Text Classification and Document Layout Analysis of Paper Fragments

Markus Diem, Florian Kleber and Robert Sablatnig
Computer Vision Lab
Vienna University of Technology, Austria
Email: diem@caa.tuwien.ac.at

***Abstract*—In general document image analysis methods are pre-processing steps for Optical Character Recognition (OCR) systems. In contrast, the proposed method aims at clustering document snippets, so that an automated clustering of documents can be performed. Therefore, words are classified according to printed text, manuscripts, and noise. Where, the third class corrects falsely segmented background elements. Having classified text elements, a layout analysis is carried out which groups words into text lines and paragraphs. A back propagation of the class weights - assigned to each word in the first step - enables correcting wrong class labels. The proposed method shows promising results on a dataset consisting of document snippets with varying shapes, content writing and layout. In addition, the system is compared to page segmentation methods of the ICDAR 2009 Page Segmentation Competition.**

***Keywords*-local features; text classification; layout analysis;**

I. Introduction

Text localization, text classification and layout analysis are typical pre-processing steps for Optical Character Recognition (OCR) systems. Since these methods allow for an analysis of document images, they are additionally applied for indexing digitized images and clustering of documents according to their content. The methods presented in this paper are applied to cluster document fragments.

In total, 600 million-odd snippets of Stasi documents were discovered after the fall of the Berlin Wall [10]. The documents were fragmented in 1989 when Stasi officers tried to destroy secret files. The data considered consists of manually torn documents with German, English, and Russian text. Thus, snippets have irregular shapes and their content varies from two words up to hundreds of words. Additionally machine printed and handwritten text is present. The dataset contains documents which are carbon copies, colored paper, lined or checked paper, or old fashioned copies.

To handle such amounts of document fragments an automated clustering based on the described features can be performed. Features for document clustering include amongst others, the paper color, the writing color, the background texture (e.g. lined/checked), text localization, text classification, and layout analysis [6]. In this paper the last two methods are discussed in detail.

For the text classification three classes are introduced, where the first two classes (*print, manuscript*) distinguish between machine printed and handwritten text, the third class (*noise*) detects falsely segmented background elements. The classification is based on so-called Gradient Shape Features (GSF) which can deal with noisy text. Multiple Support Vector Machines (SVM) are trained for the final class decision. Subsequently a layout analysis is performed on word blobs which groups words into text lines and paragraphs. A global voting finally corrects false class decisions based on neighboring words.

This paper is organized as follows. The subsequent section discusses current state-of-the-art methods that deal with text classification. Then, Section III details the proposed method. Finally, an evaluation on real world data is presented in Section IV.

II. Related Work

Common document analysis steps include skew estimation [11], document binarization [12], text line extraction [4], text classification, and layout analysis. These processing steps are, on the one hand, needed to perform OCR of documents. On the other hand, they allow for structuring digitized documents with respect to their content. In our case, document analysis aims at clustering document snippets that have varying supporting material, type face, and layouts for the automated clustering of document fragments.

An early work on text classification was done by Kuhnke et al. [8]. They try to distinguish between machine printed and hand written characters in order to support OCR. Therefore, line features such as the straightness of lines and symmetry features are extracted and classified by a neural network.

Kandan et al. [7] classify text into hand written and machine printed characters as a pre-processing step of OCR. They extract invariant moments from the binary image which are classified by means of a SVM. Subsequently a voting scheme based on delaunay triangulation improves the classification performance.

Recently Chanda et al. [5] proposed a text classification method applied to torn documents which is capable to identify noise, hand written and printed text. They implement a two tier approach where text and non text elements are identified based on Gabor filter and by means of directional features.

