Document Analysis Applied to Fragments: Feature Set for the Reconstruction of Torn Documents

Markus Diem*
Institute of Computer Aided Automation
Favoritenstr. 9/1832
1040 Vienna
diem@prip.tuwien.ac.at

Florian Kleber*
Institute of Computer Aided Automation
Favoritenstr. 9/1832
1040 Vienna
kleber@prip.tuwien.ac.at

Robert Sablatnig
Institute of Computer Aided Automation
Favoritenstr. 9/1832
1040 Vienna
sab@prip.tuwien.ac.at

ABSTRACT

Document analysis is done to analyze entire forms (e.g. intelligent form analysis, table detection) or to describe the layout/structure of a document. In this paper document analysis is applied to snippets of torn documents to calculate features that can be used for reconstruction. The main intention is to handle snippets of varying size and different contents (e.g. handwritten or printed text). Documents can either be destroyed by the intention to make the printed content unavailable (e.g. business crime) or due to time induced degeneration of ancient documents (e.g. bad storage conditions). Current reconstruction methods for manually torn documents deal with the shape, or e.g. inpainting and texture synthesis techniques. In this paper the potential of document analysis techniques of snippets to support a reconstruction algorithm by considering additional features is shown. This implies a rotational analysis, a color analysis, a line detection, a paper type analysis (checked, lined, blank) and a classification of the text (printed or hand written). Preliminary results show that these features can be determined reliably on a real dataset consisting of 690 snippets.

Categories and Subject Descriptors

I.7 [**Computing Methodologies**]: Document and Text Processing; I.7.5 [**Document and Text Processing**]: Document Capture—*Document Analysis*; I.4 [**Computing Methodologies**]: Image Processing and Computer Vision

Keywords

Document reconstruction, skew, layout analysis

1. INTRODUCTION

To make information (writings, drawings) inscribed on writing materials (paper, parchment, papyrus) unreadable one possibility is to fragment the writing material. Although parts of the information on single fragments still exist, the entire text and therefore the context of the document is destroyed. Reasons for an intended tearing of writing materials are either criminal intentions (business crime, tax fraud investigation, secret service documents [6]) or e.g. the protection of sensitive data/personal information (bank details, credit card numbers). Unintended fragmenting of documents concern either ancient manuscripts that are fragmented due to environmental effects (influence of mold, water) or due to catastrophes like the collapse of the historical archive of the City of Cologne (a total of 18 shelve kilometers of books has been destroyed)[11]. A reconstruction of fragmented writing materials allow to retrieve and to analyze the lost content. This is done on objects of cultural and historic value, or e.g as already mentioned for crime investigation.

In this paper only fragments of "manually" teared paper containing German or English text are considered. This means that snippets have an irregular shape and overlapping or missing parts are possible. Contrary mechanically document shredders that produce either stripes or parallelograms (cross-cut-shredder) are not treated. The reconstruction of shredded paper is discussed in e.g. Ukovich et al. and De Smet et al. [44, 39]. The Fraunhofer Institute for Production Systems and Design Technology Berlin has also developed a system for the reconstruction of shredded paper which has already been used by the German police and tax fraud investigation [36]. It is also assumed that all snippets are scanned with the same resolution with a defined background (allows to apply a global threshold to get a mask image).

Reassembling algorithms use either the shape of the fragments[31, 12], the content of the fragments (e.g Nielsen et al. [30]) or a combination of shape and content as a feature (e.g Yao et al. [46]). As content feature either color analysis [10] or e.g. inpainting and texture synthesis techniques [34] are done for each piece. By taking only the border regions into account the main information printed on the snippet (which can be used for reconstruction) is lost. Problems according manually torn paper are overlapping fragments (if shearing effects appear on the disrupted border), and that gaps can occur if pieces of borders are broken or lost.

In this paper document analysis techniques are applied to calculate following features: the rotation of each snippet

*Corresponding Authors

DAS '10, June 9-11, 2010, Boston, MA, USA

according the inscribed text, the color of the text and the paper, a line segmentation for e.g. form analysis, a classification of the paper type (lined, checked, blank) and a classification of the text (printed vs. handwritten). Additional characteristics that will be considered in the future but are not described in this paper is the document layout (e.g. line spacing, font). As a result all fragments can be clustered according their inscribed content. This means that e.g. only the shape of snippets of paper type *checked* have to be matched against each other. Due to the complexity and the computational effort of the puzzle problem [14] a clustering as described is necessary (see e.g. Nickolay and Schneider [29, 37]). The evaluation of the algorithm has been done on a test set of 690 snippets of the Stasi-files[29]. The distribution of the snippets' sizes is shown in Figure 1. The content ranges from well preserved to faded out handwritten documents, typewriter printed documents, forms, drawings and carbon copies. Due to the confidentiality it is not possible to publish the dataset. The main benefit of the presented algorithm is that it is only limited to a scan resolution of 300 dpi, German or English text (see Section 3.1.3) and at least a line of text printed on the snippet (e.g. no constraint on the detectable angle range for the skew estimation). The Fraunhofer Institute for Production Systems and Design Technology (IPK) is investigating methods for the reconstruction of torn Stasi-files [36, 37]. The presented algorithms are used by Fraunhofer IPK within this project for the clustering of snippets.

