Expert Systems With Applications 127 (2019) 35–46

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A solution to reconstruct cross-cut shredded text documents based on constrained seed K-means algorithm and ant colony algorithm

Junhua Chen, Miao Tian, Xingming Qi, Wenxing Wang, Youjun Liu*

College of Life Science and Bioengineering, Beijing University of Technology, No.100 Pingleyuan, Chaoyang District, Beijing 100124, PR China

ARTICLE INFO

Article history:
Received 28 January 2018
Revised 29 January 2019
Accepted 28 February 2019
Available online 1 March 2019

Keywords:
Reconstruction of cross-cut shredded documents (RCCSTD)
Constrained seed K-means algorithm
Horizontal projection
Penalty coefficient
Ant colony algorithm

ABSTRACT

The reconstruction of cross-cut shredded text documents (RCCSTD) is an important problem in forensics and is a real, complex and notable issue for information security and judicial investigations. It can be considered a special kind of greedy square jigsaw puzzle and has attracted the attention of many researchers. Clustering fragments into several rows is a crucial and difficult step in RCCSTD. However, existing approaches achieve low clustering accuracy. This paper therefore proposes a new clustering algorithm based on horizontal projection and a constrained seed K-means algorithm to improve the clustering accuracy. The constrained seed K-means algorithm draws upon expert knowledge and has the following characteristics: 1) the first fragment in each row is easy to distinguish and the unidimensional signals that are extracted from the first fragment can be used as the initial clustering center; 2) two or more prior fragments cannot be clustered together. To improve the splicing accuracy in the rows, a penalty coefficient is added to a traditional cost function. Experiments were carried out on 10 text documents. The accuracy of the clustering algorithm was 99.1% and the overall splicing accuracy was 91.0%, according to our measurements. The algorithm was compared with two other approaches and was found to offer significantly improved performance in terms of clustering accuracy. Our approach obtained the best results of RCCSTD problem based on our experiment results. Moreover, a more complex and real problem – reconstruction of cross-cut shredded dual text documents (RCCSDTD) problem – was tried to solve. The satisfactory results for RCCSDTD problems in some cases were obtained, to authors' best knowledge, our method is the first feasible approach for RCCSDTD problem. On the other hand, the developed system is fundamentally an expert system that is being specifically applied to solve RCCSTD problems.

1. Introduction

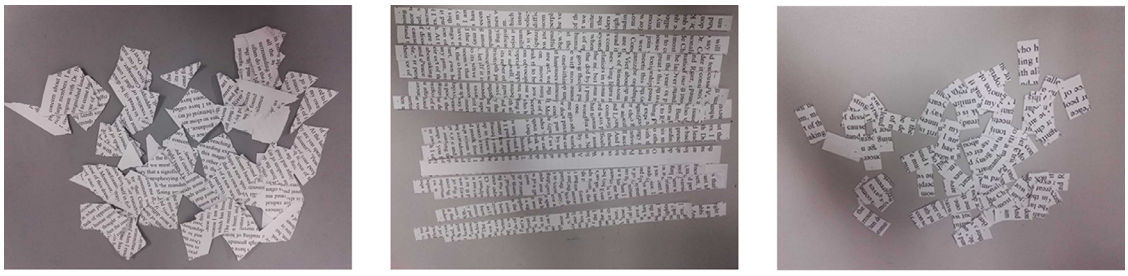
Questioned Document Examination (QDE) is a sub-field of forensic science that is related to law enforcement and the pursuit of federal and civil justice. In order to perform a reliable analysis, a forensic document examiner requires well-preserved documents. However, the documents have often been shredded (accidentally or on purpose) and are in need of reconstruction. Rebuilding a document can be tedious and time-consuming because of the huge number of fragments, some of which may be missing. This underscores the need to develop algorithms capable of accurately reconstructing shredded text documents to meet the requirements of QDE (Justino, Oliveira, & Freitas, 2006).

There are three types of shredded text documents that can be classified according to the way in which they were shredded: manually torn documents; strip shredded documents; and cross-cut shredded documents (see Fig. 1) (Schauer & Prandtstetter, 2010). A great deal of research has already been dedicated to the reconstruction of manually torn and strip shredded documents (De, 2008; Justino et al., 2006; Richter, Ries, Cebron, & Lienhart, 2013; Ukovich & Ramponi, 2008; Xing & Zhang, 2016). In this paper we focus on the Reconstruction of Cross-Cut Shredded Text Documents (RCCSTD).

Most of the research relating to RCCSTD has been published within the last decade. Pomeranz, Shemesh, and Ben-Shahar (2011) proposed a fully-automated greedy square jigsaw puzzle solver and defined a cost function based on boundary color information to measure the dissimilarity between two parts. The accuracy of the solver was between 95% and 100% for some examples (the demonstration image was shredded into more than 2000 shreds). This solver was the state-of-the-art solution for square jigsaw puzzles at the time. The RCCSTD problem can be regarded

* Corresponding author.
E-mail addresses: chenjunhuaemc2@hotmail.com (J. Chen), tianm@emails.bjut.edu.cn (M. Tian), e_shiming@emails.bujt.edu.cn (X. Qi), wangwenxin@emails.bjut.edu.cn (W. Wang), lyjlma@bjut.edu.cn (Y. Liu).

https://doi.org/10.1016/j.eswa.2019.02.039
0957-4174/



(a) A manually torn documents (b) A strip shredded documents (c) A cross-cut shredded documents

Fig. 1. A document shredded by hand (a) and by paper shredders ((b) and (c)).

as a special kind of square jigsaw puzzle and a number of people have borrowed square jigsaw puzzle approaches to solve the RCCSTD problem (Gong, Ge, Li, Zhang, & Ip, 2016; Justino et al., 2006). Schauer et al. (2010) proposed an approach using a genetic algorithm with a (restricted) variable neighborhood search. This was the first research paper to actively propose proceeding in this way. Gong et al. (2016), meanwhile, have proposed a memetic algorithm based on evolutionary algorithms. As a part of this, they defined four key operators together with a comprehensive cost function. Whilst these methods are based on square jigsaw puzzle approaches, it is important to note that there are significant differences between the RCCSTD problem and typical square jigsaw puzzles in terms of the images. In most cases, images of shredded text documents are gray and can therefore be regarded as binary images. At the same time, there is far less boundary information in the individual parts of shredded text documents than there is in the fragments of a normal square jigsaw puzzle. In that case, solvers that are qualified to deal with jigsaw puzzles will not necessarily perform well when it comes to the RCCSTD problem. Sleit, Massad, and Musaddaq (2013) have defined a cost function that is mainly based on black pixels at the border of a fragment's image to measure the cost of pairing two shreds together. This approach reduces the search space and thus improves splicing accuracy. However, the definition of the cost function plays an important role in the splicing. This makes the method highly sensitive in relation to the cost function. Xu, Zheng, Zhuang, and Fan (2014) looked at clustering fragments into several classes based on the features associated with the location of words in the fragments and used a genetic algorithm to solve the splicing problem class by class. This approach divides the RCCSTD problem into several Reconstruction of Strip Shredded Text Document (RSSTD) problems. This, too, serves to reduce the search space and improve the splicing accuracy and is a promising way of solving the RCCSTD problem. However, the clustering algorithm in Xu et al.'s paper lacks power and is not very robust when the text in the fragments is in an abnormal state or the number of fragments is very large. Chen, Ke, Wang, and Liu (2018) have made some improvements to Xu et al.'s method so that it is better able to handle abnormal fragments. They have also introduced two splicing strategies that improve the splicing accuracy for rows. However, the clustering algorithms proposed by Xu et al. (2014) and Chen et al. (2018) are unsuitable for English fragments. A simple summary of the principal advantages and disadvantages of the methods mentioned above is provided in Table 1.

As can be seen from the table, there are two basic ways of tackling the RCCSTD problem. One regards the problem as a regular square jigsaw puzzle and uses methods based on traditional jigsaw puzzle solvers. The key limitation of this approach is that RCCSTD cost functions cannot measure adjacent relations amongst the fragments such as a sudden change of key stroke in the words and this error is amplified by traditional jigsaw puzzle solvers. Another way (using two-stage algorithms) takes more of the characteristics of the RCCSTD problem into account by dividing the whole problem into several sub-problems and then solving these sub-problems independently. Overall, two-stage algorithm based approaches are more promising. In this paper, we therefore propose a new clustering algorithm based on two-stage algorithms that seeks to address the shortcomings of two-stage algorithms outlined in Table 1 (i.e. poor abnormal fragment handling and language-bound feature extraction).