A remarkable approach was proposed by Zheng et al. [14] which identifies handwritten and machine printed text

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.175

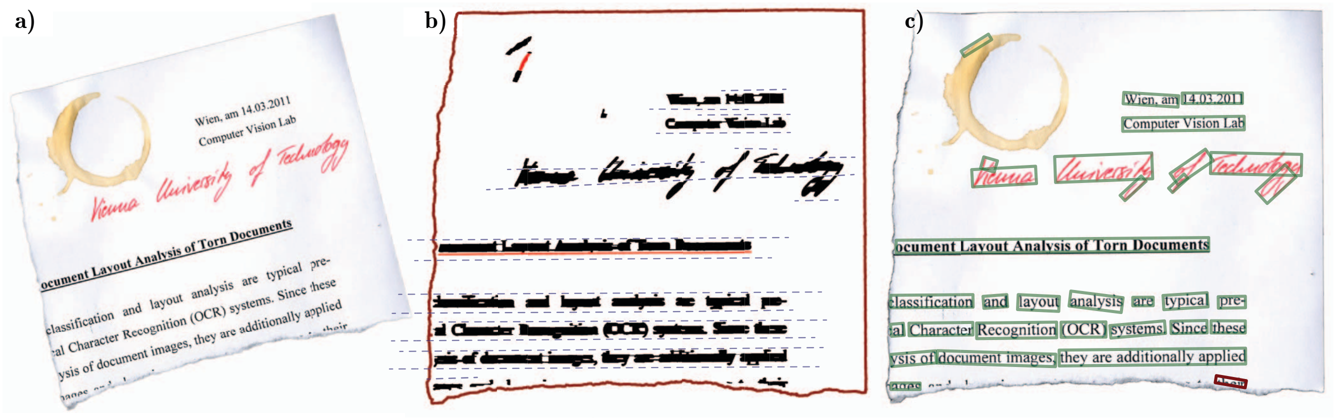

Figure 1. Input image a), word blobs obtained by eroding the LPP image b) the light blue dashed lines are text lines which separate fused blobs while red (gray) blobs denote lines which are detected in the segmented image. The final minimum area rectangles are shown in c). Note that solely one text blob is classified falsely.

in noisy images. They extract a total of 140 features that capture structure, stroke properties and texture in order to identify noise, printed and hand written text. After feature selection, 31 features are left which are classified by means of a SVM. In order to improve the classification results, a MRF models the geometrical structure of all classes and corrects the words' class labels.

III. Methodology

To handle amounts of document fragments in the order of millions layout analysis permits an automated clustering as a preprocessing step for further analysis. Hence, features that describe the content of document fragments, such as text classification, background texture, and the layout, can be determined to cluster documents according to their subject.

Pre-processing steps such as binarization or skew estimation, which are needed for the subsequently introduced text classification are shown in Figure 1. In order to localize and classify text regions, words are estimated by means of Local Projection Profiles (LPP) [2]. Then, automatically detected text lines split word blobs which are falsely merged between text lines (dashed lines in Figure 1 b)). Text decorations such as underlines are additionally removed in order to improve the text localization (red/gray lines in Figure 1 b)). Finally, minimum area rectangles are found by means of Rotating Calipers [13] (see Figure 1 c)). Minimum area rectangles are the data-structure for all subsequent processing steps since they can be stored efficiently while still being a close approximation to words.

Having detected possible word candidates, Gradient Shape Features (GSF) which are described in the subsequent section are computed for each character. However, an accurate character localization is not needed since it is rather desired to capture local structure than individual characters. Besides, character segmentation is still a challenging task when manuscript images are considered. Thus, characters are estimated by squared windows which fit into the minimum area rectangle. This square is shifted along the rectangle's principle axis in order to compute features of overlapping image regions. This interest region detection additionally guarantees that features robust against changes are extracted in the word's scale.