This paper is organized as follows: Section 2 reviews the state of the art of 2D reconstruction, skew estimation methods and line segmentation algorithm. In Section 3 the features calculated for each snippet are described in detail, while Section 4 presents the results of the algorithm developed. Finally a conclusion is given in Section 5.

2. RELATED WORK

There are several studies for the automated assembly of shredded or torn paper. For text documents the reassembly of strip-shredded [44] or cross cut shredded [33] is the main search area in the forensic domain. An approach that is dealing with the reconstruction of torn paper using the shape is described in Berger [5] and de Smet [38]. The complexity of algorithm for solving jigsaw puzzles, edge-matching and polyomino packing puzzles are shown in Demaine [14]. Methods proposing algorithm for skew estimation include techniques based on projection profiles [2, 40], the Hough transformation [1] and methods based on properties of the Fourier transform [32]. A summary and a classification of skew estimation algorithm is shown in Lins et al.[26] and Hull[21]. The main drawback of proposed skew estimation algorithm is a limitation to specific applications.

Since a binary image is used for the calculation of the colors as well as for the skew estimation a segmentation of the snippets is necessary. A global binarization approach is published by Otsu. For degraded images adaptive binarization methods can be used (e.g. Sauvola et al. [35]). A different adaptive approach is described by Gatos [18]. This algorithm extracts an estimated background surface of the image, from which the original image is subtracted and afterwards thresholded. A comparison of different thresholding techniques is also done in Gatos[18]. An overview paper is

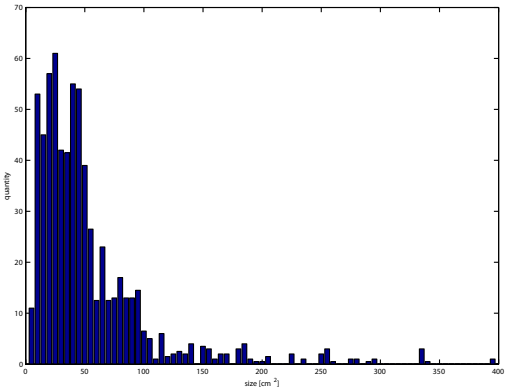

Figure 1: Snippet size distribution of the test set

presented in Leedham et al.[24].

To analyze forms or to remove underlines [3] or tables [17] from an image a line segmentation has to be performed. In Wu et al. [45] a morphological approach is done while methods based on analyzing pixel run lengths are presented in Yu et al. [47] and Zheng et al. [48]. Run lengths represent line segments that are merged by described constraints.

3. METHODOLOGY

In this section the selected fragment features are presented. For each snippet the orientation, the color of the inks/paper, a line detection in the segmented image, the paper type and the type of the text is calculated. The orientation assignment and the classification of the writing type (print or handwritten text) is based on the gradient orientation of each pixel. To determine the color a foreground/background segmentation is performed. In addition a line detection/removal in the binary image based on runlengths is done. The determination of the paper type is done using FFT.

3.1 Skew Estimation

In Kavallieratou et al. [22] problems according skew estimation are the dependence of text type, the computational effort and a limitation to specific applications. Furthermore the requirement of large text areas and a certain resolution, a constraint on the detectable angle range and the page layout is listed [22].Figure 1 shows the distribution of the snippet sizes' of the used test set (stamp size up to Din A4). In addition there is no restriction to the detectable angle range or to the writing type (printed or handwritten). Assumptions are that all snippets are scanned at the same resolution (at least 300 dpi) and the existence of text. Since the up/down orientation is based on statistical analysis of ascenders and descenders [7] the algorithm depends on the script as well as the language. The alignment of a snippet is calculated in three steps called *Global Orientation*, *Quadrant Estimation* and *Up/Down Orientation*. The global orientation is based on the gradient orientation (see Sun et al. [41]) of each pixel, which allows to determine the alignment up to 90°. This is based on the fact that gradients of text are aligned either in writing direction or perpendicular to the writing direction. Characters and their corresponding pixel gradients are visualized in Figure 2. To decide after the

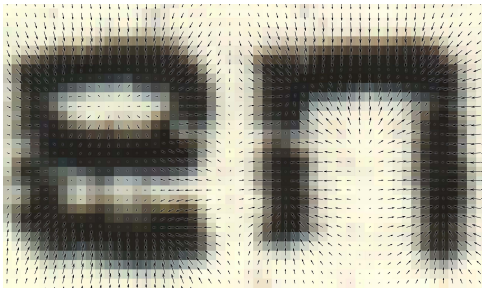

Figure 2: Gradient vectors of a manuscript image

global orientation step if the snippet is aligned either horizontal (independent if the text is flipped upside/down) or vertical the orientation of single words is taken into account. After the quadrant estimation the snippet is aligned either correct or flipped upside/down. As a last step the up/down orientation is calculated by analyzing ascenders and descenders. Figure 3 shows the possible orientations after each step. In the following subsections the steps for the skew

