have great influence to restore, this paper make binary image processing at first.

Step1. Get the gray-level matrix

Suppose there are n pieces, then the gray-level matrix $A^{(k)}$ of k th is shown like formula (2.1). The range of grey scale is [0,255], and the value 255 means the point is an empty space, and the value between [0,254] means the point is a text pixel.

* Corresponding Author

ISSN: 2005-4246 IJUNESST

$$A^{(k)}=\begin{bmatrix} a_{1,1}^{(k)} & a_{1,2}^{(k)} & \cdots & a_{1,c}^{(k)} \\ a_{2,1}^{(k)} & a_{2,2}^{(k)} & \cdots & a_{2,c}^{(k)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{r,1}^{(k)} & a_{r,2}^{(k)} & \cdots & a_{r,c}^{(k)} \end{bmatrix}, a_{i,j}^{(k)} \in [0,255] \tag{2.1}$$

While r says the number of rows, c says the number of columns.

Step2. Build the 0/1 matrix

Transfer gray-level matrix $A^{(k)}$ to 0/1 matrix $C^{(k)}$, the rule of transform is shown like formula (2.2).

$$c_{i,j}^{(k)}=\begin{cases} 1, & a_{i,j}^{(k)}<255 \\ 0, & a_{i,j}^{(k)}=255 \end{cases}, \quad i=1,\cdots,r, \quad j=1,\cdots,c \tag{2.2}$$

So get the 0/1 matrix $C^{(k)}$ is shown like formula (2.3).

$$C^{(k)}=\begin{bmatrix} c_{1,1}^{(k)} & c_{1,2}^{(k)} & \cdots & c_{1,c}^{(k)} \\ c_{2,1}^{(k)} & c_{2,2}^{(k)} & \cdots & c_{2,c}^{(k)} \\ \vdots & \vdots & \ddots & \vdots \\ c_{r,1}^{(k)} & c_{r,2}^{(k)} & \cdots & c_{r,c}^{(k)} \end{bmatrix}, k=1,2,\cdots n \tag{2.3}$$

2.1.2. Get the Edge Matrix: According to the data of matrix $A^{(k)}$, extract the edge data of matrix, then we could get the edge matrix $B^{(k)}$ (shown like formula (2.4)).

$$B^{(k)}=\begin{bmatrix} a_{1,1}^{(k)} & a_{1,c}^{(k)} \\ a_{2,1}^{(k)} & a_{2,c}^{(k)} \\ \vdots & \vdots \\ a_{r,1}^{(k)} & a_{r,c}^{(k)} \end{bmatrix} \quad k=1,\cdots,n \tag{2.4}$$

Also, according to the data of 0/1 matrix $C^{(k)}$ can get the 0/1 edge matrix $D^{(k)}$ shown like the formula (2.5).

$$D^{(k)}=\begin{bmatrix} c_{1,1}^{(k)} & c_{1,c}^{(k)} \\ c_{2,1}^{(k)} & c_{2,c}^{(k)} \\ \vdots & \vdots \\ c_{r,1}^{(k)} & c_{r,c}^{(k)} \end{bmatrix} \quad k=1,\cdots,n \tag{2.5}$$

2.2 Model for Strip Shaped Shredded Document

Similarity matching model is the key algorithm. Strip-type shredded papers' data are big, so it is easy to get an ideal result.

2.2.1. Boundary Flag Definition Model: daily text files are usually surrounded by large margins, so the blank space could be used as the basis for looking for the round pieces. Based on the left margin, we can get the first strip debris, and then use model to pick up the next debris and go on, then we could get the result [1].