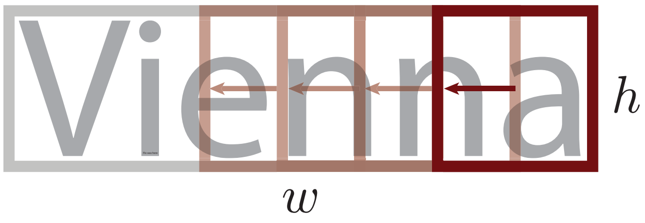

Figure 2. The gray rectangle shows the minimum area rectangle. The dark red square represents the character estimation. Having computed a feature within a square, it is shifted by $h/2$ in order to calculate the feature of the subsequent character.

For each character window, a GSF is computed which is classified by multiple SVMs (see Section III-B). Section III-C discusses the layout analysis with the global voting scheme.

A. Gradient Shape Features

The proposed features for font classification are based on Shape Context features proposed by Belongie et al. [3]. However, they tolerate failures of previous processing steps since the feature extraction itself is not based on the binary image. As proposed by Mikolajczyk et al. [9], the features are robust against all anticipated transformations including changes of the word's scale, rotation, and illumination (contrast). They are not robust with respect to affine transformations which improves their discriminativity.

The gradient magnitude image is the basis for the feature computation. As previously mentioned, the feature detection

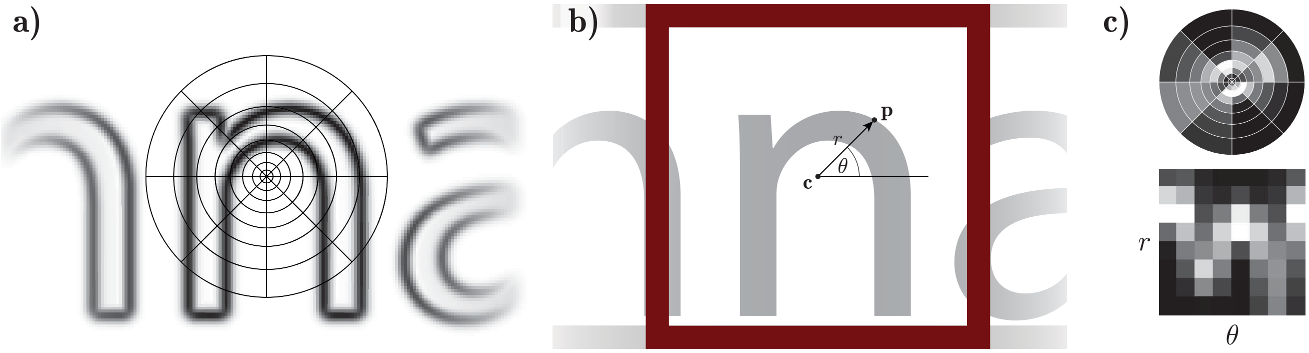

Figure 3. Log-polar grid on an inverted gradient magnitude image a), the log-polar coordinates of a pixel b) and the resulting feature vector c).

is based on character windows which are sliding windows that fit into the word's minimum area rectangle. In order to compute a GSF, solely pixel within the current character window are observed. First, the pixel coordinates are computed relative to the center $\mathbf{c}$ of the character window which is the point of origin in the log-polar coordinate system (see Figure 3 a)). Then, the log-polar vector $\mathbf{p} = (r, \theta)$ can be computed by:

$$r = \log \sqrt{x^2 + y^2} \quad (1)$$

$$\theta = \tan^{-1} \frac{y}{x} \quad (2)$$

with x, y being the relative coordinates of the current pixel. Figure 3 b) illustrates the character window and log-polar coordinates of a relative pixel vector $\mathbf{p}$. The word's dominant orientation θ_w is subtracted from the angular coordinates in order to achieve robustness with respect to rotation.

The distribution of the area of bins with increasing radius is not linear which leads to an inhomogeneously distributed gradient histogram: Bins near to the center have a lower area than those at the border. Thus, the feature's rows (see Figure 3 c) need to be normalized according to their area.

Finally, a feature consisting of 64 bins (8 radial and 8 angular bins) is created which locally captures the gradient magnitude robust against orientation, scale and contrast changes. The proposed feature qualifies for text classification since it captures the stroke width, the stroke's straightness and the appearance of junctions.