Reconstruction of Cross-Cut Shredded Dual Text Document (RCCSDTD) problems are a natural extension of the RCCSTD problem. In RCCSDTD problems the fragments come from two different documents. As a result there are usually more fragments than would be found in an RCCSTD problem, which makes the problem more challenging. To the best of our knowledge, no feasible approach for tackling this problem has yet been published. However, a solution to this problem is a part of the overall approach we will be outlining here.

In this paper, we propose a new algorithm for tackling the RCCSTD problem and find, during our experiments, that the method is also suitable for dealing with the RCCSDTD problem. Our approach is inspired by certain prior research findings (Lin & Fan-Chiang, 2012; Ng, 2000; Ukovich & Ramponi, 2008; Zhou, 2016) that we shall discuss in more detail as we go along. The reported work pursued the following basic logic: (1) First of all, we project fragments in a horizontal direction and transform the their images into unidimensional signals; (2) We then treat the classification of the fragments as an unsupervised classification problem and use a constrained seed K-means algorithm to cluster the fragments into several rows; (3) After this, we convert the RSSTD problem into a traveling salesman problem (TSP) by using a cost function and introduce a penalty coefficient to modify its operation; (4) Next, we use an ant colony algorithm to solve the TSP; (5) Finally, we use feature matching to merge the reconstructed fragments. A flow chart for the method is shown in Fig. 2.

The main contribution of the paper is the development of a horizontal projection and constrained seed K-means algorithm that can be used to significantly improve clustering accuracy. We report on experiments that confirm that the proposed approach outperforms other approaches to tackling the RCCSTD problem. In light of the effectiveness of our approach, we also used it to try to solve RCCSDTD problems. A number of satisfactory results were also obtained here, making this the first feasible approach for dealing with RCCSDTD problems so far reported. The specific contribution of the paper to expert and intelligent systems is its use of an expert knowledge-based feature extraction scheme and the use of prior knowledge about clustering to develop the constrained seed K-means algorithm. Thus, the developed system is fundamentally an expert system that is being specifically applied to solve RCCSTD problems.

Table 1
Advantages and disadvantages of prior work addressed to the RCCSTD problem.

Methods	Advantages	Disadvantages
Schauer and Prandtstetter (2010)	First method applied	Cannot handle complex situations
Sleit et al. (2013)	The cost function reduces the search space to improve splicing accuracy	Overly-sensitive to the cost function parameters
Gong et al. (2016)	A sophisticated cost function that can measure adjacent relations more accurately than other methods	The splicing accuracy declines sharply as the number of fragments increases
Xu et al. (2014)	A typical two-stage algorithm for the RCCSTD problem that provides a new scheme	The clustering algorithm cannot handle abnormal fragments
Chen et al. (2018)	An improvement on Xu et al. that can deal with more unusual fragments	Cannot cluster English fragments very well because of the feature extraction method

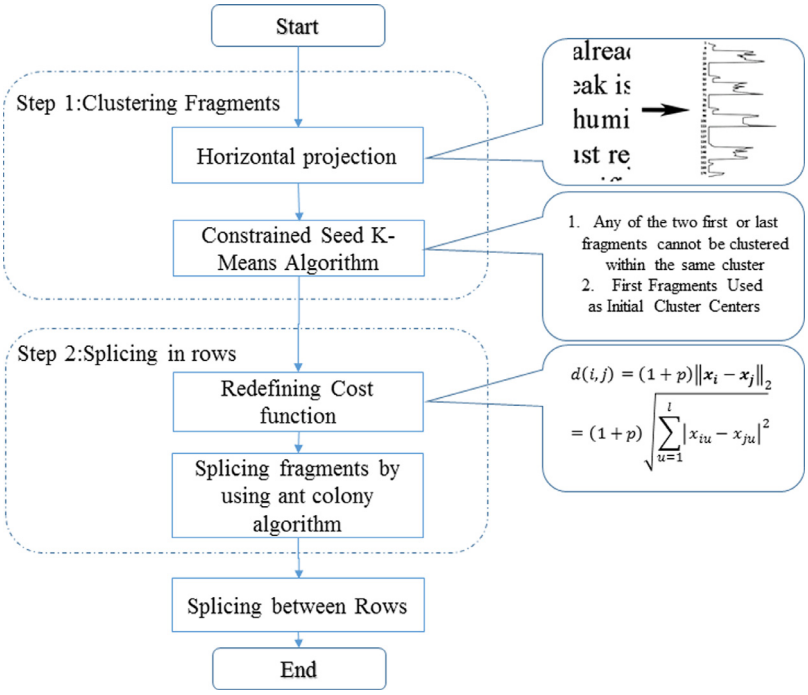


Fig. 2. Flow chart of our method.

The structure of the paper is as follows: Section 2 introduces our proposed method. In Section 3 we present experimental results relating to its application. Section 4 discusses these results and we draw some conclusions in Section 5.

2. Method

In a notable paper by Xu et al. (2014), information about the position of a word in a fragment was used as a basis for clustering. A 4×1 clustering vector was extracted on the basis of such features (see Fig. 3). This vector was then defined as $CV = [a_1, a_2, a_3, a_4]^T$, where a_1 represents the lower position of the unidentified word line. This word line was cut in a horizontal direction, with part of the content appearing at the top or the bottom of the fragment. At the top of the fragment; a_4 represents the upper position of the unidentified word line at the bottom of the fragment; a_2 represents the upper position of the last identified word line at the bottom of the fragment; and a_3 represents the lower position of the last identified word line word at the bottom of the fragment. Clustering vectors like this were used as clustering features in Xu et al.'s paper, with fragments in the same row being clustered as a class and the clustering vector of the first fragments in each row being used as a clustering center.

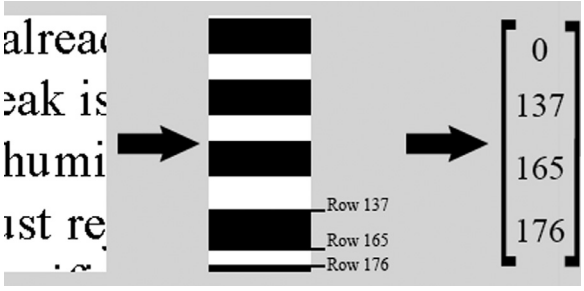


Fig. 3. Xu et al.'s clustering algorithm (2014).

The clustering approach in Xu et al.'s paper is intuitive, easy to implement and obtained good results when applied to their test datasets. However, as they noted themselves, it lacks discriminative power when two or more clusters have the same clustering vector in one shredded document. In addition, the algorithm is unlikely to help with the reconstruction of multiple shredded documents. The main cause of these issues is that the clustering vector is not precise enough to represent a unique fragment. This implies that more feature information needs to be extracted from individual fragments so that there is much greater precision in the row

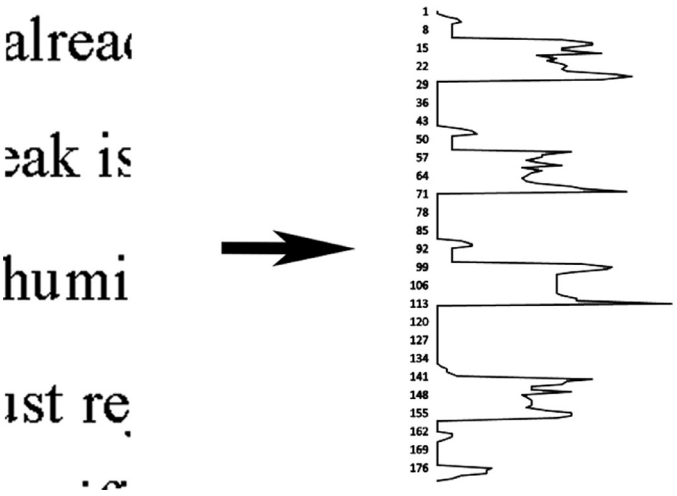


Fig. 4. The horizontal projection of fragments.

clustering. This forms the basic concern against which our own approach has been formulated.