Suppose $flag_j^{(k)}$ means whether the j th column of the k th debris is blank, its definition sees the formula (2.6).

$$flag_j^{(k)} = \begin{cases} 1, & \sum_{i=1}^{r} c_{i,j}^{(k)} = 0, \quad j = 1,\cdots,c \\ 0, & \sum_{i=1}^{r} c_{i,j}^{(k)} \neq 0, \quad j = 1,\cdots,c \end{cases} \tag{2.6}$$

Then get the width of left blank area $D_l^{(k)}$ of debris, like the formula (2.7).

$$D_l^{(k)} = \sum_{j=1}^{w} flag_j^{(k)}, \quad w = \min \quad r\{flag_j^{(k)} = 0\} \tag{2.7}$$

While w means the column $j(j = 1,\cdots,c)$ the first text pixel shows, so we can get the width of left blank area of each of the debris, pick the debris whose left margin is biggest as the first debris, sees the formula (2.8).

$$P_l = \max\left\{D_l^{(k)}\right\}, \quad k = 1,\cdots n \tag{2.8}$$

In the same way, we can get the last debris P_r, sees the formula (2.9).

$$P_r = \max\left\{D_r^{(k)}\right\}, \quad k = 1,\cdots n \tag{2.9}$$

2.2.2. Similarity Matching Model: This paper use similarity to define whether two pieces of debris are match. As shown in Figure 2.1, suppose the k th debris is the debris matched, the x th debris is the debris to match, the c th column of the x th debris and the first column of the k th debris both have r pixels which contain the value of grey level. Get the difference of the grey level of the same row of two pieces of debris. If the difference is smaller, the similarity is higher [2].

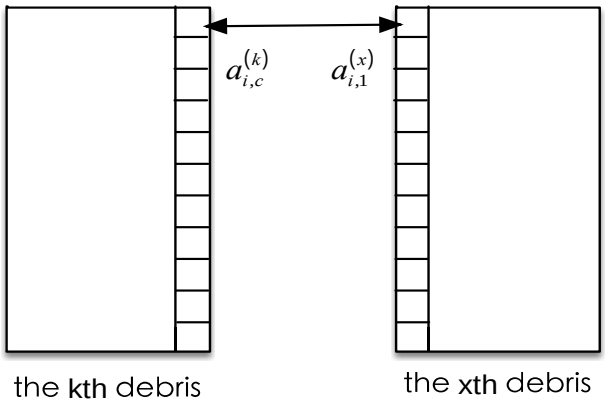

Figure 2.1. The Calculation of Similarity

The difference can be defined as formula (2.10).

$$dif(k,x) = \sum_{i=1}^{r} \left| a_{i,c}^{(k)} - a_{i,1}^{(x)} \right|$$
$$dif(k,x) = \sum_{i=1}^{r} \left(a_{i,c}^{(k)} - a_{i,1}^{(x)} \right)^2 \quad (2.10)$$

When the formula （2.11） is established, we can get the debris who is most suited debris for the kth debris. That is, the $k+1$th debris should be the debris who make the least difference withthe kth debris.

$$dif(k,k+1) = \min \quad dif(k,x) \quad ,x = 1,\cdots,n \quad (2.11)$$

3. Reconstruction of Block Shaped Shredded Document

3.1 Summarize

The flow chart of restore for massive-type debris is shown like Figure 3.1. At first, this paper use row cluster model to classify, combined with human interfere, to make them to M groups. Then use inline debris restore model to restore the shredded paper within every group, combined with human interfere if necessary. At last, use line-spacing matching model to get the final result [2, 3].

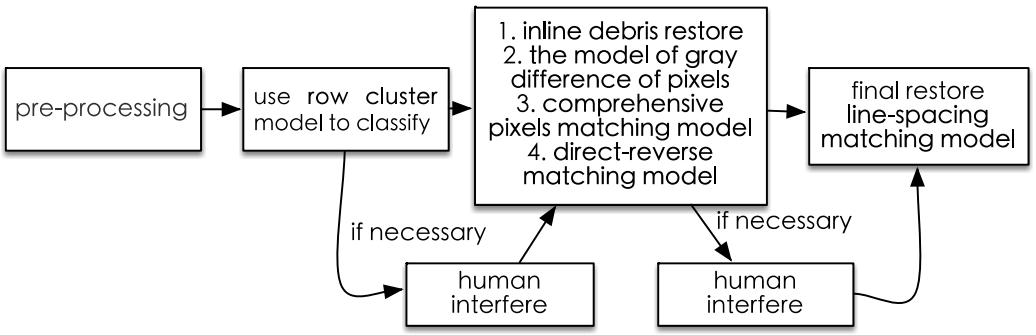

Figure 3.1. Flow Chart of Restore for Massive-type Debris

3.2 Row Cluster Model

3.2.1 K-means Clustering Algorithm: Given a set of observations $\{x_1, x_2, \cdots, x_n\}$, find K clustering centers $\{a_1, a_2, \cdots, a_K\}$ to minimize the within-cluster sum of squares. In other words, as shown like formula (3.1), its objective is to find:

$$W_n = \sum_{i=1}^{n} \min_{1 \le j \le K} \left| x_i - a_j \right|^2 \tag{3.1}$$

