SEMI-AUTOMATIC ASSEMBLY OF REAL CROSS-CUT SHREDDED DOCUMENTS

Aaron Deever and Andrew Gallagher

Eastman Kodak

ABSTRACT

This paper introduces a semi-automatic approach for cross-cut shredded document reassembly. Automatic algorithms are proposed for segmenting and orienting individual shreds from a scanned shred image, as well as for computing features and ranking potential matches for each shred. Additionally, a human-computer interface is designed to allow semi-automatic assembly of the shreds using the computed feature and match information. Our document de-shredding system was tested on puzzles from the DARPA Shredder Challenge, allowing successful reconstruction of multiple shredded documents and demonstrating the effectiveness of the automatic algorithms.

Index Terms— document assembly, cross-cut, shreds

1. INTRODUCTION

Documents are shredded for a variety of reasons, but the foundational reason is to destroy the information on the document. Naturally, this leads one to ask, "How secure is the information on a shredded document?"

There are a variety of shredders and shredding methods. In general, a shredder that produces more (i.e., smaller) pieces, or shreds, from the page is more secure. For example, a high level of security is provided by cross-cut shredding where the resulting shreds are 0.8 mm $\times$ 4 mm [1]. Least secure are strip shredding (cutting the document into strips that span the length of the document) or hand-shredding into large pieces. A cross-cut shredder employs two cutter drums rotating in opposite directions. A document is forced through the drums to produce the shreds. This paper describes reconstructing a document by assembling the shreds from a cross-cut shredder in a semi-automatic fashion where the human and the computer collaborate.

This work was motivated in part by the DARPA Shredder Challenge [2]. In order to evaluate the possibility of reconstructing shredded documents, a test set of five puzzles of increasing difficulty was created and posed as a challenge to the public from October 27 to December 3, 2011. Each puzzle had one or more associated questions that could be answered based on information contained in the shredded document(s). Complete reconstruction of documents was not strictly necessary. A puzzle was considered solved once it was reconstructed sufficiently to extract the information necessary to answer the associated questions. Our team (EK555) produced a semi-automatic solution that allowed us to completely solve two puzzles, and partially solve another two puzzles, resulting in 11 points and 17$^{\text{th}}$ place out of over 9000 registered teams, while only 13 teams produced more points.

(a) Puzzle 1

Fig. 1: Puzzle 1 of the DARPA Shredder Challenge.

2. RELATED WORK

In the general sense, document de-shredding is a type of puzzle. The computational assembly of puzzles has been explored in the literature, beginning with Freeman et al. [3]. Some success has recently been shown for assembling square-piece jigsaw puzzles arranged on a grid [4, 5]. These methods assume that the pieces are perfect squares, and that the assembled puzzle forms a grid-graph of pieces. However, neither assumption is true for shredded documents, and the results do not necessarily translate to the problem of assembling the shreds of a document.

The automatic assembly of shredded documents is a particularly difficult puzzle for a number of reasons. First, the number of pieces (shreds) can be large (thousands of pieces per page), and the complexity of assembly is exponential in the number of pieces. A few authors specifically address the challenges of shredded document recovery. In [6, 7], reassembly of hand-torn images is proposed based on shape and color. The number of pieces is relatively low (30 pieces in the largest example), and the piece shapes are relatively distinct. In [8, 9], simulated strip-shredded documents (in strip-shredded documents, the strip runs the length of the docu-

ment) are reconstructed using color cues.

Our work has the following contributions: First, we propose a semi-automatic interface for reconstructing real (instead of simulated) cross-cut shredded documents with hundreds of pieces. Second, we propose shred orientation features. Third, we propose a fast-matching procedure for determining potential matches for a given shred. Finally, we propose a set of quantitative measures of document reassembly performance.

3. APPROACH

Our system was developed to solve the DARPA Shredder Challenge puzzles. Each puzzle contained images of one or more pages of shreds that had been manually placed face-up on a pink background and scanned at 400 dpi. Several different shredders were used to produce the Challenge materials. Fig. 1 shows the image from Puzzle 1. Subsequent puzzles all have more than one page of scanned shreds, up to a total of 20 pages for Puzzle 5.

Our system first performs preprocessing steps to extract and orient each of the puzzle pieces from the scanned shred image. Next, we perform feature extraction and matching to characterize the appearance of each shred and to determine likely matches for each shred. These initial two stages are automatic. Finally, the human user enters the loop through semi-automatic assembly. In the following subsections, we will describe each of these stages in more detail.