2.1. From project fragments to unidimensional signals

Just four bits of information about the location of a word are included in Xu's clustering vector. In order to use more features from the fragments, however, they are projected fragments in a horizontal direction (after correctly positioning them) and their images are transformed into unidimensional signals (see Fig. 4). The mathematical equation relating to this procedure is as follows:

$$f(y) = \int_0^w I(x, y)dx \tag{1}$$

where $I(x, y)$ represents the gray value of image I at point (x, y) and w represents image I's width. As the image is a digital image, a discrete form of the equation can be expressed as follows:

$$f(y) = \sum_{x=1}^{N} I(x, y) \tag{2}$$

where N represents the horizontal resolution of the digital image.

The principal reason for adopting this procedure is that it enables us to make more use of a fragment's features during classification. A unidimensional signal is a reasonable replacement of a clustering vector in a clustering algorithm.

2.2. Clustering fragments by using a constrained seed K-means algorithm

The signal classification problem posed by the above procedure is basically an unsupervised classification problem. Turney (2002) finds K-means algorithms to be effective at solving unsupervised classification problems. However, we found that a basic K-means algorithm did not deliver good results in clustering experiments. After analyzing the experiments, we came to the conclusion that some prior knowledge needs to be introduced to the basic K-means algorithm to improve the clustering accuracy. This amounts to the following: 1) the first fragment in each row is easy to distinguish (it has many notable features, such as the left side of the fragment's image being white), so the unidimensional signals that are extracted from the first fragment can work as the initial clustering center for the K-means algorithm; 2) two or more last fragments cannot be clustered in the same cluster. This variant of K-means clustering is called a constrained seed K-means algorithm (Wagstaff, Cardie, Rogers, and Schrödl, 2001).

2.2.1. The constrained seed K-means algorithm

As noted above, a constrained seed K-means algorithm is a variant of a K-means clustering algorithm. Its standard description is shown in Table 2 (Wagstaff et al., 2001; Bradley, Bennett, & Demiriz, 2000):

In most RCCSTD problems, the first and last fragment in each row will have a number of notable features. For instance, the left-hand side of the first fragment's image will be white and the right-hand side of the last fragment's image will also be white. So, the unidimensional signal extracted from the first fragment in each row will need to be constrained as a cannot-link fragment (Wagstaff et al., 2001). The same goes for the unidimensional signal extracted from the last fragment in each row. These constraints ensure that several such instances will not be clustered in the same cluster. Those signals that were extracted from the first fragments can then be used as initial clustering centers in the constrained K-means algorithm, thus improving the clustering accuracy (Wagstaff et al., 2001; Ng, 2000).

2.2.2. Clustering the fragments

The m signals that were extracted from the first fragment in each row can be used as m cannot-link constraints (i.e. the document can be considered to have been shredded into $m \times n$ fragments and the size of fragments are equal). The steps involved in clustering the fragments are as follows:

S1. Let the m unidimensional signals (called $\boldsymbol{CV}_1 \ldots \boldsymbol{CV}_m$) be the initial clustering centers (also called 'seed clustering centers').

S2. Use a Euclidean cost function to measure the cost between data point $\boldsymbol{d}_i$ and clustering center $\boldsymbol{CV}_j$. This can be expressed as follows:

$$dist_{ed}\left(\boldsymbol{d}_i, \boldsymbol{CV}_j\right) = \left\|\boldsymbol{d}_i - \boldsymbol{CV}_j\right\|_2 = \sqrt{\sum_{u=1}^{l} \left|d_{iu} - CV_{ju}\right|^2} \tag{3}$$

Where l means the vertical resolution of fragments' image.

S3. Assign data $\boldsymbol{d}_i$ to the closest cluster $\boldsymbol{C}_j$, with the expression of this being:

$$\underset{j}{argmin}\, dist_{ed}\left(\boldsymbol{d}_i, \boldsymbol{CV}_j\right) \tag{4}$$

S4. Update each clustering center $\boldsymbol{C}_i$ according to point $\boldsymbol{d}_i$, which is assigned by averaging all of the points in cluster $\boldsymbol{CV}_i$. The relevant equation for this is:

$$\boldsymbol{CV}_i' = \frac{1}{|\boldsymbol{C}_i|} \sum_{\boldsymbol{d} \in \boldsymbol{C}_i} \boldsymbol{d} \tag{5}$$

S5. Iterate steps (2), (3) and (4) until convergence is achieved.

S6. Return the clusters $\{\boldsymbol{C}_1 \ldots \boldsymbol{C}_K\}$.

We use the n signals that were extracted from the first fragment in each row as the initial clustering centers. Thus, all the fragments that are in the same row are clustered together. In other words, we can classify the fragments into m rows by using the constrained seed K-means algorithm. After clustering, we can transform the RCCSTD problem into m reconstruction of strip shredded text document (RSSTD) problems.

A simple demonstration of the clustering algorithm using unidimensional signals can be found in Appendix I.

2.3. Splicing the fragments in a cluster

Splicing fragments in a row is a special form of the RSSTD problem. In most cases, RSSTD problems can be regarded as a traveling salesman problem (TSP) (Morandell, 2008; Ukovich et al., 2004). More specifically, fragments in an RSSTD problem can be regarded as vertexes in a TSP, with the cost functions among the fragments being regarded as edges. Thus, the methods used to solve TSPs are likely to be useful for RSSTD problems as well. According to Dorigo and Gambardella (1997), ant colony algorithms offer

Table 2
The constrained K-means algorithm.

Constrained K-means Algorithm (data set D, cannot-link constraint $Con_{\neq} \subseteq D \times D$)
1. Let $\boldsymbol{C}_1 \ldots \boldsymbol{C}_K$ be the initial clustering centers.
2. For each data point $\boldsymbol{d}_i$ in data set D, assign data $\boldsymbol{d}_i$ to the closest cluster $\boldsymbol{C}_j$.
3. For each clustering center $\boldsymbol{C}_i$, update it on the basis of point $\boldsymbol{d}_i$, which is assigned by averaging all of the points in cluster $\boldsymbol{C}_i$.
4. Iterate steps (2) and (3) until convergence is achieved.
5. Return the clusters $\{\boldsymbol{C}_1 \ldots \boldsymbol{C}_K\}$.

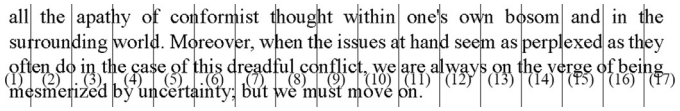


Fig. 5. Fragments in the same row with different lines of words.

an effective and highly accurate approach to solving TSPs. We are therefore using an ant colony algorithm here to solve the separate RSSTD problems that together make up the overall RCCSTD problem.

We will treat the cost function matrix of each of the fragments in the same cluster as an adjacency matrix in a TSP. Thus, the cost function between two fragments $d(i, j)$ can be defined as follows:

$$d(i,j) = (1+p)\left\|\boldsymbol{x_i} - \boldsymbol{x_j}\right\|_2 = (1+p)\sqrt{\sum_{u=1}^{l}\left|x_{iu} - x_{ju}\right|^2} \tag{6}$$

where, $\boldsymbol{x_i}$ is the vector of fragment i's right boundary; $\boldsymbol{x_j}$ is the vector of fragment j's right boundary; and x_{iu} and x_{ju} are the values of $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ at position u. The cost function is based on Euclidean distance and p is a penalty coefficient. p can be defined as follows:

$$p = \frac{M}{\sum_{u=1}^{l} x_{iu} + \sum_{u=1}^{l} x_{ju}} \tag{7}$$

The penalty p and the content information are inversely proportional. Thus, M is a constant in the inversely proportional function that can be determined by experiments. We have introduced the penalty coefficient p to modify the errors that are caused by variation in the amount of information about a fragments' boundary. p is able to correct the cost function because of the absence of boundary information (there are too few black pixels and too many white ones) rather than because the two fragments are similar.

The need to introduce a penalty coefficient in the cost function will be discussed further in Section 4.

Having acquired the adjacency matrix that is needed for the specification of a TSP, an ant colony algorithm is used to solve it. The definition of the cost function is one of the sources of splicing errors, so, ideally, it would be best to find a better definition of the cost function to measure the adjacent correlation between the fragments. However, it seems to be impossible to find a perfect scheme to measure the adjacent correlations because of the finite boundary information.