3.2.2. Text Feature of English and Chinese: There are some differences between English text and Chinese text, so this paper discusses them respectively.

◆ **Chinese Text Feature**

For every piece of Chinese debris $A^{(k)}$, record the row number d, so the $\P_i^k$ means the debris $A^{(k)}$'s eigenvalue in $d = i$ row. Scan every piece of Chinese debris $A^{(k)}$ from $d = 1$, if it is text pixel in d th row, that is, $\P_i^k = 1$,or, $\P_i^k = 0$, like formula (3.2).

$$\partial_i^{(k)} = \begin{cases} 1, & \sum_{i=1}^{r} c_{i,j}^{(k)} = 0, \quad i = 1, \cdots, r \\ 0, & \sum_{i=j}^{j} c_{i,j}^{(k)} \neq 0, \quad i = 1, \cdots, r \end{cases} \tag{3.2}$$

Take a piece of Chinese text debris as an example, the result is like Figure 3.2.

Figure 3.2. Chinese Debris Text Feature Analysis

We can get the feature vectors of $A^{(k)}$, like formula (3.3).

$$\partial^{(k)} = \left[\partial_1^{(k)}, \partial_2^{(k)}, \cdots, \partial_r^{(k)} \right]^T \tag{3.3}$$

◆ **English Text Feature**

The English text is not regular, but they can be written in the middle of 4 lines, like Figure 3.3. And many letters do not exceed the third line, only g, j, p, q, y and Q do.

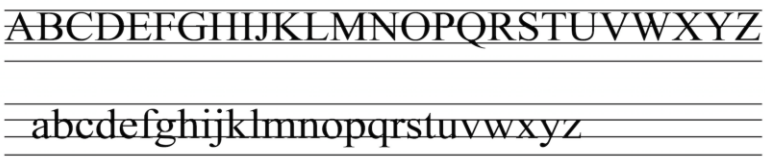

Figure 3.3. 26 Letter Analysis

This paper recognizes that the space between the second line and the third line is the integral part lowercases exist. For every piece of English debris $A^{(k)}$, record the row

number d, so the $¶_i^k$ means the debris $A^{(k)}$'s eigenvalue in $d=i$ row. Scan every piece of English debris $A^{(k)}$ from $d=1$, if there are M not-null text pixel in d th row, that is, $¶_i^k=1$,or, $¶_i^k=0$, like formula (3.4).

$$\partial_i^{(k)}=\begin{cases}1, & \sum_{i=1}^{r} c_{i,j}^{(k)} \geq M, \quad i=1,\cdots,r \\ 0, & \sum_{i=j}^{j} c_{i,j}^{(k)} < M, \quad i=1,\cdots,r\end{cases} \tag{3.4}$$

Take a piece of English text debris as an example, the result is like Figure 3.4.

Figure 3.4. English Debris Text Feature Analysis

We can get the feature vectors of English debris $A^{(k)}$, like formula (3.5).

$$\partial^{(k)}=\left[\partial_1^{(k)},\partial_2^{(k)},\cdots,\partial_r^{(k)}\right]^T \tag{3.5}$$

3.2.3. Row Cluster Model: According to this method, the gray strip can be got. If gray strip of two pieces are similar, they have the same image features, located in the same row for the original file [4].

After getting row feature vector, this paper define that debris $A^{(j)}$belongs to the debris $A^{(i)}$'s row, make difference or operation $\oplus$ of $¶^{(i)}$ and $¶^{(j)}$ to get the module as line-spacing matching distance, sees formula (3.6).

$$d(i,j)=\left(\partial^{(i)}\oplus\partial^{(j)}\right) \tag{3.6}$$

Define $q$as the Threshold of line-spacing matching distance, when formula (3.7) is established, the matching rate that debris $A^{(j)}$belongs to the debris $A^{(i)}$'s row is high.

$$d(i,j)\,³\,q \tag{3.7}$$

According to this model, classify all debris to some groups.