3.1. Preprocessing: Parsing the Shreds

We use pixel color to segment the shreds of the document. Pink pixels are considered background, and the remaining pixels are considered foreground (shred) pixels. A connected components algorithm is used to extract connected groups of shred pixels. Each connected group is considered a shred piece, and each has a corresponding mask. Example shreds from Puzzle 1 are shown in Fig. 2.

Following segmentation, each shred is oriented with the following two-stage process. First, Principle Component Analysis determines the dominant axis of the shred, and a rotation is applied to vertically align this axis. Next, the up-down orientation of a shred is determined based on the observation that shreds contain two distinctly different profiles at each end: one an *arrow*, and the other a *tail*. The shreds from a cross-cut shredder typically all have the same orientation; either the arrow or the tail is consistently towards the top of the document. To distinguish the arrow from the tail, features are computed from the shred mask, considering the top and bottom N (we use $N = 20$) rows of the shred, and a linear classifier is applied. The features are the following:

notch: a notched row is a row of the mask that contains one or more background pixels between the left and right mask border of the shred on the n^{th} row. The **notch** feature is the total number of background pixels for each of N rows. This is typically high for the tail, and low (or zero) for the arrow.

slope: the number of consecutive (non-notched) rows that have more foreground pixels than the row before. Usually, the arrow end of the shred has a larger value than the tail end.

point: the distance from the vertical axis of the shred to the centroid of shred pixels in the first row. Usually, the arrow end has a smaller value because the point is near the vertical axis of the shred.

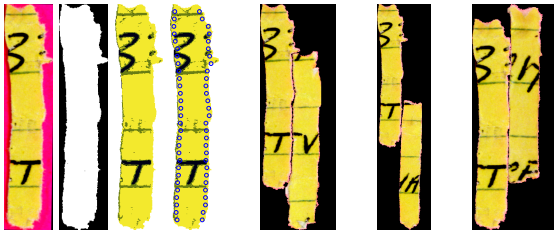


Fig. 2: Using color, the pixels from a shred in (a) and mask in (b) are classified as writing, ruled lines, or background, and the result is shown in (c) (color-coded). The left and right edges of the classification result are sampled (d). Then, matching is performed to find the most compatible matches (on the right side, in this case). The top three matches are shown in (e)-(g). Note that the offsets are automatically found. In this case, the top match (e) is correct.

The classifier output is used to orient the shreds so that the arrow is down (as shown, for example, for the Fig. 2 shreds). We have found the auto-orientation performs reliably. For Puzzle 1, 111 of 119 (93%) of the important shreds (containing handwriting) were correctly oriented with the classifier. Errors were generally the result of paper tearing inside of the shredder, instead of being cleanly cut. For Puzzle 2, 91% of the important shreds (275 of 304) were oriented correctly. Even when the classifier is incorrect, it is not catastrophic. During, or even before, the human-computer assembly stage, the human has the opportunity to correct any errors.

3.2. Feature Extraction and Matching

For each shred, features are extracted from the left and the right edges of the piece. (For the present time, we neglect characterizing the image content at the narrow top or bottom edges of the shred.) In the DARPA challenge, a handwritten message is written on either lined (i.e., ruled) or unlined paper. In several puzzles, multiple colors of ink are used for writing. Our goal is to exploit all available information for computationally suggesting matches. For example, we know true matches will have ruled lines that extend across the border. Further, we expect that handwriting near the edge of one shred will often extend across the boundary and into the neighboring shred. For this reason, a valuable clue is the identity of the pen that made a particular marking.

The idea of the feature extraction stage is to capture the locations of markings and lines on the shreds. First, we classify each pixel of a shred to indicate whether it is writing ink, paper base color, or part of a ruled line. We support classifying each different color of ink when multiple colors of writing ink sometimes appear on a document. After pixel classification, we sample locations along the boundary of the shred and record the spatial positions of non-background features.

The first stage of feature extraction is to perform pixel classification. Here, the user indicates the ground truth labels for each of the ink types as well as the paper base color and ruled lines. In our examples, the user selects about 5-10 samples for each ink type, paper base, and ruled line type. Each pixel is classified with a nearest-neighbor classifier. Results of the classification are shown in Fig. 2.

In the next stage, ruled lines are found. Lines are identified on those rows where the portion of ruled line pixels is greater than T (we use $T = 0.25$). Next, the pixel-classified shred is sampled along both the left and the right edges to determine the locations of markings. For example, samples on the left edge of the shred are taken on each row of the shred by sampling at position $(i_o, j_o + B)$ where (i_o, j_o) is the left-most foreground pixel in the mask on row i_o, and B is a small offset (we use $B = 5$ pixels) for avoiding pixels with colors that may be mixed with the pink background. Fig. 2 shows representative samplings positions (for clarity, the sampling positions for only every 16^{th} row is shown). The row index i of non-background pixels is stored in a feature structure.