As a result, we have adopted both a combination strategy and a divide-and-conquer strategy to increase the splicing accuracy in a row (which is equivalent to a cluster). The combination strategy involves specifying that, whenever the cost function of two fragments is less than a certain threshold, the two fragments will be merged into one fragment to reduce the number of fragments in a row. The divide-and-conquer strategy involves first of all recognizing that, where the fragments in the same row relate to different lines of words, this will increase the splicing error if the same parameters are used (see Fig. 5, where fragments 1–11 have four lines of words. but there are only three lines of words in fragments 12–17). As a result, it is necessary to divide these fragments into two parts and handle them individually. More detail about these two strategies can be found in [2].

2.4. Merging the reconstructed fragments

When the splicing of the fragments in a row is finished, splicing between the rows is much easier. The steps required for interrow splicing are as follows:

S1. Extract the clustering vector (called $\boldsymbol{CVC}_i = [a_1, a_2, a_3, a_4]$) from each row fragment according to the approach adopted by Xu et al. (2014), as shown in Fig. 3.

S2. Determine the first row of fragments by locating those with the feature of the top border being white. Now determine its clustering vector, $\boldsymbol{CVC}_F$. We can let $\boldsymbol{CVC}_C = \boldsymbol{CVC}_F$, where $\boldsymbol{CVC}_C$ is the current clustering vector being matched .

S3. If the clustering vector $\boldsymbol{CVC}_i$ meets the conditions $l - \delta < a_4^c + a_1^i < l + \delta$ or $nl + l' - \delta < a_3^c + a_2^i < nl + l' + \delta$, we can let $\boldsymbol{CVC}_C = \boldsymbol{CVC}_i$ and merge fragment i, where a_4^c represents a_4 of the current clustering vector and a_1^i represents a_1 of the clustering vector i, l represents the height of the word and l' represents the height of interline space, n is an integer (n greater than or equal to 0 less than the number of identified word lines in this row), and δ is a number of error range, we set as 3 in our experiments.

S4. Repeat S3 until the splicing is finished.

3. Experiments and results

3.1. Experiments and results based on simulated data

In order to examine the viability and splicing accuracy of the proposed algorithm, we needed to simplify the problem by eliminating any interference arising from other factors. There are many factors that can influence splicing accuracy beyond just the algorithm itself. These can include such things as: the loss of paper that is turned to dust by the shredding knives; the skew of the cuts relative to the lines of text; the resolution of the scanner; noise in the image arising from the scanning process; and so on. Bearing this in mind, we created a test dataset by using a digital simulation of a physical cross-cut shredder. The resolution of the text document's image was 1980 × 1224.

Our test data consisted of a set of 10 text documents shredded into 11 × 17 shreds by the computer (5 English text documents in Times New Roman 10 font with exactly 13 point line spacing and 5 English text documents in Bold Times New Roman 10 font with exactly 13 point line spacing). The size of the photos was 180 × 72 pixels, the height of the words in the photos was about 26 pixels and the interline space height was about 29 pixels. All of the algorithms were implemented in C++ and the tests were all performed on a Core i7 4790 CPU with 4GB RAM.

3.1.1. Clustering analysis and comparison

3.1.2.1. Evaluation method. To measure the accuracy of a clustering algorithm we focus upon precision, which is a widely used concept in the machine learning domain. Precision, in this case, can be defined as follows: for multi-classification problems, samples can be divided into 4 classes according to whether they can be labeled

Table 3
Confusion matrix for the multi-classification problem.

True label	Predicted result from the classifier	
	ω_i	$\widetilde{\omega}_i$
ω_i	$TP = c_{ii}$	$FN = \sum_{l=1,l\neq i}^{m} c_{il}$
$\widetilde{\omega}_i$	$FP = \sum_{k=1,k\neq i}^{m} c_{ki}$	$TN = \sum_{k=1,k\neq i}^{m}\sum_{l=1,l\neq i}^{m} c_{kl}$

true and the predicted result from a classifier. Thus, samples can be: True Positive; False Positive; True Negative; or False Negative (Townsend, 1971). *TP, FP, TN, and FN* can be used to represent the various samples, respectively. The confusion matrix for the multi-classification problem is shown in Table 3.
where ω_i represents the cluster i, $\widetilde{\omega}_i$ represents all of the clusters except cluster i, c_{ii} represents the number of samples where ω_i is given a true label and clustered into ω_i by the classifier, and c_{il},c_{ki},c_{kl} have a similar meaning to c_{ii}.

For the purposes of this paper, precision is defined as follows:

$$P = 1 - \frac{FN + FP}{TP + FN + FP + TN} \tag{8}$$

3.1.2.2. Comparison of results. Table 4 shows the clustering results obtained by applying Xu et al.'s clustering vector algorithm, a basic K-means algorithm, and our own constrained seed K-means algorithm to the test datasets mentioned above (datasets 1–5 are in Times New Roman 10 font with exactly 13 point line spacing and datasets 6–10 are in Bold Times New Roman 10 font with exactly 13 points line spacing). We calculated the precision attained by each of the three methods in terms of False Negatives and False Positives.

The clustering results for each algorithm were compared with an independent t-test. The test results showed that there was no significant difference ($p = 0.4032$) between the clustering vector approach and the basic K-means algorithm. Our own approach, on the other hand, was significantly better than the others ($p < 0.01$). The results of the t-test reveal that a K-means algorithm on its own cannot significantly improve the clustering accuracy, so it takes the features present in our own approach to capture the characteristics of the fragments missing in Xu et al.'s approach. We can therefore conclude that our approach displays much better performance than the others in terms of clustering accuracy.

3.1.2. Splicing accuracy

As can be seen from Table 4, our clustering algorithm did not cluster all of the test documents correctly (there were 28 clustering errors in total). If these errors were to remain uncorrected, there would be redundancies in some rows and deficiencies in others and the splicing results would look bizarre. In order to validate the splicing accuracy of our method, we needed to modify the clustering errors shown in Table 4 manually (because of the small number of errors, this was an easy task). We divided the RCCSTD problems into several RSSTD problems. This gave us 11 RSSTD problems for every RCCSTD problem. As presented above, we transformed the RSSTD problems into traveling salesman problems and used an ant colony algorithm to solve them. In accordance with the procedure proposed by Dorigo and Gambardella (1997), we set the parameters of the ant colony algorithm to $\alpha = 1$, $\beta = 5$, $\rho = 0.5$, and $n = 17$. We used the similarity measurement proposed by Sleit et al. (2013) to measure the splicing accuracy. As the accuracy of the splicing between the rows was very high, the key concern was the splicing accuracy within the rows themselves. There were 17 pairs in a row. We accumulated the number of error pairs in every row and calculated the splicing accuracy as follows:

$$\begin{aligned} Accuracy &= 1 - \frac{the\ total\ number\ of\ error\ pairs\ in\ rows}{the\ number\ of\ pairs\ in\ rows} \\ &= 1 - \frac{the\ total\ number\ of\ error\ pairs\ in\ rows}{17*11} \end{aligned} \tag{9}$$

In order to improve the splicing accuracy in a row, the combination and divide-and-conquer strategies outlined in Section 2.3 were used.

The number of error pairs and the splicing accuracy for every test dataset are listed in Table 5 and the reconstruction results for dataset 1 and dataset 6 are shown in Fig. 5. The original documents relating to datasets 1 and 6 are shown in Fig. 7.

The splicing accuracy for the two kinds of test data were compared with an independent t-test. It can be seen from Table 5 that there was a significant difference in the splicing accuracy for the two kinds of documents (93.4% ± 2% vs 88.6% ± 3.4%, $p = 0.0038$). This suggests that the cost function defintion proposed in this paper is sensitive to whether text is highlighted in bold.

The mean splicing accuracy across all of the test data was 91.0% according to our measurements.

3.1.3. Reconstruction

In view of the high precision demonstrated by our clustering algorithm, we decided to try and solve an RCCSTD problem with fragments from two documents rather than one (i.e. an RCCSDTD problem). We therefore selected two random test documents from those listed in Table 4 as a series of 10 new test datasets. In each case, the two documents were shredded into 11×17 shreds by a computer and the fragments were mixed together randomly. The results of these 10 clustering experiments are shown in Table 6.

It can be seen from Table 6 that the mean clustering accuracy across all of the test data was 90.8%, with the clustering accuracy remaining quite high when different kinds of documents were mixed. This means that the proposed approach is able to treat RCCSDTD problems in the same way as normal RCCSTD problems, even if there are different kinds of documents. Thus, we can get same results mentioned in Section 3.1.2 when we mix test datasets 1 and 6 together randomly (see Fig. 6). To the best of our knowledge, solutions to the RCCSDTD problem have not yet been reported. As our method obtains satisfactory results for RCCSDTD problems (at least, in some cases) it may represent the first feasible approach to tackling this problem

3.2. Experiments and results based on real data

In order to test the capacity of our algorithm to deal with real shredded data, we used a genuine shredded document as the basis of a new test dataset. The steps involved in dealing with the real data were as follows:

S1. We printed the original document for test dataset 1 on A4 paper using an HP LaserJet M4345 MFP printer and shreded this document into 11 × 17 shreds.