3.3 Similarity Matching Model

For massive-type debris, the similarity model is improved a lot.

3.3.1 The Model of Gray Difference of Pixels: the pixels of massive-type debris are less, so there are some special conditions. Like Figure 3.5.Due to specificity of cut location, the

difference between the left debris and the right debris is big, like 'h'. So this paper define not-null pixel matching model and pure-black pixel matching model to restore.

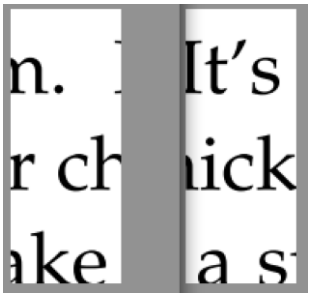

Figure 3.5. Special Condition

3.3.2. Not-null Pixels Matching Model: Compare right vector of the k th debris with the left vector of the x th debris, like Figure 3.6.

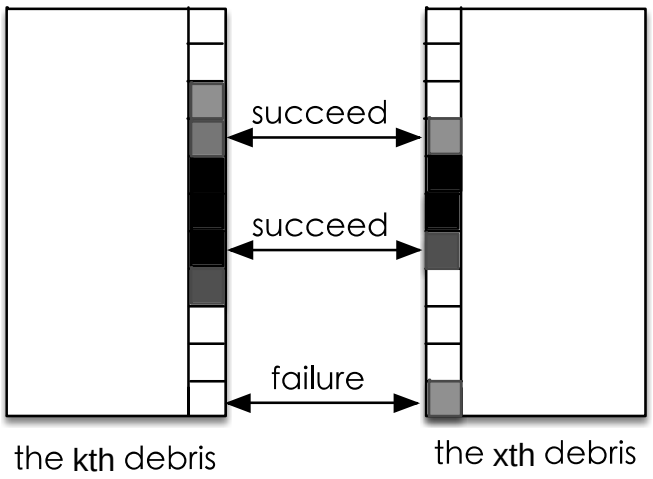

Figure 3.6. Not-null Pixel Matching Model

Define the sum of pixels of right vector of the k th debris is $s_{unempty}^{(k)}$, the sum of pixels of left vector of the x th debris is $s_{unempty}^{(x)}$, $s_{unempty}$ is the number of anastomotic pixels of two debris, $s_{unempty}$ is equal to 0 before matching, the calculation after matching is shown like formula (3.10)

$$s_{unempty} = s_{unempty} + 1, \ if \ c_{i,c}^{(k)} = c_{i,1}^{(x)} = 1, i = 1, \cdots, r \tag{3.10}$$

The rate $r_{unempty}^{(k)}$ is shown like formula (3.11).

$$r_{unempty}^{(k)} = \frac{s_{unempty}}{s_{unempty}^{(k)}} \tag{3.11}$$

The rate $r_{unempty}^{(x)}$ is shown like formula (3.12).

$$r_{unempty}^{(x)} = \frac{s_{unempty}}{s_{unempty}^{(x)}} \tag{3.12}$$

3.3.3. Pure-black Pixels Matching Model: Like the description in 3.3.2, Compare right vector of the k th debris with the left vector of the x th debris, like Figure 3.7.