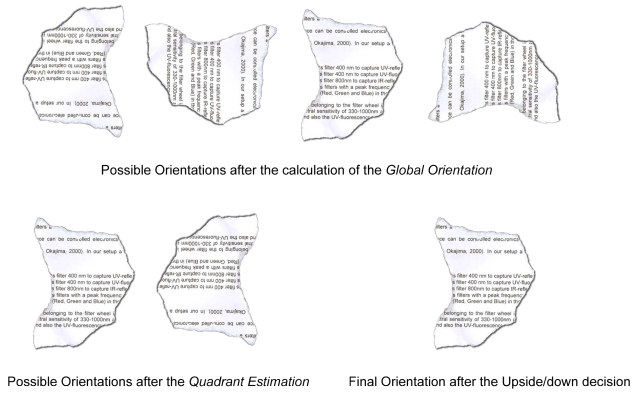

Figure 3: Orientations of a fragment after the main processing steps

estimation are described in detail.

3.1.1 Global Orientation

The rotation estimation of a given snippet is inspired by Lowe's SIFT [27] and Sun et al. [41]. Thus, the gradient magnitude $m(x, y)$ and the gradient orientation $\theta(x, y)$ of a snippet are computed. The gradient orientation of each pixel is accumulated into the bin corresponding to its orientation and weighted by the gradient magnitude $m(x, y)$. Peaks in the resulting histogram indicate the main orientation of a given snippet (see Figure 4). Since reflected gradient vectors indicate whether a border is black-white or white-black but do not make a statement about the exact local orientation, $\theta(x, y)$ is computed on the interval $[-\pi/2 \;\; \pi/2]$. To determine the global orientation the highest histogram bin is taken into account. To solve problems like straight snippet borders (which will produce the highest peak) and italic text (a second peak is produced) local statistics are applied. For a detailed description see Kleber et al. [23].

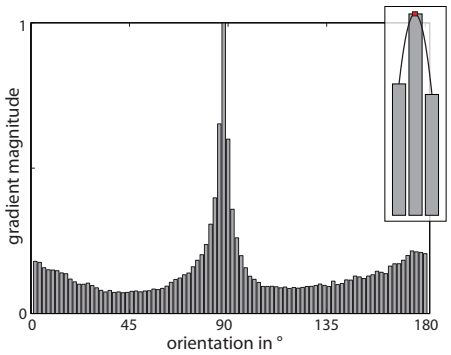

Figure 4: Orientation histogram with spline interpolation

3.1.2 Quadrant Estimation

The orientation histogram determines the main orientation within a quadrant $[0\ \pi/2]$. In order to determine the quadrant, the content of a snippet needs to be considered. The snippet is binarized (see Section 3.2) and a local projection profile is calculated. For each blob the minimum area rectangle is the basis for the quadrant estimation (see Figure 5). A blob generally represents a word which is guaranteed by a previous smoothing of the image. In order to determine the quadrant, the minimum area rectangles are first rotated relative to the main orientation. Then they are accumulated into an x and y bin depending on their angle. A weight based on the angle, size and aspect ratio is assigned to each rectangle. Thus, rectangles having a relative orientation of 45° have a lower weight than those with 1°. If the resulting y bin is higher than the x bin, the snippet needs additionally to be rotated by 90°. Since the entropy of snippets can be

Figure 5: Minimum area rectangles

low (only a few words written), it is necessary to establish a confidence measure which allows for soft clustering decision in the final matching algorithm. For a detailed description see Diem et al. [15].

3.1.3 Up/Down Orientation

The page up/down orientation determination is based on the work of Caprari [7]. Therefore the decision is based on the frequency of ascenders and descenders of roman letters and arabic numerals. Statistics of German and English text show that the occurrence of ascenders is dominating [25].Caprari analyzes the asymmetry of the line histogram based on the ascender and descender frequency. Since the algorithm is sensitive to the correct skew, the entire page is divided into

stripes. It turned out that the best results are gained when a snippet is divided into 6 stripes (see Diem et al. [15]).

3.2 Color

An additional feature that is used to cluster the given snippets is the color of the paper as well as the main color of the printed or handwritten text. It is obvious that different colors of inks/paper belong to different documents. Color segmentation for text extraction is a common field in document analysis (see Mancas-Thillou and Gosselin [28], Hase et al. [19]). To segment the snippets into background and

Figure 6: Degraded characters and their mean color value (without weighting with the gradient magnitude)

printed information, color spaces (RGB, CIE $L^*a^*b^*$, HSV, XYZ) [42] have been tested. It turned out, that for the segmentation of the foreground the value channel of the HSV color space is the best choice (see Diem et al. [15]). The threshold for the colored text is determined in the Saturation channel of the HSV color space and black/gray colors are determined in the V channel. To determine a threshold the 1-D histogram of the luminance channel is calculated. It is assumed that the background (paper, parchment,...) has a higher V value than the foreground (writings, images). The threshold is set by Otsu's method.