S2. We scanned the fragments using a KONICA MINOLTA bizhub 16 scanner with an optical resolution of 300 dpi. In order to simplify the segmentation process, background paper with a colour significantly different to black and white was used (the colour had an RGB value of (54,186,207)). The scanned image is shown in Fig. 8(a).

S3. We converted the RGB image into a gray image and used an Otsu threshold algorithm (Otsu, 1979) to segment the scanned image into fragments (a covering of blue paper was used to improve the segmentation accuracy) (see Fig. 8(b)).

S4. We took the image of each fragment from the segmented image and coded them randomly from 000–187. In order to place

Table 4
Results of classification accuracy for three different algorithms.

Methods	Clustering Vector (Xu et al., 2014)		Basic K-means algorithm		Method used in this paper	
Test instances	FN	FP	FN	FP	FN	FP
Test data 1	16	16	17	17	0	0
Test data 2	7	7	24	24	0	0
Test data 3	43	43	1	1	0	0
Test data 4	29	29	20	20	0	0
Test data 5	31	31	25	25	0	0
Test data 6	45	45	19	19	0	0
Test data 7	17	17	26	26	1	1
Test data 8	17	17	31	31	9	9
Test data 9	29	29	15	15	0	0
Test data 10	6	6	19	19	4	4
Precision	74.3%	78.9%	99.1%			

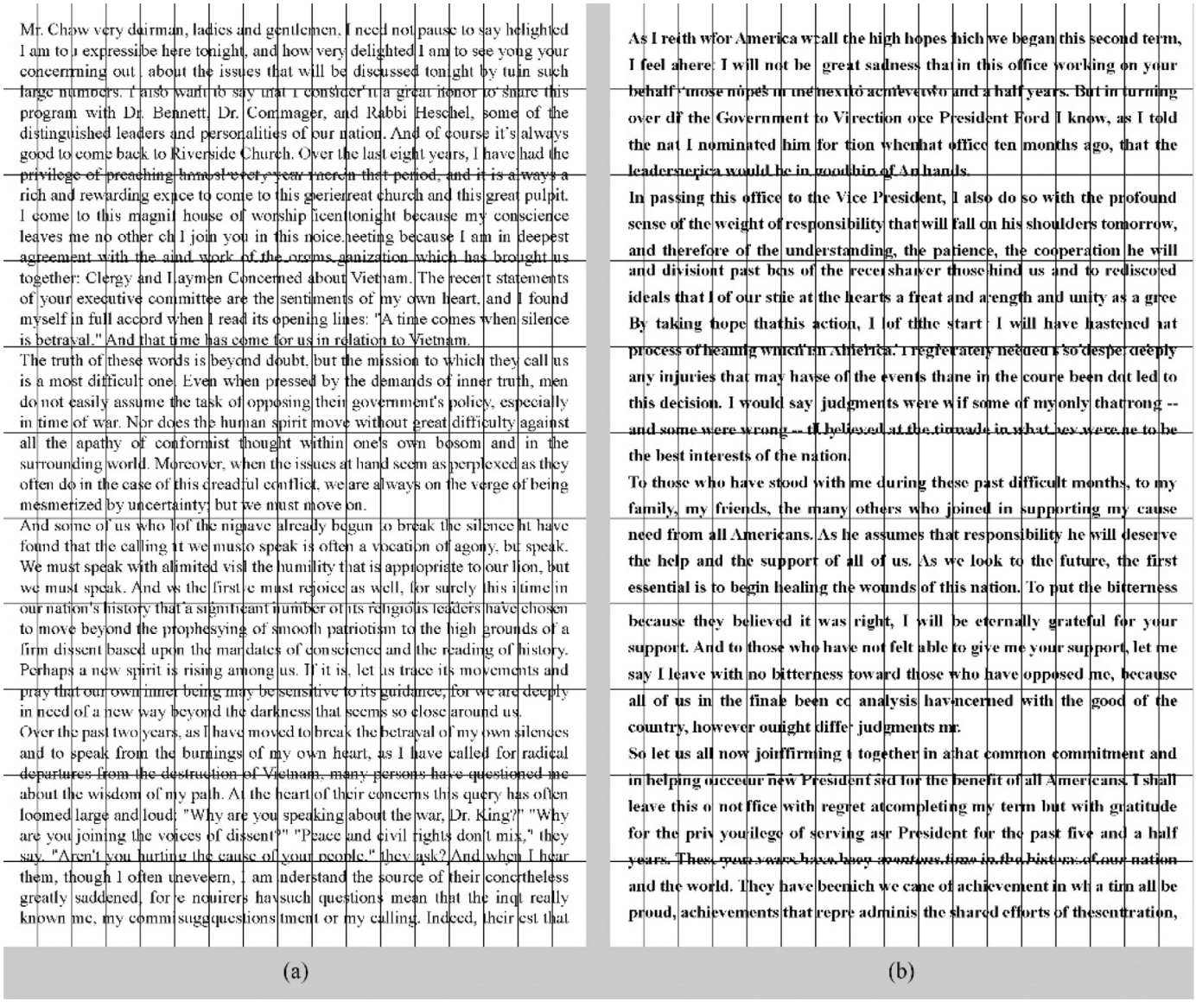


Fig. 6. The reconstructed documents for test dataset 1 (a) and for test dataset 6 (b).

the images' edges in a normal position, a further step (rotation) was required to fix the skew before performing the projection. The specific rotation angle was found by using a Radon transform. The horizontal resolution of the fragments' images ranged from 135 to 141 and the vertical resolution ranged from 345 to 349. We resized all of the images to 138 × 349.

S5. We used the feature extraction scheme and clustering algorithm that was proposed earlier in this paper to cluster the fragments into 11 rows. We found that our method was able to cluster these real data fragments correctly, with a clustering accuracy of 100%.