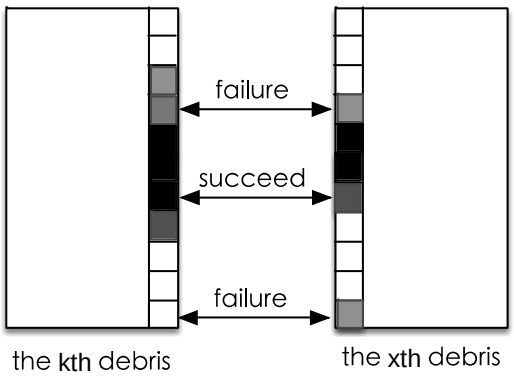

Figure 3.7. Pure-black Pixel Matching Model

The rate of pure black is shown like formula (3.13).

$$r_{black}^{(k)} = \frac{s_{black}}{s_{black}^{(k)}} \tag{3.13}$$

The rate $r_{black}^{(x)}$ is shown like formula (3.14).

$$r_{black}^{(x)} = \frac{s_{black}}{s_{black}^{(x)}} \tag{3.14}$$

3.4 Inline Debris Restore Model-- Comprehensive Pixels Matching Model

The forward and reverse direction matching will use two different models, they are matching model for forword direction and inconformity matching model.

- Matching model for forward direction

There are many different situations.

- General

$$sim_{composite}(k,x) = \frac{1}{4}\left(\frac{s_{unempty}}{s_{unempty}^{(k)}} + \frac{s_{unempty}}{s_{unempty}^{(x)}} + \frac{s_{black}}{s_{black}^{(k)}} + \frac{s_{black}}{s_{black}^{(x)}}\right), 0\% \le sim_{composite}(k,x) \le 100\% \tag{3.15}$$

- Both null

The situation is shown like Figure 3.8, $s_{unempty} = 0$, $s_{unempty}^{(k)} = 0$, $s_{unempty}^{(x)} = 0$.

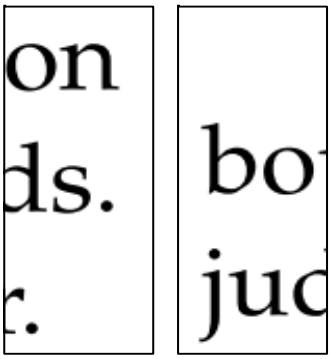

Figure 3.8. No Null Pixel Point

The matching rate $sim_{composite}(k,x)$ can be got through formula (3.16).

$$sim_{composite}(k,x)=1 \tag{3.16}$$

◆ No Null Matching Point

The situation is like Figure 3.9, $s_{unempty}=0$, $s_{unempty}^{(k)}$ and $s_{unempty}^{(x)}$ could be 0 or not.

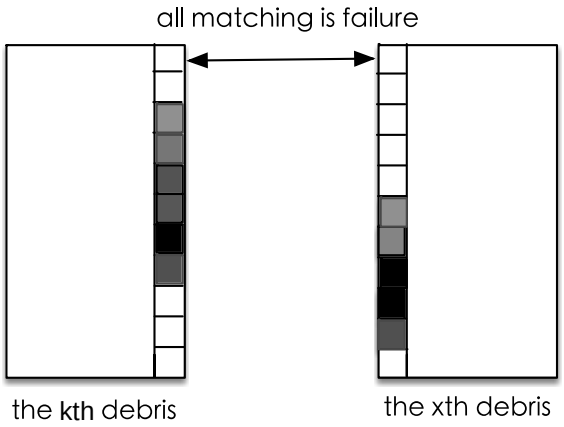

Figure 3.9. No Null Matching

The comprehensive matching rate is like formula (3.17), π is a threshold value.

$$sim_{composite}(k,x)=\begin{cases}\frac{1}{2}\left(1+\frac{m-s_{unempty}^{(k)}}{m}\right) & s_{unempty}^{(k)}\neq 0, s_{unempty}^{(x)}=0\\ \frac{1}{2}\left(1+\frac{m-s_{unempty}^{(x)}}{m}\right) & s_{unempty}^{(k)}=0, s_{unempty}^{(x)}\neq 0\\ \frac{s_{unempty}^{(k)}}{s_{unempty}^{(x)}} & s_{unempty}^{(k)}\neq 0, s_{unempty}^{(x)}\neq 0, \frac{s_{unempty}^{(k)}}{s_{unempty}^{(x)}}<1\\ \frac{s_{unempty}^{(x)}}{s_{unempty}^{(k)}} & s_{unempty}^{(k)}\neq 0, s_{unempty}^{(x)}\neq 0, \frac{s_{unempty}^{(x)}}{s_{unempty}^{(k)}}<1\end{cases} \tag{3.17}$$

◆ No Pure Black Pixel

The situation is shown like Figure 3.10, $s_{black}=0$, $s_{black}^{(k)}=0$, $s_{black}^{(x)}=0$.