B. *Classification*

In order to classify the previously introduced features, one-against-all tests are performed. Therefore, one SVM with a Radial Basis Function (RBF) kernel is trained per class using manually annotated groundtruth data. Thus, each classifier decides if a feature belongs to the class trained (e.g. manuscript) or not. In our case, the training set consists of 56 document snippets, resulting in $\approx$ 10000 training features. The SVM's parameters (γ, C) – where γ controls the RBF kernel and C is the SVM's cost – are determined by a cross validation with a logarithmic parameter grid.

The one-against-all scheme assigns not only a class label but also a weight which indicates the distance from the current feature to the hyperplane. As previously mentioned, n features are computed per word blob. Thus, in order to assert a class label to a word blob, the weights of all features – belonging to the current blob – are accumulated.

C. *Layout Analysis*

Text clustering aims at grouping the previously classified word blobs. Therefore, words are clustered according to text lines and paragraphs. The former groups words within text lines, while the latter detects paragraphs, headings or single lines.

Text Clustering: In order to group the detected word blobs according to text lines and paragraphs, the minimum area rectangle of each word blob is taken into account. Its major axis is extended by a so-called *fuse factor*. Then, a fusing test is performed with all remaining minimum area rectangles, that are not extended. If a corner or a midpoint of the rectangle's sides lies within the currently observed rectangle, a potential fusing candidate is found.

Having found a fusing candidate, the minimum area rectangle of both rectangles is computed. Both word blobs are then assigned as children and the clustering is carried out with the newly created rectangle.

Global Voting: As soon as the words are grouped according to text lines and paragraphs, the class labels are re-computed. In order to assign a class label to text lines and paragraphs, the weight histograms – established by the SVMs – of its children are taken into account. The class label corresponding to the maximal bin in the accumulated weight histogram gets assigned to the text line or paragraph.

In order to improve the text classification, a back propagation corrects falsely classified words. Thus, the weights of the parent's histogram are voted against the weights of its children. If the maximum bin changes, a new class label is assigned to the respective child. This technique especially improves the classification performance, since a global class decision is added to the local class decision