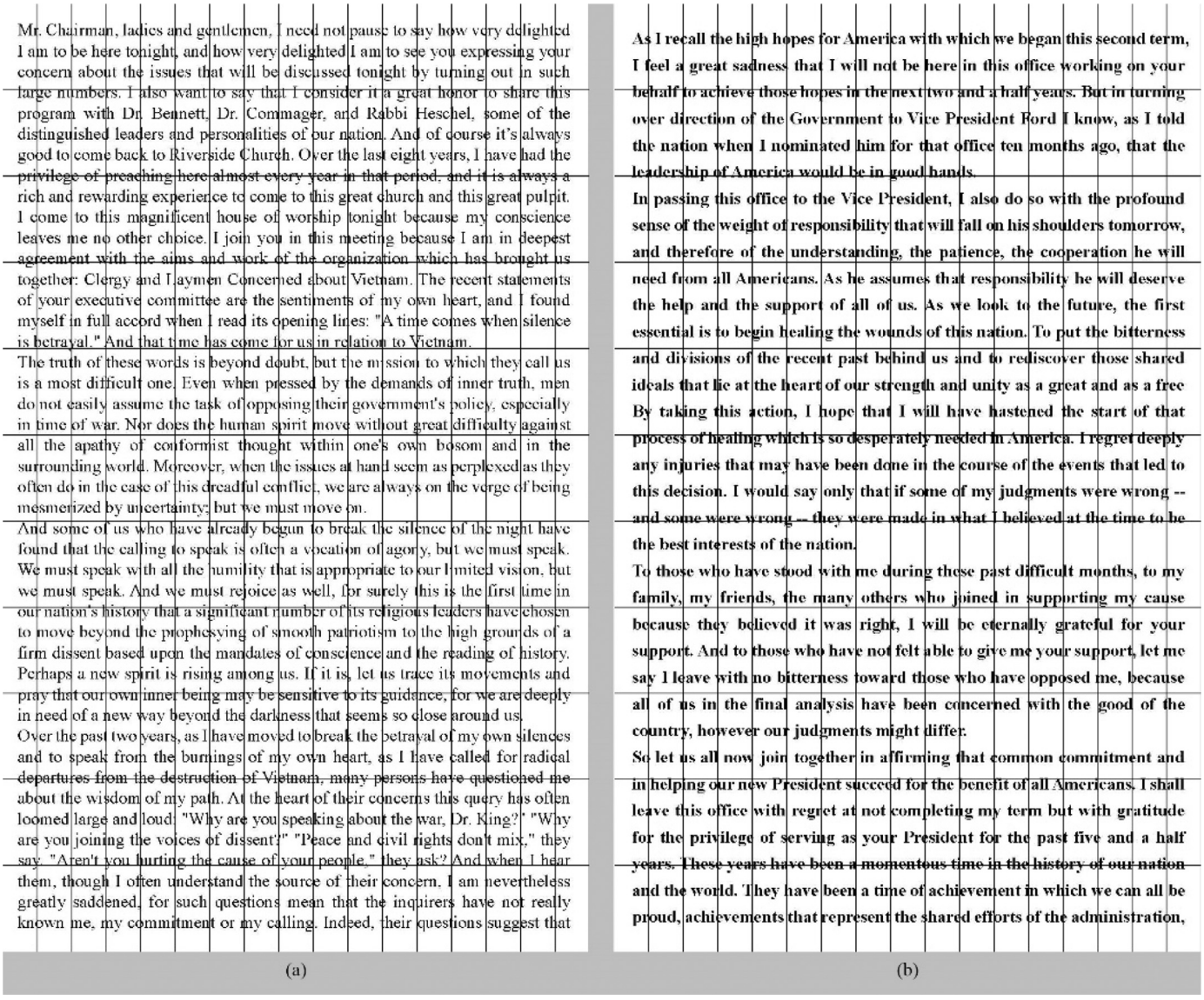


Fig. 7. The original documents for test dataset 1 (a) and test dataset 6 (b).

Table 5
Splicing accuracy for every test dataset (T6-T10 in bold font and others are no).

Test data	T1	T2	T3	T4	T5	Total
Errors	10	13	11	16	12	62
Accuracy	94.7%	93.0%	94.1%	91.4%	93.6%	93.4%
Test data	T6	T7	T8	T9	T10	Total
Errors	24	23	19	15	26	107
Accuracy	87.2%	87.7%	89.8%	92.0%	86.1%	88.6%

Table 6
Clustering results for RCCSDTD problems.

Test data	T1&T6		T2&T8		T2&T9		T4&T9		T5&T9	
	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP
Errors	0	0	8	8	0	0	17	17	9	9
Accuracy	100%		97.2%		100%		94.1%		96.9%	
Test data	T1&T2		T2&T3		T2&T4		T7&T9		T8&T10	
	FN	FP	FN	FP	FN	FP	FN	FP	FN	FP
Errors	34	34	56	56	32	32	48	48	62	62
Accuracy	88.1%		80.6%		88.9%		83.4%		78.5%	

S6. We again used our proposed method to splice the fragments in a cluster and between the clusters. The results are shown in Fig. 9(a) (the overall result was reszied to 1980 × 1224). We then compared the results using real data shown in Fig. 8 with the prior results shown in Fig. 6(a).

It can be seen from Fig. 9 that there were 14 differences in three rows between the real data and the simulated data. We have marked these differences in Fig. 8(a) with ellipses. The number of error pairs for the real data was 22 and the accuracy, based on the equation presented in Section 3.1.2, was 88.2%. We consider it likely that these differences mainly originate from the loss of paper that was turned to dust by the shredding knives and the image re-scaling that was conducted in Step 4 above.

On the basis of our experiments, we found that our clustering algorithm significantly outperformed the other methods mentioned in Table 1. One of reasons for this may be that the feature vector at the root of the horizontal projection can represent fragments better than the four-dimensional feature vector used by Xu et al. Another important element is the expert knowledge that formed the basis of the constrained seed k-means algorithm. This changed the problem from being an unsupervised clustering

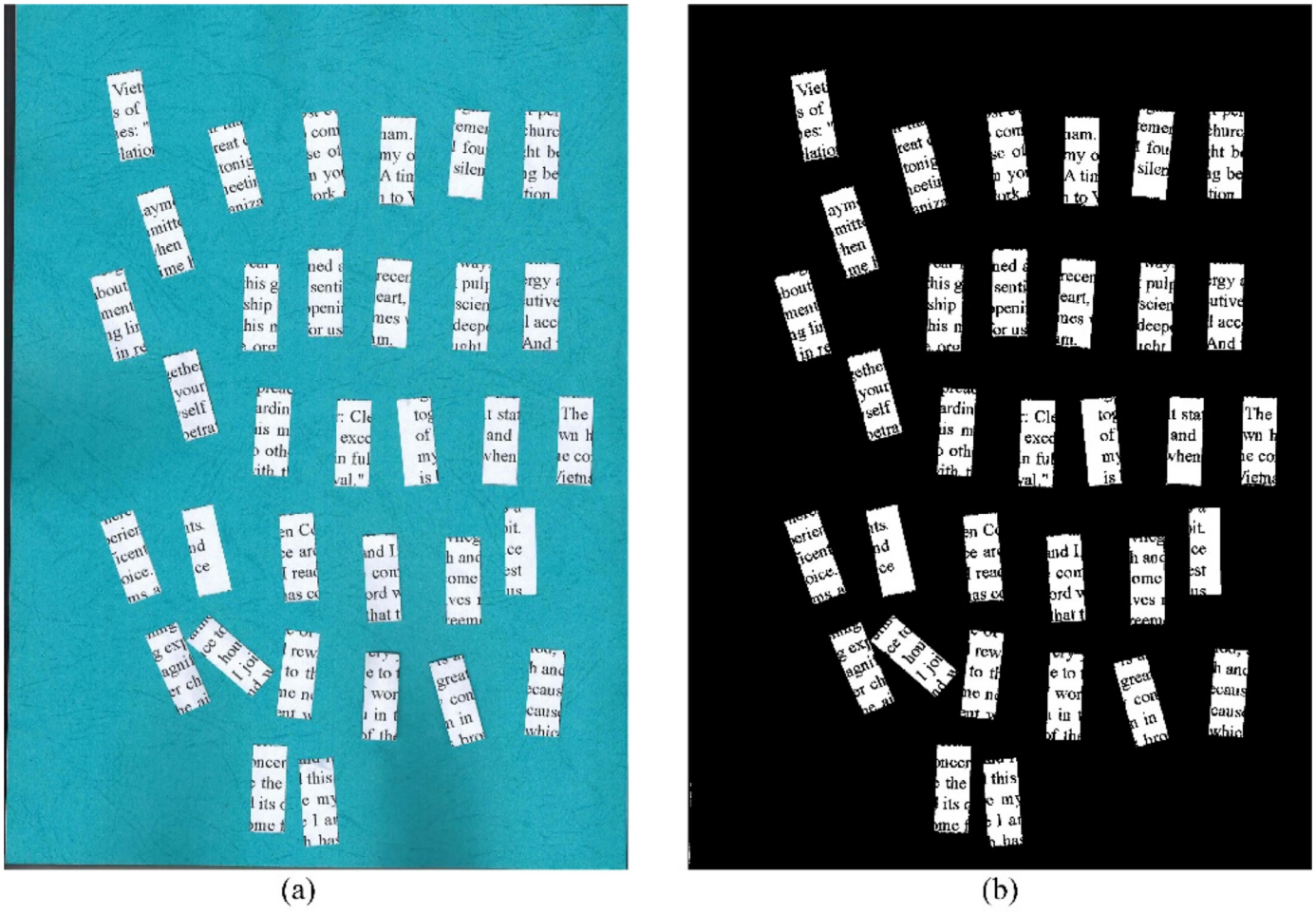


Fig. 8. The scanned image (a) and the segmented image (b).

problem to a weakly-supervised clustering problem, so it may have played an important role in improving the clustering algorithm. When it comes to the splicing results, a cost function still plays an irreplaceable part in the construction of the two-stage algorithm. After clustering, it was possible to divide the RCCSTD problem up into several RSSTD problems. However, there is much less boundary information associated with the fragments than would be the case with a regular RSSTD problem. Therefore, the error between the cost function and real adjacent relations remains a source of splicing mistakes. As we can see from the differences between the results when using real shredded fragments and simulated fragments, there are many factors can affect splicing accuracy, including resizing of the segmented images, loss of paper (and thus context) because of it being turned to dust by the shredding knives, and so on.

