Reconstructing Regular Shredded Documents by Similarity Measure

Cheng Le, Nanjie Deng, Xiao Jin
Beijing University of Posts and Telecommunications
BUPT
Beijing, China
lechengbupt@gmail.com

***Abstract*—This paper describes a method to the problem of reconstructing regular shredded documents by similarity measure. Regular shredded document can be quantized as gray-level matrix or two-value matrix. So we can match the gray value of the edge of the shredded document, which is indicated as similarity value. We calculate the similarity value by using improved minimum-error method, a kind of similarity measure method. The two shredded owning the biggest similarity value is probable to be the neighboring ones. So we can reconstruct these regular shredded documents through this method until it is recovered.**

Keywords-reconstruct shredding; minimum-error method; similarity algorithm; similarity measure

I. INTRODUCTION

Documents often suffer damages at some levels, such as moisture, obliteration, charring and shredding. This paper focuses on the shredding. Shredding can be performed by a machine or by hand. In both cases, documents need to be reconstructed so that the people can analyze them.

In our modern society, there are many significant applications of reconstructing shredded documents in various fields such as the recovery of judicial evidence, the repair of historical documents, the obtaining of military intelligence and so on. Traditionally, the reconstructing process is disposed by hand, which is accurate but efficient. Especially for the too many shredded documents, it can hardly be finished in a short artificially. With the development of the compute technique, people try to reconstruct shredded documents automatically by compute technique to increase the efficiency.

For solving the problems of reconstructing regular shredded documents, we think about dealing with it by computer to achieve it efficiently. Edson Justino[1] had described it by using feature matching, but ours is based on the similarity measure. In this work, we describe the methodology of our idea in the METHODOLOGY section. Then we use our method to solve two kinds of problems. The first one is that the shredded documents are cut in vertical direction. The second one is that the shredded documents are cut in both vertical and horizontal direction. We show the process of solving these two problems in PROCESS section.

II. METHODOLOGY

A. Quantization of Image

The original image is made up by pixel, which can be represented by the additive primary color, namely, red, blue and green. We express the pixel as: I (p)=(R(p), G(p), B(p)), of which R(p) represents the weight of red in the pixel, G(p) represents the weight of green in the pixel and B(p) represents the weight of blue in the pixel.

There is another method to show the pixel, the gray value. The pixel can be quantized ranging from 0 to 255. When it comes to black, the gray value is equal to 0. When it comes to white, the gray value is equal to 255. So a regular shredded documents can be expressed as gray-level matrix as follows:

$$M=\begin{pmatrix} 4 & 0 & 2 \\ 0 & 3 & 1 \\ 0 & 2 & 0 \end{pmatrix}$$

We can simplify the gray-level matrix as two-value matrix[2] by replace the nonzero number with 1.

$$M=\begin{pmatrix} 4 & 0 & 2 \\ 0 & 3 & 1 \\ 0 & 2 & 0 \end{pmatrix} \longrightarrow M'=\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

B. Sparse Similarity

To match the similarity of two-value matrix, we use the sparse similarity[3], whose definition is as follows:

$$S(x_i, x_j)=\frac{b}{b+c}. \quad (1)$$

Of which b is the number that both of the x_i and x_j are equal to 1, a is the number that the value of x_i and x_j are not the same, x_i and x_j are the two-value matrix of two shredded documents.

C. Minimum-error Method

There are two kinds of methods of similarity matching. One is focus on the degree of difference and another emphasizes the degree of similarity. The minimum-error method is the former.

By this method, we slide the template image T on the searching image S. Sum up the absolute difference of the two images in all locations.

$$E(i,j)=\sum_{m=1}^{M}\sum_{n=1}^{N}|S^{i,j}(m,n)-T(m,n)|. \quad (2)$$

The scale of template is $M \times N$ and the smallest $E(i,j)$ means the closest matching locations.

Advantages of the method are: easy algorithm, high speed of calculating, suitable results at the base of easy background.

Disadvantages of the method are: sensitive to the abnormal individual, low reliability, unsuitable for the narrow scale.