We then search for good matches on both the left and the right sides of a shred. We define a matching cost $C(p_i, p_j, o_{ij})$ to indicate the compatibility of matching shred p_j to the right of shred p_i with an offset of o_{ij} in pixels. The offset o_{ij} indicates, in pixels, the row offset of shred p_j relative to shred p_i and can be negative (when p_j is above p_i) or positive (in the opposite case). For puzzles with a large number of shreds, it is intractable to find $C(p_i, p_j, o_{ij})$ for every possible offset. Instead, we consider only offsets o_{ij} in a pruned set $\mathbf{O_{ij}}$ of offsets. When ruled lines are detected, the offset set $\mathbf{O_{ij}}$ contains all offsets that align a ruled line in p_i with one in p_j. When ruled lines are not present, the offset set $\mathbf{O_{ij}}$ is the set of all offsets that align a non-paper base pixel sampled from the left edge of p_i with the same type of non-paper base pixel sampled from p_j. The cost $C(p_i, p_j, o_{ij})$ itself is the sum of local rewards (negative values) and costs (positive values), and more compatible matches have more negative costs. A local reward $-K$ is scored whenever pixels across the boundary have the same label (for non-background classes), and a penalty $+K$ is scored when the class labels disagree.

3.3. Human-Computer Assembly

Once all the shreds have been analyzed to extract features and rank potential matches, a user interface allows human confirmation of suggested matches. In cases where the correct match is near the top of the ranked list, a human can quickly scroll through the suggested matches and locate the correct shred. Fig. 3(a) shows the interface used to allow assessment and confirmation of potential matches. This interface focuses on one reference piece, matching either the left or right side. Potential matching pieces are displayed on the appropriate side, offset by a number of rows as determined during the feature matching process. The row offset can be adjusted as necessary. Potential matches can be evaluated in ranked order, beginning with the piece suggested as the best match for the reference piece.

(a) Shred Matching (b) Component Viewer

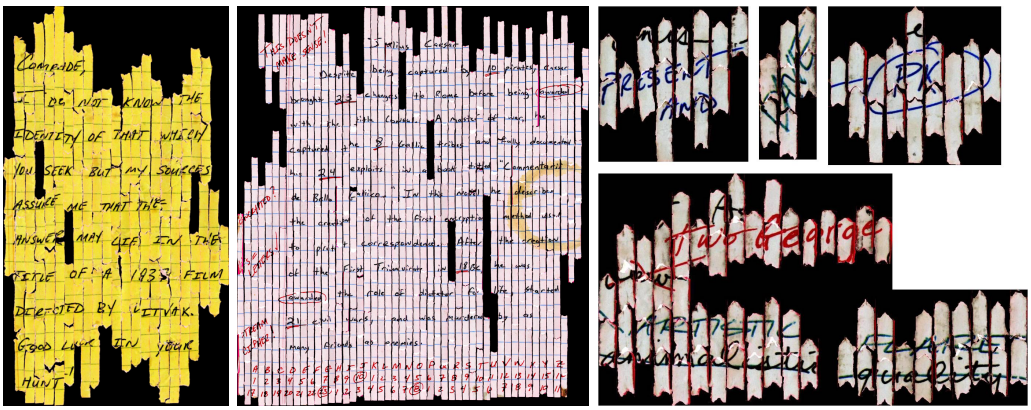
Fig. 3: (a) User interface to evaluate potential matches. The displayed shreds are from Puzzle 1. (b) User interface to evaluate image components. The displayed shreds are from Puzzle 3.

At various stages in the puzzle-solving process, it is beneficial to look at the current state of progress. This allows for identification of matches that look good in isolation, but can be rejected as incorrect when viewed as part of a larger component. It also allows for broader evaluation of partially reconstructed regions and identification of specific characters to search for to match pieces based on what is required to complete a word or sentence or figure. Fig. 3(b) shows a second interface that allows the human to view the image components defined by the current set of matched shreds. It allows the user to zoom in and out, and mouse over regions of the image to identify the indexes of the underlying pieces.