4. Discussion

As mentioned above, in this section we will discuss the necessity of introducing a penalty coefficient p to the cost function in RCCSTD algorithms. The main reason for introducing a penalty coefficient is to modify any errors originating from the definition of the cost function. During our experiments, we found that there was occasionally a small Euclidean distance between two fragments resulting from links in their boundary information rather than them having a strong adjacent correlation. In Fig. 10, the Euclidean distance between fragment i's right boundary and fragment j's left boundary is 2.82, which is very small. This might be taken to suggest that fragment i has a strong adjacent correlation with fragment j, but this isn't the case. The question therefore arises as to how deal with this kind of error.

In our view, the cost function should be linked to the content of a fragments' boundary information. For this reason, we have introduced a penalty coefficient as a means of correcting the cost function. This penalty coefficient can be defined as follows:

$$p = \frac{M}{\sum_{u=1}^{l} x_{iu} + \sum_{u=1}^{l} x_{ju}} \tag{10}$$

where M is a parameter that can be determined experimentally. In our own case, we set M to be twice the average number of pixels in a row that had a pixel value of zero during our experiments.

Let us take Row 10 from test dataset 1 to demonstrate the role of a penalty coefficient in the RCCSDT problem. As shown in Fig. 11(a), there are 7 pairing errors. However, there are no pairing errors in Fig. 11(b) (the parameters of the ant colony algorithm were set to be $\alpha = 1$, $\beta = 5$, $\rho = 0.5$, $n = 17$ and parameter M was set to 25). This is not an incidental phenomenon: introducing a penalty coefficient to the cost function was able to reduce the pairing errors in 10 rows without any negative effect upon the other rows. Overall, its use reduced the number of pairing errors from 35 to 10 in test dataset 1.

To conclude, the introduction of a penalty coefficient p to the cost function in an RCCSTD problem is a necessity.

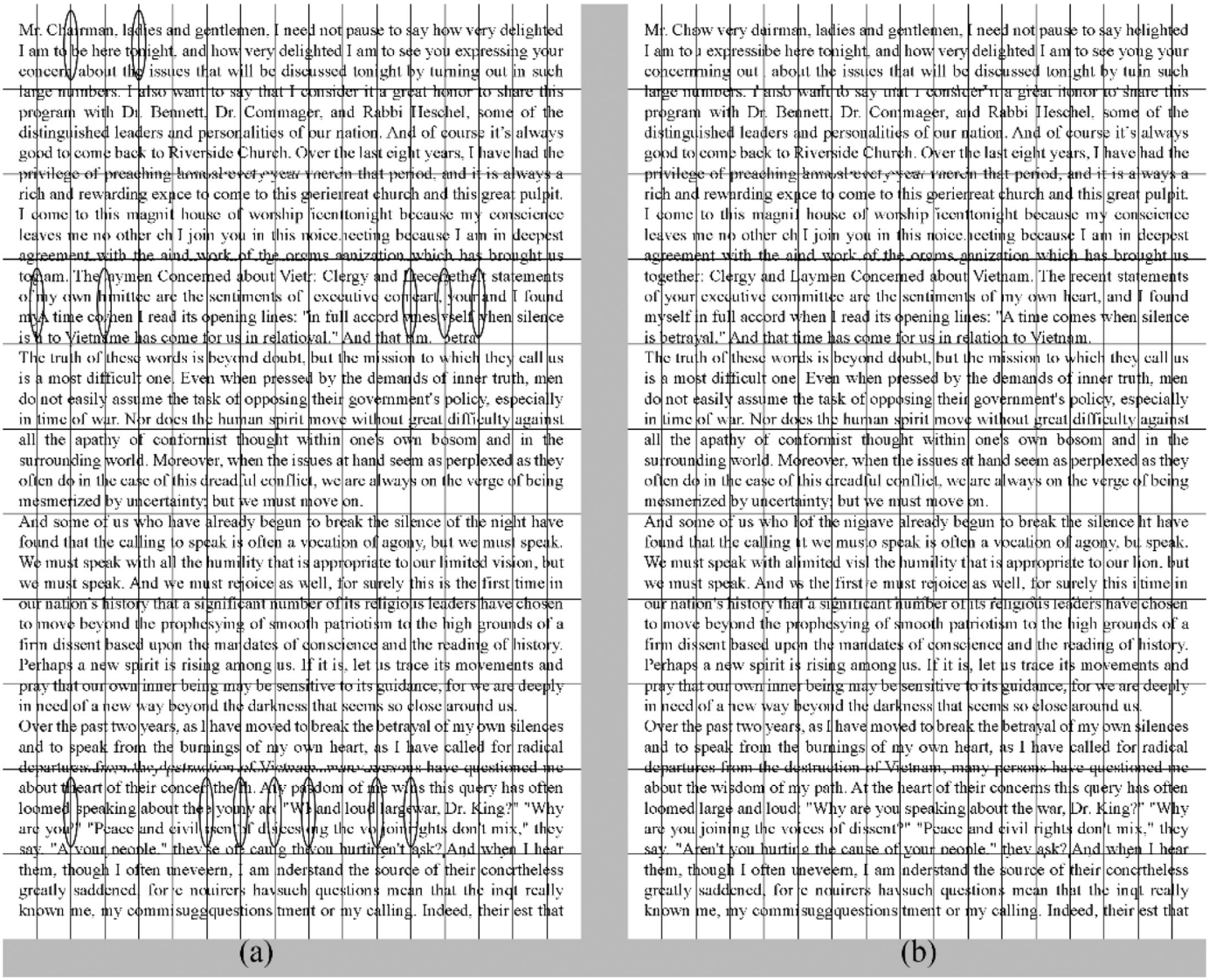


Fig. 9. The document reconstructed from the real dataset (a) and the document reconstructed from test dataset 1 (b).

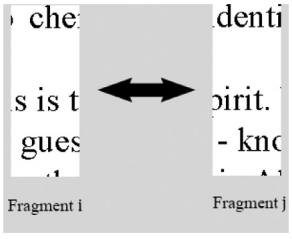


Fig. 10. A demonstration of the kind of error that can arise from the definition of the cost function.

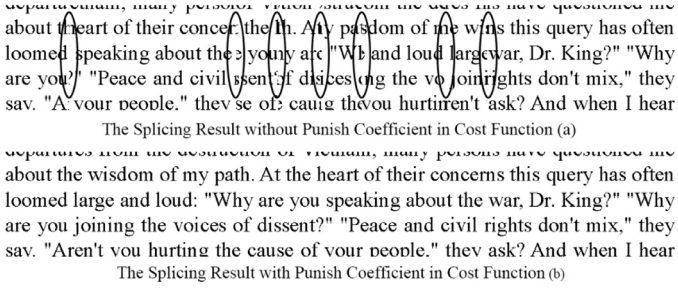


Fig. 11. Comparison of the splicing results with and without a penalty coefficient being included in the cost function.

5. Conclusion and future work

The clustering accuracy produced by the RCCSTD algorithm proposed in this paper was 99.1%, based on 10 RCCSTD scenarios. A comparison was made between the proposed algorithm and two other clustering algorithms. The results showed that the proposed algorithm offers significantly improved performance in terms of clustering accuracy. It was also found that introducing a penalty coefficient to the cost function in the RCCSTD algorithm can improve the splicing accuracy. The mean splicing accuracy across all of the test data was 91.0%, based on our measurements. In view of the high precision offered by our clustering algorithm, we attempted to use it to solve a reconstruction of cross-cut shredded dual text documents (RCCSDTD) problem. It performed well, especially when the fragments came from different kinds of text documents. The mean clustering accuracy for the RCCSDTD problem was 90.8% and, in some cases, the splicing accuracy was over 95%, so our method produced some promising results. When it came

to dealing with real shredded documents, the clustering algorithm still worked efficiently, but the splicing accuracy decreased. However, it should be noted that the proposed algorithm does have some issues that need to be borne in mind.