In addition to segment colors a Gaussian Mixture Model (GMM) [8] is applied to the S channel of the HSV color space. As a result the two gaussian distributions representing the background p_b and the foreground colors p_f are fitted into the histogram using Expectation Maximization (EM)[9, 20]. After the segmentation, the mean color for each blob is calculated. To avoid the influence of degraded characters (see Figure 6), the pixel values are weighted with $1-|m(x,y)|$, where m is the gradient magnitude of the pixel at the coordinates x,y. This reduces effects like the fading out of the ink at border pixels. The mean color for each blob is accumulated in a 3D RGB color histogram. The local maxima determine the existing colors of a snippet. This is done to reduce the amount of colors. For a detailed description see Diem et al. [15].

3.3 Line Detection

For form and text analysis all lines in a snippet have to be segmented. Also for the planned classification of the text (printed or handwritten) the line blobs have to be removed in advance, to avoid that lines are classified as text or connect different text lines. In this section the line segmentation method based on [48] and [47] is described.

The input image of the algorithm is a binary image, which is the result of the segmentation. On the binary image a vectorisation algorithm is performed that computes horizontal and vertical Directional Single Connected Chains (DSCC) according to [48]. After the horizontal and vertical DSCC analysis blobs with multi-connected DSCCs are fragmented.

It is assumed that lines or parts of lines have a length that is greater than the width of a character, the aspect ratio is less than 0.5 (lines have elongate shape), and the size (pixel area) is greater than the size of a character. To determine the aspect ratio the minimum area rectangle of a DSCC is calculated (see Figure 7 (b)). In addition the ripple-rating of a DSCC is calculated. The ripple rating is defined as the area of the DSCC in proportion to the area of the minimum area rectangle. After removing all DSCCs that do not

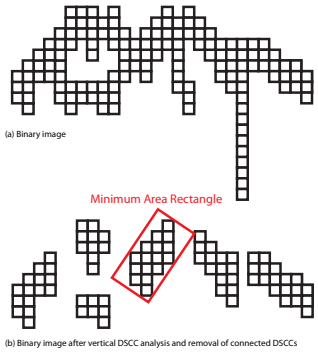

Figure 7: Example for a vertical DSCC analysis of an binary image

fullfill the described characteristics the remaining parts are merged. This is done by analyzing the orientation and the gap between two DSCCs. The orientation is defined as the angle between the axis of abscissae and the major axis of the DSCC. The gap between two line candidates is defined dynamically as $1.5 \times linelen$, where $linelen$ is the length of the longer line candidate. The deviation of the orientation of two candidates has to be less than 4°. The thresholds $(4°, 1.5 \times linelength)$ have been evaluated empirically.

Figure 8 shows an example of a snippet with printed text (parts of the text are underlined) and the steps of the line segmentation algorithm. The result after the horizontal and vertical calculation of DSCCs and the background-marking of DSCCs regions which have neighbouring more than one pixel runlength is shown in Figure 8 (d) and (e). After the filtering process the lines are shown in Figure 8 (f), and the final result is shown in Figure 8 (c) (gaps are merged).

3.4 Paper Type Classification

The Fourier transform is exploited for the analysis of a snippets' lining. Therefore, the gradient image without the snippets' border is computed and transformed using a FFT. If a snippet is lined, repeated peaks occur along the main direction of the lining. These peaks are present in two orthogonal directions if checked snippets are regarded (see Figure 9). In order to extract the peaks invariant to rotation, a polar transformation is applied on the FFT magnitude image. Obviously, the repeating peaks are now parallel to each other which allows for a simple detection based on accumulation. Thus, statistical features of the two orthogonal main directions are extracted. Finally, a k-NN is used to classify each

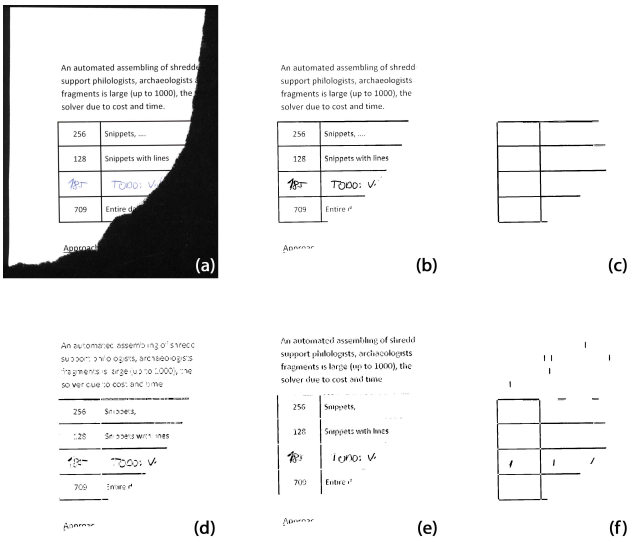

Figure 8: (a) Snippet (b) segmented image (c) merged line image (d) horizontal lines (e) vertical lines (f) filtered line image

snippet into void, lined and checked.