D. *Improved Minimum-error Method*

For the minimum-error method, there is an improved version. We set a threshold value T for the formula and change its value dynamically. We consider the two locations are similar when the absolute difference is smaller than T. Sum up the number of similar ones and divided by the total scale to get the similarity. The improved formula is as follows:

$$D(i,j)=\sum_{m=1}^{M}\sum_{n=1}^{N}R(T(m,n),S^{i,j}(m,n))/M\times N. \quad (3)$$

Of which

$$R(T(m,n),S^{i,j}(m,n))=\{^{1,abs(T(m,n)-S^{i,j}(m,n)<T}_{0,other}. \quad (4)$$

The difference between improved minimum-error method and traditional one is that taking the number of similar locations other than the sum of absolute difference of gray value into account. For the threshold value T, the smaller its value is, the more precise the formula is. With the increasing of the threshold value T, the figure of the objective is more intact but the precise decreases.

III. PROCESS

We want to test the accuracy of our similarity measure. So we do two kinds of experiments. The first one is relatively easy. The number of times to match is smaller compared to the second one. The first one is that the shredded documents are cut in vertical direction, while the second one is that the shredded documents are cut in both vertical and horizontal direction. The number of shredded documents is different if the document is of the same size. The second problem must match these shredded documents in both directions but the first problem is no need to do that. Cutting in both directions obviously increase the number of shredded documents. What's more, the gray value for matching is smaller. So it's more difficult. But we can successfully deal with these two problems by our method.

When it comes to the first problem, namely, reconstruct the shredded documents that are cut in the vertical direction. We first cut the document into 19 pieces. Then we try to reconstruct them. So there are totally 19 shredded documents. We display them in *Figure 1* as examples.

First we quantize all the shredded documents as two-value matrix through *Matlab*. We can easily find the left side and right side of the document because it exists margin in both side of the document. So we pick up the one whose two-value matrix are all equal to 1 in the first column as the left side and the one whose two-value matrix are all equal to 1 in the last column as the right side. Then we can match the right side of the first shredded documents with the left side of the rest shredded documents by similarity measure, calculate the sparse similarity and find the biggest one as the neighbor of the first shredded. The rest can be done in the same. Finally, an intact document can be reconstructed. All the similarity of the shredded is showed in the *Figure 2.*

According to the *Figure 2*, we can find the most similar shredded document for each one. Take the 18th shredded document for example, the similarity of the rest 17 shredded document with it are 73, 70, 73, 69, 70, 75, 72, 72, 73, 70, 91, 72, 73, 72, 79, 74 and 69. It is obvious that the biggest similarity of the 18th shredded document is 91. It's the matching result of the 18th one and the 11th one, so they are neighboring. We then just put the 11th one on the right of 18th one. After ensuring the left side, we can recover the document step by step according to *Figure 2*.

When we deal with the shredded documents, which are cut by vertical and horizontal like *Figure 3*, the precise must be increased. So we quantize the shredded documents as gray-level matrix, which means the pixel is ranging from 0 to 255 other than 0 and 1. When it comes to black, the gray value is equal to 0. When it comes to white, the gray value is equal to 255. It is the same to the similarity algorithm, which should be improved. So we replace the sparse similarity by using improved minimum-error method.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0		67	72	66	73	71	95	72		73	74	71	77	70	71	75	71	74	76
1	68		68	69	90	73	72	68		71	73	69	74	69	70	73	73	70	70
2	69	72		74	74	75	70	78		75	73	75	75	73	72	79	96	72	71
3	72	68	77		73	72	73	74		72	92	72	79	72	73	74	73	75	70
4	71	72	74	71		94	72	73		73	73	79	76	71	81	79	74	69	77
5	69	71	74	68	71		68	77		93	70	71	76	72	72	76	74	70	72
6																			
7	72	70	73	66	74	71	75	76		71	72	71	76	71	72	76	74	94	72
8	75	72	71	68	72	79	71	74		71	72	76	74	69	91	77	71	71	78
9	73	70	74	72	71	72	70	75			70	76	73	96	69	76	72	72	71
10	69	70	96	70	69	75	71	76		76		74	77	74	71	74	72	71	71
11	74	73	74	68	71	75	72	94		76	71		76	74	74	79	76	74	76
12	78	70	75	70	74	79	72	78		79	75	80		75	78	96	77	76	77
13	75	71	72	69	70	76	73	75		73	69	78	72		75	80	72	70	94
14	75	70	72	68	78	78	76	78		74	78	76	95	75		80	76	77	74
15	68	68	71	91	67	70	64	68		66	67	71	68	70	66		71	65	68
16	67	94	68	69	67	72	65	74		71	66	70	68	69	71	69		68	71
17	93	69	72	67	69	73	69	76		71	69	75	74	74	75	77	69		74
18	73	70	73	69	70	75	72	72		73	70	91	72	73	72	79	74	69	