With reference to the summary of related works presented in Table 1 and the following analysis, our algorithm would be classified as a two-stage algorithm. In comparison to other related expert and intelligent systems designed to address the RCCSTD problem, our approach offers some advantages. Specifically, if it is compared to methods that consist of global optimization and the use of an expert knowledge-based cost function, such as Schauer & Prandtstetter, 2010; Sleit et al., 2013; and Gong et al., 2016, our approach is better able to reduce the effect of errors arising from the measurement of adjacent relations. Our approach is also able to solve larger problems. The number of fragments was no more than 80 in the papers cited, while our approach was tackling up to 374 fragments. If our approach is compared to other two-stage algorithms (e.g. Chen et al., 2018; Xu et al., 2014), on the one hand, our feature extraction scheme is able to handle more features and is more robust. On the other hand, our expert knowledge-based clustering algorithm can significantly improve clustering accuracy and is able to handle both Chinese and English text documents. On top of this, to the best of our knowledge, our method offers the first feasible solution to the RCCSDTD problem. Overall, then, our approach can be considered a better application of expert systems to the RCCSTD problem than other approaches to date.

Admittedly, there are still many weaknesses in our algorithm. First of all, the paper shredders used in places like banks will shred documents into more than 1000 fragments and it is quite normal to shred more than two documents at once. As it stands, our algorithm would not be able to cope with these kinds of situations. Secondly, the first fragment of each row still requires initial identification. Having said this, there are certain cases where this may not apply, for instance if the initial clustering center is designated randomly, though the clustering accuracy will decrease. Thirdly, we are currently setting aside a number of factors that can affect splicing accuracy, such as abnormal printing, inclined cutting of the paper, dust from the blades, and so on. Fourthly, there are currently two approaches to evaluating the splicing accuracy for RCCSTD problems: one, proposed by Sleit in 2013, is based on the number of error pairs; the other, proposed by Xu in 2014, is based on the number of places where there are errors. In this paper, we adopted the first approach. However, there is no unitary standard for this problem and this can make horizontal comparison between methods more difficult. In other words, our splicing accuracy cannot be compared with Xu et al.'s approach directly. Finally, the cost function relating to the fragments in a row remains a significant bottleneck that is preventing the further improvement of splicing accuracy.

In our future work, the cost function will be the primary focus of our concern, because it limits the splicing accuracy. Until now most cost functions have been based on expert knowledge. However, a deep learning-based cost function may be a good solution. More specifically, Recurrent Neural Networks (RNN) are likely to be well-suited to our situation. The boundary information can be used as the input feature vectors in a RNN and the scorer that assesses the confidence score for the merging of two words in the RNN can be used as the cost function. In most real cases, some fragments of a document will be lost and some people have focused upon tackling this using regular square jigsaw puzzle approaches (Paikin & Tal, 2015). However, this has not yet been formulated as a part of the RCCSTD problem. In that case, reconstruction of cross-cut shredded text documents with missing pieces will also require research in the future. There are many factors that can affect the splicing accuracy of RCCSTD algorithms, including the available information in the fragments, the use of different languages, and so on, so further work is needed to build assessment of these factors into the algorithms. A further, interesting variant of the RCCSDTD problem is where two adjacent pages from the same document have been shredded and scanned. Here, clustering the fragments is easier than the RCCSDTD problem we discussed above, but splicing the fragments in the rows is harder. Finally, tireless effort needs to be devoted to arriving at solutions to the RCCSTD problem that can actually be applied in real-world practice.

Author contributions

Junhua Chen, designed the research project; Junhua Chen, Miao Tian and Wenxin Wang performed experiments; Wenxin Wang and Xingming Qi analyzed data and interpreted results; Junhua Chen, Miao Tian and Xingming Qi prepared figures and drafted manuscript; Youjun Liu supervised the project and approved final version of manuscript.

Acknowledgments

The authors would like to thank the reviewers for their constructive and thoughtful comments. This research was supported by the National Natural Science Foundation of China (11832003, 11772016, 11472022) and the key project of science and technology of Beijing Municipal Education Commission (KZ201810005007). The authors thank Editsprings (www.editsprings.com) for its linguistic assistance.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.eswa.2019.02.039.

Appendix I. A simple demonstration of how the clustering algorithm can be applied

In order to demonstrate the procedures used in our clustering algorithm, we randomly selected 6 fragments from 2 rows (including the first fragment of the two rows) as our demonstration data (see Supp. Fig. 1). The actual labels for Initial Center 1, Initial Center 2, Test Fragment 1, Test Fragment 2, Test Fragment 3 and Test Fragment 4 were 1,2, 1, 1, 2 and 2, respectively.

As mentioned above, the fragments needed to be positioned in the right way before performing the projection. After that, we scanned each row of the fragment's image horizontally to calculate the number of black pixels in the row (see Supp. Fig. 2). From this, we obtained a 180-dimensional vector (unidimensional signal) for each fragment. We selected 10 components from the 180-dimensional vector for the convenience of demonstration. The number of black pixels in these rows is shown in Supp. Fig. 2 (based on Initial Center 1).

References

Bradley, P. S., Bennett, K. P., & Demiriz, A. (2000). Constrained k-means clustering. *Microsoft Research, Redmond*, 1–8.

Chen, J., Ke, D., Wang, Z., & Liu, Y. (2018). A high splicing accuracy solution to reconstruction of cross-cut shredded text document problem. *Multimedia Tools and Applications, 77*(15), 19281–19300.

De, S. P. (2008). Reconstruction of ripped-up documents using fragment stack analysis procedures. *Forensic Science International, 176*(2–3), 124.

Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation, 1*(1), 53–66.

Gong, Y. J., Ge, Y. F., Li, J. J., Zhang, J., & Ip, W. H. (2016). A splicing-driven memetic algorithm for reconstructing cross-cut shredded text documents. *Applied Soft Computing, 45*, 163–172.

Justino, E., Oliveira, L. S., & Freitas, C. (2006). Reconstructing shredded documents through feature matching. *Forensic Science International, 160*(2–3), 140.

Lin, H. Y., & Fan-Chiang, W. C. (2012). Reconstruction of shredded document based on image feature matching. *Expert Systems with Applications, 39*(3), 3324–3332.
Morandell, W. (2008). *Evaluation and reconstruction of strip-shredded text documents*. Austria: Vienna University of Technology.
Ng, M. K. (2000). A note on constrained k-means algorithms. *Pattern Recognition, 33*(3), 515–519.
Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics, 9*(1), 62–66.
Paikin, G., & Tal, A. (2015). Solving multiple square jigsaw puzzles with missing pieces. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4832–4839).
Pomeranz, D., Shemesh, M., & Ben-Shahar, O. (2011, June). A fully automated greedy square jigsaw puzzle solver. In *Computer vision and pattern recognition (CVPR), 2011 IEEE conference on* (pp. 9–16). IEEE.
Schauer, C., & Prandtstetter, M. (2010). A memetic algorithm for reconstructing cross-cut shredded text documents. In *International conference on hybrid metaheuristics: 6373* (pp. 103–117). Springer-Verlag.
Sleit, A., Massad, Y., & Musaddaq, M. (2013). An alternative clustering approach for reconstructing cross cut shredded text documents. *Telecommunication Systems, 52*(3), 1491–1501.
Richter, F., Ries, C. X., Cebron, N., & Lienhart, R. (2013). Learning to reassemble shredded documents. *IEEE Transactions on Multimedia, 15*(3), 582–593.
Townsend, J. T. (1971). Erratum to: Theoretical analysis of an alphabetic confusion matrix. *Attention, Perception, & Psychophysics, 10*(4) 256-256.
Turney, P. D. (2002). Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics, July* (pp. 417–424). Association for Computational Linguistics.
Ukovich, A., Ramponi, G., Doulaverakis, H., Kompatsiaris, Y., & Strintzis, M. G. (2004). Shredded document reconstruction using MPEG-7 standard descriptors. In *Signal Processing and Information Technology, 2004. Proceedings of the Fourth IEEE International Symposium on, December* (pp. 334–337). IEEE.
Ukovich, A., & Ramponi, G. (2008). Feature extraction and clustering for the computer-aided reconstruction of strip-cut shredded documents. *Journal of Electronic Imaging, 17*(1), 013008.
Wagstaff, K., Cardie, C., Rogers, S., & Schrödl, S. (2001). Constrained k-means clustering with background knowledge. In *ICML, June: 1* (pp. 577–584).
Xing, N., & Zhang, J. (2017). Graphical-character-based shredded Chinese document reconstruction. *Multimedia Tools and Applications, 76*(10), 12871–12891.
Xu, H., Zheng, J., Zhuang, Z., & Fan, S. (2014). A solution to reconstruct cross-cut shredded text documents based on character recognition and genetic algorithm. *Abstract and applied analysis*: 2014. Hindawi.
Zhou, Z. H. (2016). *Machine learning* Beijing: Tsinghua University (pp. 197–220). press. Chinese.