Beside using the FFT for the texture analysis, the horizontal and vertical line frequencies can be extracted from the spectral image. Moreover, the relative rotations' result are further verified by simply comparing the global maximum of the (accumulated) polar image with the current relative angle. To improve the results of the texture classification,

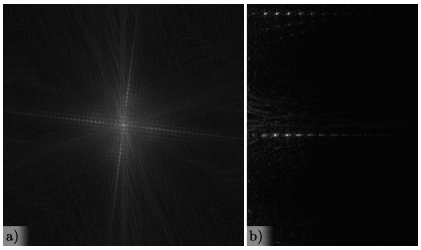

Figure 9: Fourier spectrum of a checked snippet (a). Note the repeating peaks along the main orientation axis. Polar transformed Fourier image (b)

the FFT is calculated only on an image patch having a size of 512×512 px instead of the entire image. The position of the patch is calculated to exclude text regions (if possible). Therefore an integral image of the snippet is calculated to apply an averaging filter. It is estimated that background regions (e.g. white) have higher values than regions that contain text (dark ink). In addition an Otsu Threshold is applied to the image patch to filter out text.

3.5 Writing Type Classification

An additional feature calculated for every snippet is the type of the writing to distinguish handwritten text from printed text. This supports the matching algorithm since the probability that two snippets of the same type match is higher. The feature to distinguish handwritten from printed text is the main orientation of static stroke components from single characters. The classification is done for every text

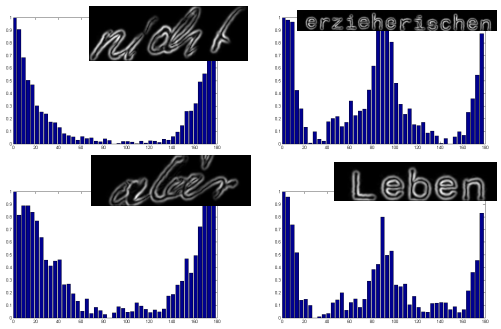

Figure 10: Text classification of word blobs

word. For the word extraction the binary image calculated in Section 3.2 is used. The line image determined in Section 3.3 is excluded from the binary image to avoid lines being classified as text. To get words from the binary image, the local projection profile is calculated (see [4]), which estimates also the line spacing and is used to cut connected lines (through ascenders and descenders). The minimum area rectangle of every word determines the regions that are classified in the input image. Figure 10 shows a handwritten text and printed text words and their corresponding feature histogram. It can be seen that the main orientation of printed text is in writing direction as well as perpendicular to the writing direction, which can be clearly distinguished from the main orientation of handwritten text. A k-NN is used to classify each text blob into printed or handwritten text.

4. RESULTS

In order to generate the groundtruth, a tool was developed which allows for a manual assignment of the text (print/ manuscript), the paper (void/lined/checked) and the rotation angle. In addition the lines have been assigned manually in the binary image. For the evaluation of the writing type classification entire pages with either printed or handwritten text have been analyzed. The annotated test set consists of 690 images containing torn documents of all classes.

4.1 Skew Estimation Results

Based on this annotated test set, the rotational analysis described in Section 3.1 is evaluated. Additionally, parameters needed are determined so that a good generalization performance of the methods is given. For a detailed description of the parameters and the evaluation of the parameters see Diem et al. [15].

The statistical moments of the relative angle error are given in Table 1. In this evaluation 678 images are considered even though the test set consists of 690 images as mentioned before. This arises from the fact that some snippets exist (e.g. no content, no straight border) where the main orientation cannot be assigned to. The difference of 1.58° between the mean error and the median error can be traced back to the

Number of images:	678	
Mean error:	1.95ř	$\sigma \pm 6.13^\circ$
Median error:	0.37ř	$q_{0.25}$ 0.16° – $q_{0.75}$ 0.82°

Table 1: Results of the skew estimation

fact that 32 outliers exist where the rotational analysis completely fails. These outliers push the mean error.

In addition to the tests made concerning the relative orientation, a test was performed on the quadrant estimation. There, 47 (7.63%) images could not be rotated correctly as a consequence of false binarization and errors of the relative orientation estimation. In addition to the full testset, a set consisting of 164 snippet images was created which contain at least 5 partially visible text lines. On this test set 5 (3.05%) snippets were not rotated correctly. If 5 images having a relative error above 5° are not regarded, the quadrant estimation fails in 3 cases (1.89%).

The results gained with the method for determining the up/down orientation are given in Table 2 when varying the number of stripes. On the whole testset 9.04% were not correctly rotated even though images are present which contain solely capital letters or less than a half textline.