Figure 2. Similarity of each two shredded documents[%]

Figure1. Shredded documents

Figure3. A shredded document

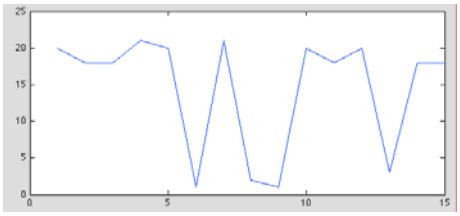

Figure4. Interval

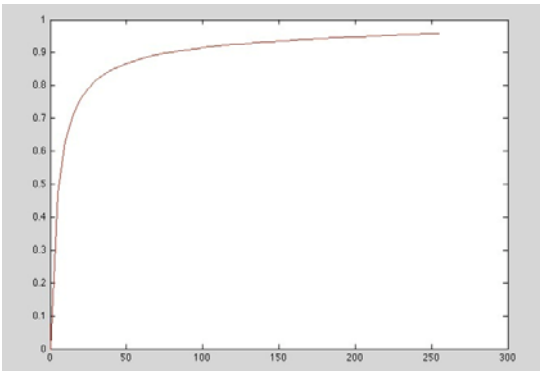

Figure5. Relationship between T and average similarity

First we should determine the left or right side of the document. By analysing the *Figure 3*, we can see that the interval between characters is narrower than the margin. We find the shredded document whose value of last column is all equal to 255. Then we look left to count the column until the one whose value is not all equal to 255. That can look as the interval of the margin. It is the same as the interval between characters. Here are the results of the interval, which are showed in *Figure 4*.

So we can find the right side of the document. But we still can't know the accurate sort of these shredded documents. Next we should to handle the sort in vertical and other sort in horizon.

Using the improved minimum-error method, we must make sure the most appropriate threshold value *T* because it has influence on the precise of the formula. For low contrast image, the threshold value *T* should be set a little smaller. As for high contrast image, the threshold value *T* should be set a little bigger. We use Matlab to make the *T* increase at an even speed to obtain the relationship between *T* and average similarity in order to determine the best *T*. The relationship is showed in *Figure 5*.

From the *Figure 5*, we can see that with the increase of *T*, the average similarity is increasing. So we determine to set threshold value *T* as 255. Getting the most appropriate threshold value *T* can provide much more precise match.

For the smaller shredded document, we can initially construct them in vertical direction. After that, the problem is the same as the shredded documents we have completed like the ones in *Figure 1*.

IV. SUMMARY

In the practical process, when solving the first problem, it can be completed absolutely by computer because these shredded documents are enough long so that their similarity is quite different. But when solving the second problem, we should add some artificial interference. Some shredded documents are extremely special because they are too small and the similarity is almost the same. In this case, we have to judge it artificially. So we combine our similarity measure with artificial interference to achieve reconstructing the document. The reconstructed intact documents are showed in *Figure 6*.

Figure6. Reconstructed intact documents

ACKNOWLEDGMENT

Le Cheng thanks teacher Jia Hongwei for her suggestions and appreciates the school of software engineering of *Beijing University of Posts and Telecommunications*.

REFERENCES

[1] Edson Justino, Luiz S. Oliveria and Cinthia Freitas, "Reconstructing shredded documents through feature matching", Forensic Science International, 2005.

[2] SHI Ting-ting, WU Ming-zhu and CHEN Yong, "Image Retrieval Method Based on Binary Color Co-occurrence Matrix", Computer Engineering, 2011, 37(1), pp. 207-209+212.

[3] ZHAO Ya Qin, ZHOU Xian Zhong, HE Xin and WANG Jian Yu, "An Effective High Attribute Dimensional Sparse Clustering", Pattern Recognition and Artificial Intelligence, 2006, 19(3), pp. 289-294.

[4] LIU Ying, CAO Jian-zhong, XU Zhao-hui, TIAN Yan, FU Tong-tang and WANG Feng, "Improvement of image matching algorithm based on gray correlation", Journal of Applied Optics, 2007, 28(5), pp. 536-540.