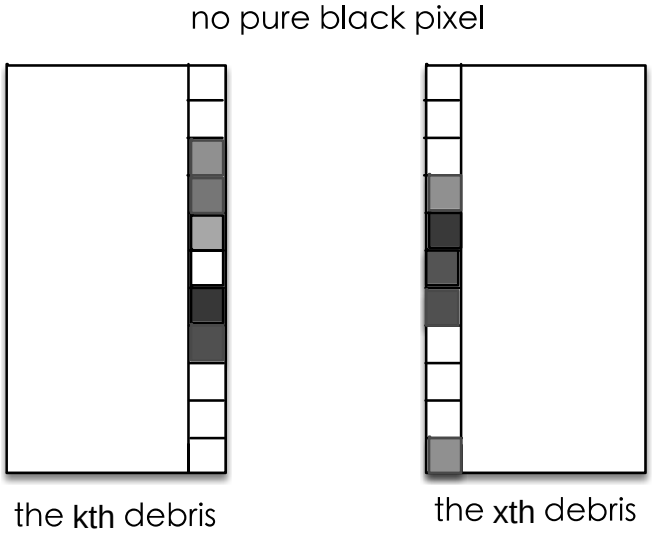

Figure 3.10. No Pure Black Pixel

$sim_{composite}(k,x)$ can be got through formula (3.18).

$$sim_{composite}(k,x)=\frac{1}{2}\left(\frac{s_{unempty}}{s_{unempty}^{(k)}}+\frac{s_{unempty}}{s_{unempty}^{(x)}}\right), 0\% \le sim_{composite}(k,x) \le 100\% \tag{3.18}$$

◆ No pure black matching point

The situation is shown like Figure 3.11, $s_{black}=0$, $s_{black}^{(k)}=0$, $s_{black}^{(x)}=0$.

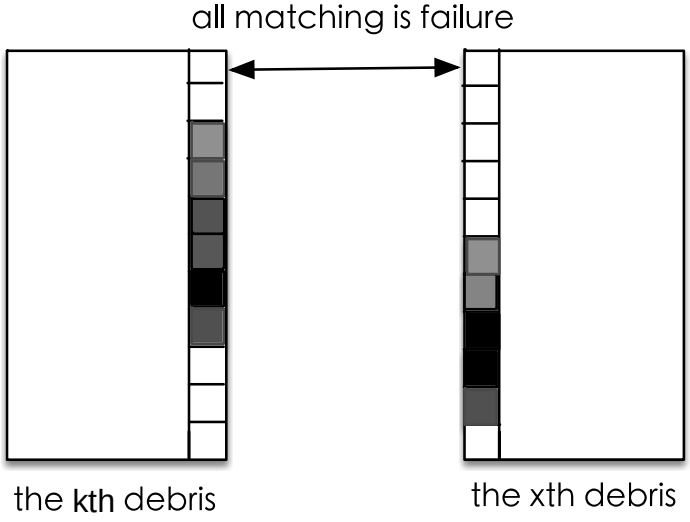

Figure 3.11. No Pure Black Matching Point

The comprehensive matching rate is like formula (3.19), π is a threshold value.

$$sim_{composite}(k,x)=\begin{cases}\frac{1}{3}\left(\frac{s_{unempty}}{s_{unempty}^{(k)}}+\frac{s_{unempty}}{s_{unempty}^{(x)}}+\frac{m-s_{black}^{(k)}}{m}\right) & s_{black}^{(k)}\neq 0, s_{black}^{(x)}=0\\ \frac{1}{3}\left(\frac{s_{unempty}}{s_{unempty}^{(k)}}+\frac{s_{unempty}}{s_{unempty}^{(x)}}+\frac{m-s_{black}^{(x)}}{m}\right) & s_{black}^{(k)}=0, s_{black}^{(x)}\neq 0\\ \frac{1}{3}\left(\frac{s_{unempty}}{s_{unempty}^{(k)}}+\frac{s_{unempty}}{s_{unempty}^{(x)}}+\frac{s_{black}^{(k)}}{s_{black}^{(x)}}\right) & s_{black}^{(k)}\neq 0, s_{black}^{(x)}\neq 0, \frac{s_{black}^{(k)}}{s_{black}^{(x)}}<1\\ \frac{1}{3}\left(\frac{s_{unempty}}{s_{unempty}^{(k)}}+\frac{s_{unempty}}{s_{unempty}^{(x)}}+\frac{s_{black}^{(x)}}{s_{black}^{(k)}}\right) & s_{black}^{(k)}\neq 0, s_{black}^{(x)}\neq 0, \frac{s_{black}^{(x)}}{s_{black}^{(k)}}<1\end{cases} \tag{3.19}$$