# Stripes	Total	Relative Error $< 5^\circ$
1	10.98% (18/164)	5.37% (8/149)
2	8.54% (14/164)	3.36% (5/149)
4	7.32% (12/164)	1.34% (2/149)
6	6.71% (11/164)	1.34% (2/149)
8	6.71% (11/164)	2.01% (3/149)

Table 2: Results of the up/down orientation determination if the number of stripes is changed.

4.2 Line Detection Results

The evaluation of the line detection was done on a test set of 27 images. The test set contains images with underlined text, snippets without lines and images with tables have been choosen. In the binary image the groundthruth was manually tagged. Table 3 shows the correct classification rate (true positive) as well as the amount of falsely as line classified pixel (false positiv). This error results from the merging process in the alghorithm. The high value of false

Number of images:	27
Correct as line classified pixel:	91.58%
Falsely as line classified pixel:	15.05%

Table 3: Line classification results

positives arise from the calculated line width which is used to fill gaps. The line width is estimated as the height of the minimum area rectangle of the two connecting blobs, which is higher than the correct line width. In addition varying line widths are not treated. Also if gaps are filled correctly that are not in the segmented image, lead to a higher percentage of falsely classified line pixel.

4.3 Paper Type Results

The evaluation of the texture analysis was carried out on a general dataset (SetA) consisting of the first 235 images and on a cleaned dataset (SetB) where forms and tables where rejected since their groundtruth is ambiguous. The overall classification rate on SetA is 86.50% (205/237) which is certainly increased in SetB having a performance of 89.81% (194/216). The confusion matrix of both datasets is given in Table(4, 5).

computed/GT	void	lined	checked
void	**144/159**	15/159	0/159
lined	14/58	**44/58**	0/58
checked	2/20	1/20	**17/20**
computed/GT	void	lined	checked
void	**90.57%**	9.43%	0.00%
lined	24.14%	**75.86%**	0.00%
checked	10.00%	5.00%	**85.00%**

Table 4: Texture analysis confusion matrix of SetA

computed/GT	void	lined	checked
void	**138/151**	13/151	0/151
lined	6/45	**39/45**	0/45
checked	2/20	1/20	**17/20**
computed/GT	void	lined	checked
void	**91.39%**	8.61%	0.00%
lined	13.33%	**86.67%**	0.00%
checked	10.00%	5.00%	**85.00%**

Table 5: Texture analysis confusion matrix of SetB

4.4 Writing Type Results

To analyze the performance of the writing classification entire pages with printed and snippets with handwritten text (different hands) have been used for the evaluation. Text pages scanned at 300 dpi with the following serif and sans-serif fonts have been used: Times New Roman (serif, 12pt, 467 words), Courier New (serif, 12pt, 316 words), Arial (sans-serif, 12pt, 439 words), Myriad Pro (sans-serif, 12pt, 439 words), Blackadder ITC (look similiar to a handwritten style, 12pt, 467 words). The same text has been used for every page (if necessary the text has been shortened fo fit an entire page due to the different fonts). For the handwritten data 51 snippets with handwritten text (and different hands) have been analyzed. For the evaluation regions determined by minimum area rectangles of word blobs generated by the local projection profile (see Section 3.1.2) are used. Therefore it is possible that the number of word blobs in the evaluation image differs from the number of text words given for every font (e.g. two words have been connected by the local projection profile). The performance rate of the text type classification for each font is summarized in Table 6. The main error in printed text results from words with "rounded" characters like e.g. *do, good* which are classified as handwritten text. It can also be seen that handwritten like fonts, e.g. Blackadder ITC, is classified as handwritten text. Table 7 shows the precision of the text classification. Handwritten like fonts (Blackadder, see Figure 11) are not accounted in this statistic.

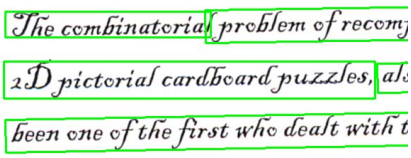

Figure 11: Classified text blobs of printed (handwritten like) text

font	classified as handwritten	classified as printed	number of words
Times	9	175	184
Courier	1	316	317
Arial	0	125	125
Myriad	0	88	88
Blackadder	88	6	94
handwritten	1070	311	1381

Table 6: Results of text classification for different fonts evaluated

font	precision	TP	total
printed	0.986	704	714
handwritten	0.775	1070	1381

Table 7: Results of text classification where TP abbreviates True Positives

5. CONCLUSIONS

In this paper a prerequisite, namely the calculation of characteristics of snippets, for a combined shape and pictorial approach that solves the tearing paper problem is presented. The methods presented namely skew and segmentation are needed for content analysis and the shape matching. On opposite the paper type, the writing type classification, color extraction and the line detection are used as features for the clustering of different snippet types. As future work additional methods to determine characteristics like the line spacing, the font type and the document layout will be analysed. In addition shape matching procedures will be evaluated and combined with the calculated snippet features.