4. RESULTS

We tested our semi-automatic document assembly system on the 5 DARPA Shredder Challenge puzzles. In the timeframe of the contest, we solved Puzzles 1 (206 shreds) and 2 (362 shreds) completely, and partially solved Puzzles 3 (1035 shreds) and 4 (2206 shreds). As such, the quantitative results presented here are based on Puzzles 1 and 2. Fig. 4 shows the reconstructed documents used to solve Puzzles 1 and 2, and our partial solution for Puzzle 4. As can be seen, the documents have not been completely reconstructed. Rather, nearly all of the important pieces of the documents, defined as those containing handwriting, have been reassembled, while the remaining pieces have been largely ignored.

The speed at which matches can be confirmed during the human-computer assembly step is dependent on the accuracy of the automatic ranking of potential matches for each piece



(a) Puzzle 1 (b) Puzzle 2 (c) Puzzle 4

Fig. 4: Reconstructions of Puzzles 1, 2, and 4 (partial) of the DARPA Shredder Challenge.

Rank	Percent of Correct Matches	
	Puzzle 1	Puzzle 2
1	5%	1%
top 5%	27%	15%
top 10%	41%	25%
top 15%	54%	35%

Table 1: Distribution of Ranks of Correct Matches for Puzzles 1 and 2

Accuracy	Percent of Accurate Offsets	
	Puzzle 1	Puzzle 2
Exact	34%	26%
Within 10 pixels	44%	34%

Table 2: Accuracy of Offsets for Correct Matches for Puzzles 1 and 2

based on the computed features. Table 1 shows the distribution of ranks among correct matches from Puzzles 1 and 2. Since a reference piece may have multiple matches on each side, one of which may only marginally overlap, only matches for which the overlap was at least 35% of the size of the smaller of the two pieces were considered. For the Puzzle 1 reconstruction shown in Fig. 4, there were 252 such matches. For the Puzzle 2 reconstruction, there were 512 such matches. As seen in Table 1, 5% of the matches were the top ranked (out of 239 total pieces) for Puzzle 1. Furthermore for Puzzle 1, more than $\frac{1}{4}$ of the correct matches were ranked in the top 5% of all matches, and more than $\frac{1}{2}$ of the correct matches were ranked in the top 15% of all matches. This allowed for rapid identification and confirmation of correct matches. For Puzzle 2, the matching was more difficult, but the automatic ranking still provided significant efficiencies for locating correct matches. We will share our pieces and matching data so other researchers can compare results.

The efficiency of the identification and confirmation of matches is further enhanced when the offset proposed in the automatic evaluation is accurate. When the correct match is proposed with the correct offset, visual inspection can very quickly confirm the accuracy of the match. Table 2 shows the accuracy of the automatic offsets determined in Puzzles 1 and 2, given two matching pieces. Again, restricting consideration to only those matches for which the overlap was at least 35% of the size of the smaller of the two pieces, it can be seen that for Puzzle 1, over 40% of the offsets were within 10 pixels of the correct value, allowing easy visual verification. For Puzzle 2, over $\frac{1}{3}$ of the offsets were close enough to allow quick visual confirmation.

5. CONCLUSION

In this paper, we introduce a new framework for semi-automatic assembly of shredded documents. Our method uses computer vision techniques for suggesting matches that are then verified by the human to build up completed document components. To the best of our knowledge, this paper is the first detailed description of a semi-automatic approach for puzzle assembly in the literature. However, a great many challenges in this area remain to best exploit the strengths of the computer and the human.

6. REFERENCES

[1] "Office machines - destruction of information carriers - part 1," Tech. Rep. DIN 32757-1, German Institute for Standardization, 1995.

[2] "DARPA Shredder Challenge," http://www.shredderchallenge.com/.

[3] H. Freeman and L. Garder, "Apictorial jigsaw puzzles: The computer solution of a problem in pattern recognition," *IEEE. Trans. on Electronic Computers*, 1964.

[4] T. S. Cho, S. Avidan, and W. T. Freeman, "A probabilistic image jigsaw puzzle solver," in *Proc. CVPR*, 2010.

[5] D. Pomeranz, M. Shemesh, and O. Ben-Shahar, "A fully automated greedy square jigsaw puzzle solver," in *Proc. CVPR*, 2011.

[6] S. Cao, H. Liu, and S. Yan, "Automated assembly of shredded pieces from multiple photos," in *Proc. ICME*, 2010.

[7] Edson Justino, Luiz Oliveria, and Cinthia Freitas, "Reconstructing shredded documents through feature matching," *Forensic Science International*, 2006.

[8] M. Marques and C. Freitas, "Reconstructing strip-shredded documents using color as feature matching," *Proc. ACM Symposium on Applied Computing*, 2009.

[9] W. Morandell, "Evaluation and reconstruction of strip-shredded text documents," M.S. thesis, Vienna University of Technology, 2008.