- Inconformity Matching Model
- General

$$sim_{composite}(k,x)=\frac{1}{2}\left(\frac{s_{unempty}}{s_{unempty}^{(k)}}+\frac{s_{unempty}}{s_{unempty}^{(x)}}\right), 0\%\le sim_{composite}(k,x)\le 100\% \tag{3.20}$$

- All is null

$$sim_{composite}(k,x)=1 \tag{3.25}$$

- No Null Matching Point

$$sim_{composite}(k,x)=\begin{cases}\frac{1}{2}\left(1+\frac{m-s_{unempty}^{(k)}}{m}\right) & s_{unempty}^{(k)}\neq 0, s_{unempty}^{(x)}=0\\ \frac{1}{2}\left(1+\frac{m-s_{unempty}^{(x)}}{m}\right) & s_{unempty}^{(k)}=0, s_{unempty}^{(x)}\neq 0\\ \frac{s_{unempty}^{(k)}}{s_{unempty}^{(x)}} & s_{unempty}^{(k)}\neq 0, s_{unempty}^{(x)}\neq 0, \frac{s_{unempty}^{(k)}}{s_{unempty}^{(x)}}<1\\ \frac{s_{unempty}^{(x)}}{s_{unempty}^{(k)}} & s_{unempty}^{(k)}\neq 0, s_{unempty}^{(x)}\neq 0, \frac{s_{unempty}^{(x)}}{s_{unempty}^{(k)}}<1\end{cases} \tag{3.22}$$

3.5 Line-spacing Matching Model

There are two different situations, one is that two pieces of debris contain text pixels, another is not. The first one should use margin picels mathing model, the other one should use line-spacing matching model [5].

3.5.1. Margin Pixels Matching Model:

It is like the model in Chapter 2.

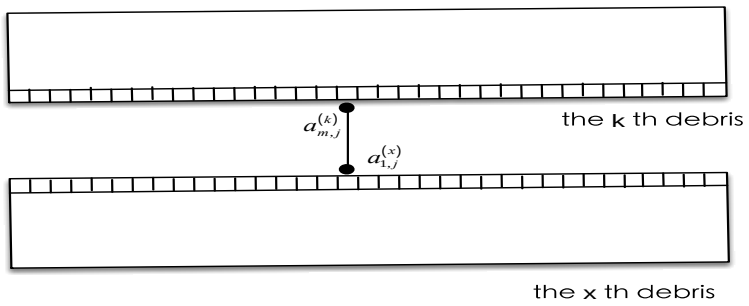

Figure 3.12. Mathing Method

The difference $dif(k,x)$ is like formula (3.23).

$$dif(k,x)=\sum_{j=1}^{c}\left(a_{r,j}^{(k)}-a_{r,j}^{(x)}\right)^2 \tag{3.23}$$

If formula (3.24) is established, the $k+1$th is the matching debris.

$$dif(k,k+1)=\min \quad dif(k,x) \quad ,x=1,\cdots,n \tag{3.24}$$

3.5.2. Line-spacing Matching Model:

Define the line spacing is constant L, the margin of upper boundary is $T_b^{(k)}$, the margin of lower boundary is $T_t^{(x)}$, space between the two pieces of debris is $l(k,x)$, like formula (3.25).

$$l(k,x)=\left(r-T_b^{(k)}\right)+T_t^{(x)} \tag{3.25}$$

The space l between the two pieces of debris is like Figure 3.13.