6. ACKNOWLEDGMENTS

This work was supported by the Austrian Science Fund under grant P19608-G12 and by the Fraunhofer-Institute for Production Systems and Design Technology (IPK), Berlin.

7. REFERENCES

[1] A. Amin and S. Fischer. A document skew detection method using the hough transform. *Pattern Analysis and Applications*, 3(3 2000):243–253, 2000.

[2] A. D. Bagdanov and J. Kanai. Projection profile based skew estimation algorithm for jbig compressed images. In *ICDAR '97: Proceedings of the 4th International Conference on Document Analysis and Recognition*, pages 401–406, Washington, DC, USA, 1997. IEEE Computer Society.

[3] Z.-L. Bai and Q. Huo. Underline detection and removal in a document image using multiple strategies. *Pattern Recognition, International Conference on*, 2:578–581, 2004.

[4] I. Bar-Yosef, N. Hagbi, K. Kedem, and I. Dinstein. Line segmentation for degraded handwritten historical documents. *Document Analysis and Recognition, International Conference on*, 0:1161–1165, 2009.

[5] F. Berger. Ein hybrides Verfahren zur automatischen Rekonstruktion von handzerrissenen Dokumentenseiten mittels geometrischer Informationen. Master's thesis, Vienna University of Technology, Institute of Computer Graphics and Algorithms, Austria, 2008.

[6] BStU Berlin. Rekonstruktion von Unterlagen (german). accessed 11th december 2009. `http://www.bstu.bund.de/cln_012/nn_714874/DE/Archiv/Rekonstruktion/rekonstruktion__node.html__nnn=true`.

[7] R. S. Caprari. Algorithm for text page up/down orientation determination. *Pattern Recogn. Lett.*, 21(4):311–317, 2000.

[8] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: image segmentation using expectation-maximization and its application to image querying. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(8):1026–1038, Aug 2002.

[9] D.-C. Cheng, X. Jiang, and A. Schmidt-Trucksaess. Image segmentation using histogram fitting and spatial information. *Advances in Mass Data Analysis of Signals and Images in Medicine, Biotechnology and Chemistry,LNCS*, 4826:47–57, 2007.

[10] M. G. Chung, M. Fleck, and D. Forsyth. Jigsaw puzzle solver using shape and color. *Signal Processing Proceedings, 1998. ICSP '98. 1998 Fourth International Conference on*, 2:877–880, 1998.

[11] A. Curry. Archive collapse disaster for historians. *Spiegel online international*, accessed 04th march 2009. `http://www.spiegel.de/international/germany/0,1518,611311,00.html`.

[12] H. C. da Gama Leitão and J. Stolfi. A multiscale method for the reassembly of two-dimensional fragmented objects. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(9):1239–1251, 2002.

[13] H. C. da Gama Leitão and J. Stolfi. Measuring the information content of fracture lines. *Int. J. Comput. Vision*, 65(3):163–174, 2005.

[14] E. D. Demaine and M. L. Demaine. Jigsaw puzzles, edge matching, and polyomino packing: Connections and complexity. *Graphs and Combinatorics*, 23(1):195–208, 2007.

[15] M. Diem, F. Kleber, and R. Sablatnig. Analysis of document snippets as a basis for reconstruction. In *10th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST 2009)*, pages 101–108, St. Julians, Malta, 2009.

[16] H. Freeman and L. Garder. Apictorial jigsaw puzzles: The computer solution of a problem in pattern recognition. *Computers, IEEE Transactions on*, EC-13(2):118–127, April 1964.

[17] B. Gatos, D. Danatsas, I. Pratikakis, and S. J. Perantonis. Automatic table detection in document images. In *ICAPR (1)*, pages 609–618, 2005.

[18] B. Gatos, I. Pratikakis, and S. J. Perantonis. Adaptive degraded document image binarization. *Pattern Recogn.*, 39(3):317–327, 2006.

[19] H. Hase, M. Yoneda, S. Tokai, J. Kato, and Y. Suen. Color segmentation for text extraction. *Int. J. Doc. Anal. Recognit.*, 6(4):271–284, 2004.

[20] N. Henderson, R. King, and R. H. Middleton. An application of gaussian mixtures: Colour segmenting for the four legged league using hsi colour space. *RoboCup 2007: Robot Soccer World Cup XI*, pages 254–261, 2008.

[21] J. J. Hull. Document image skew detection: Survey and annotated bibiliography. In J. J. Hull and S. L. Taylor, editors, *Document Analysis System II, World Scientific*, pages 40–64, 1998.

[22] E. Kavallieratou, N. Fakotakis, and K. G. Skew angle estimation for printed and handwritten documents using the wigner-ville distribution. *Image and Vision Computing*, 20:813–824, 2002.

[23] F. Kleber, M. Diem, and R. Sablatnig. Document reconstruction by layout analysis of snippets. In *IS&T/SPIE Electronic Imaging, forthcoming*, St. Jose, California, USA, 2010.