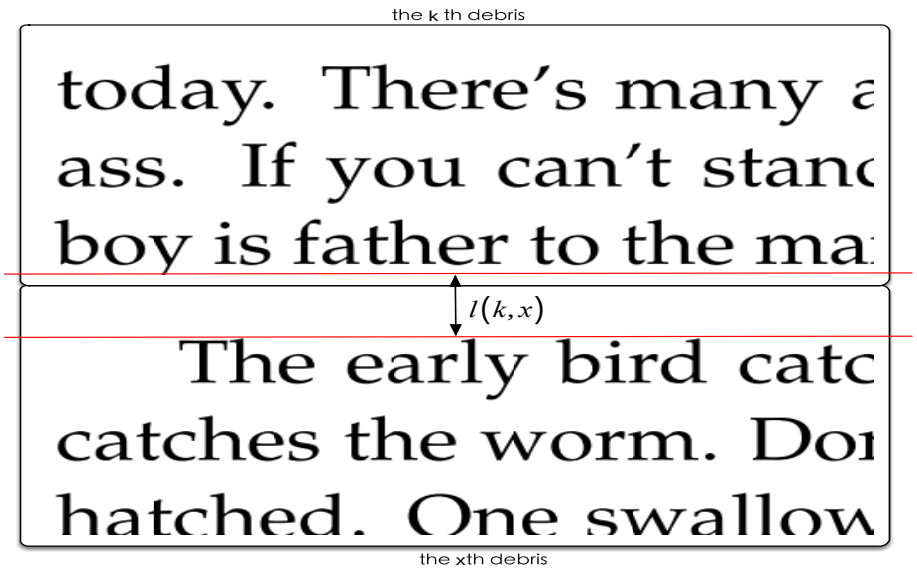

Figure 3.13. Line Space

If formula (3.26) is established, the x th debris is the matching debris.

$$L - l(k,x) = 0 \tag{3.26}$$

$L - l(k,x)$ may not 0, so get the $d(k,x)$ at first, like formula (3.27)。

$$d(k,x) = L - l(k,x) \tag{3.27}$$

Then get the smallest number. When formula (3.32) is established, it can be recognized that the x th debris is the matching debris.

$$d(k,x) = \min\ d(k,i), i = 1,\cdots,r \tag{3.28}$$

4. Conclusion

This paper make some algorithm research.For strip pdebris, achieved 100% accuracy rate and zero human intervention. For massive debris, clustering accuracy was 91.39%, inline stitching accuracy rate of 92.34 percent, up and down splicing accurate rate of 100%, the result is correct, and high accuracy, have reached more than 90%, two manual intervention points, just adjust a few adjustments each intervention point, with satisfactory results.

The model provided by this paper combines the advantages of a good number of algorithms, and avoid some of the disadvantages of the algorithm, obtained the desired results.

Acknowledgment

The authors are grateful to the director, SHUOPINGWANG, who gave us active support and lots of advice. What's more, the authors are grateful "Zhejiang University City College", cause it provides additional financial support.

References

[1] W. Kong and B. B. Kimia, "On solving 2D and 3D puzzles using curve matching", Computer Vision and Pattern Recognition, Proceedings of the IEEE Computer Society Conference, IEEE, **(2001)**.
[2] J. C. McBride and B. B. Kimia, "Archaeological fragment reconstruction using curve-matching", Computer Vision and Pattern Recognition Workshop, Conference, IEEE, **(2003)**.
[3] P. De Smet, J. De Bock and E. Corluy, "Computer vision techniques for semi-automatic reconstruction of ripped-up documents", AeroSense International Society for Optics and Photonics, **(2003)**, pp. 189-197.
[4] C. Papaodysseus, T. Panagopoulos and M. Exarhos, "Contour-shape based reconstruction of fragmented, 1600 bc wall paintings", Signal Processing, IEEE Transactions, vol. 50, no. 6, **(2002)**, pp. 1277-1288.
[5] H. Y. Lin and W. C. F. Chiang, "Reconstruction of shredded document based on image feature matching", Expert Systems with Applications, vol. 39, no. 3, **(2012)**, pp. 3324-3332.