[24] G. Leedham, C. Yan, K. Takru, J. H. N. Tan, and L. Mian. Comparison of some thresholding algorithms for text/background segmentation in difficult document images. *Document Analysis and Recognition, International Conference on*, 2:859, 2003.

[25] R. E. Lewand. *Cryptological Mathematics*. The Mathematical Association of America, 2005.

[26] R. D. Lins and B. T. Ávila. A new algorithm for skew detection in images of documents. In A. C. Campilho and M. S. Kamel, editors, *ICIAR (2)*, volume 3212 of *Lecture Notes in Computer Science*, pages 234–240. Springer, 2004.

[27] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[28] C. Mancas-Thillou and B. Gosselin. Color text extraction from camera-based images the impact of the choice of the clustering distance. In *ICDAR '05: Proceedings of the Eighth International Conference on Document Analysis and Recognition*, pages 312–316, Washington, DC, USA, 2005. IEEE Computer Society.

[29] B. Nickolay and J. Schneider. *Virtuelle Rekonstruktion "vorvernichteter" Stasi-Unterlagen. Technologische Machbarkeit und Finanzierbarkeit - Folgerungen für Wissenschaft, Kriminaltechnik und Publizistik*, volume 21, chapter Automatische virtuelle Rekonstruktion "vorvernichtender" Stasi-Unterlagen - Machbarkeit, Systemlösung Potenziale, pages 11–28. Schriftenreihe des Berliner Landesbeauftragten für die Unterlagen des Staatssicherheitsdienstes der ehemaligen DDR (German), Berlin, 2007.

[30] T. R. Nielsen, P. Drewsen, and K. Hansen. Solving jigsaw puzzles using image features. *Pattern Recogn. Lett.*, 29(14):1924–1933, 2008.

[31] C. Papaodysseus, T. Panagopoulos, M. Exarhos, C. Triantafillou, D. Fragoulis, and C. Doumas. Contour-shape based reconstruction of fragmented, 1600 bc wall paintings. *Signal Processing, IEEE Transactions on*, 50(6):1277–1288, Jun 2002.

[32] G. Peake and T. Tan. A general algorithm for document skew angle estimation. In *ICIP97*, pages 230–233, 1997.

[33] M. Prandtstetter and G. R. Raidl. Meta-heuristics for reconstructing cross cut shredded text documents. In *ACM: to appear in Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'09)*, 2009.

[34] M. S. Sagiroglu and A. Ercil. A texture based matching approach for automated assembly of puzzles. In *Proc. 18th International Conference on Pattern Recognition ICPR 2006*, volume 3, pages 1036–1041, 2006.

[35] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. *Pattern Recognition*, 33:225–236, 2000.

[36] J. Schneider and B. Nickolay. Automatische virtuelle rekonstruktion vernichteter dokumente. *Fraunhofer FUTUR*, 2:6–7, 2006.

[37] J. Schneider and B. Nickolay. The stasi puzzle. *Fraunhofer Magazine, Special Issue*, 1:32–33, 2008.

[38] P. D. Smet. Reconstruction of ripped-up documents using fragment stack analysis procedures. *Forensic Science International*, 176(2-3):124 – 136, 2008.

[39] P. D. Smet, J. D. Bock, and W. Philips. Semiautomatic reconstruction of strip-shredded documents. In *Proc. of SPIE -IS&T Electronic Imaging "Image and Video Communications and Processing 2005"*, pages 239–248, 2005.

[40] T.-H. Su, T.-W. Zhang, H.-J. Huang, and Y. Zhou. Skew detection for chinese handwriting by horizontal stroke histogram. In *ICDAR '07: Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*, pages 899–903, Washington, DC, USA, 2007. IEEE Computer Society.

[41] C. Sun and D. Si. Skew and slant correction for document images using gradient direction. *Document Analysis and Recognition, International Conference on*, 0:142, 1997.

[42] M. Tkalcic and J. Tasic. Colour spaces: perceptual, historical and applicational background. In *EUROCON 2003. Computer as a Tool. The IEEE Region 8*, volume 1, pages 304–308 vol.1, Sept. 2003.

[43] R. Tybon. *Generating Solutions to the Jigsaw Puzzle Problem*. PhD thesis, Griffith University, Australia, 2004.

[44] A. Ukovich and G. Ramponi. Features for the reconstruction of shredded notebook paper. *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, 3:III–93–6, Sept. 2005.

[45] J.-C. Wu, J.-W. Hsieh, and Y.-S. Chen. Morphology-based text line extraction. *Mach. Vision Appl.*, 19(3):195–207, 2008.

[46] F.-H. Yao and G.-F. Shao. A shape and image merging technique to solve jigsaw puzzles. *Pattern Recogn. Lett.*, 24(12):1819–1835, 2003.

[47] B. Yu and A. K. Jain. A generic system for form dropout. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(11):1127–1134, 1996.

[48] Y. Zheng, C. Liu, X. Ding, and S. Pan. Form frame line detection with directional single-connected chain. *Document Analysis and Recognition, International Conference on*, 0:0699, 2001.