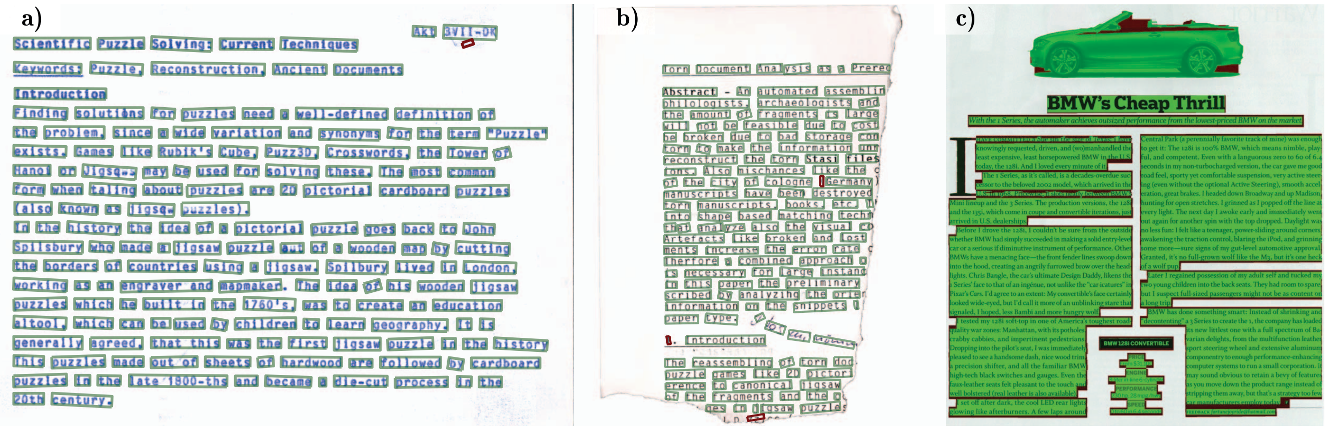

Figure 4. A carbon copy (a), a machine printed snippet with annotations (b) and page from the PRImA database (c). The light (green) rectangles in (a,b) indicate correctly classified text while dark (red) rectangles mark false classification results. In (c), light (green) areas illustrate true positives, while dark (red) areas indicate false positives and false negatives.

and weights are propagated rather than hard class decisions. Additionally, false class labels have low weights in general and are therefore corrected by neighboring words.

IV. Results

The proposed method was evaluated on real world data consisting of 446 fragmented Stasi files. The data is particularly challenging because of its great variety. Thus it comprises snippets with varying area, background, and layout. The documents were written by varying type writers, scribes, and ink colors. Additionally, old fashioned copies with background clutter and noisy character borders are present. The paper fragments have a mean area of $42.4\ cm^2$ with a standard deviation of $\pm 37.1\ cm^2$ where an unsevered Din A4 page has $623.7\ cm^2$. The original snippets must not be published due to privacy, so the examples given in Figure 4 should reasonably capture the challenges of the dataset. The snippet in Figure 4 (a) shows a carbon copy, the second snippet (b) illustrates machine printed text with annotations, and the third image (c) is a sample page from the PRImA database. The light (green) rectangles indicate correct classification results while the dark (red) rectangles mark falsely classified words.

A. Text Classification

In order to evaluate the proposed method, the dataset was manually tagged. In other words, each word was annotated according to its class (*print, manuscript*) while background was left blank. Table I shows the confusion matrix of all three classes. It can be seen that *noise* has the lowest precision (63%). This can be attributed to the fact that some snippets contain bleed-through text. These text areas where annotated as noise, however their features are similar to noisy text areas since mirror-inverted characters are present. In addition, the confusion matrix shows that hardly any text (0.5% and 1.8%) is classified as noise. Machine printed text is recognized best (94.5%) by the proposed method.

	predicted			
	noise	print	manuscript	#
noise	**0.625**	0.065	0.310	245
print	0.005	**0.945**	0.050	2180
manuscript	0.018	0.044	**0.938**	2034
	200	2166	2093	4459

Table I
The rows of the confusion matrix show the groundtruth labels, while the columns represent predicted labels (e.g. 4.4% of the manuscript is falsely classified as printed text).

In order to show the improvements of the global voting discussed in Section III-C, the system was evaluated on the same dataset without global voting. The classification performance is improved by 4.8% if a voting based on the word's neighbors is performed. Considering solely the text classes, global voting improves the performance by 5.1%. On real world data, a precision of 0.924 is achieved.

Unfortunately, there exists no public dataset that allows for a comparison of different text classification methods. However, the performance gained by the proposed method is comparable to approaches presented by other authors [5], [7].

B. Layout Analysis

An evaluation on the PRImA database [1] was performed, in order to compare the proposed method with current stat-of-the-art page segmentation methods. This dataset, which is part of the PAGE framework, was the basis of the ICDAR2009 Page Segmentation Competition [1]. The dataset consist of 55 document images, including newspapers with complex layouts or scientific papers. Thus, the documents contain images, charts and tables while handwritten text is not present.

In order to evaluate our algorithm on the PRImA dataset, the methodology presented in Section III had to be adopted. Images within documents result in large descriptors if the character estimation is performed. That is why, GSF descriptors are computed at locations found by the Difference-of-Gaussians (DoG) interest point detector. In order to assign class labels to words or images, all weight histograms of interest points, which are located within a segmented blob, are accumulated. Additionally, a new classifier was trained, so that images, charts, text and noise are detected.

The results in Table II show that the proposed method is comparable to current state-of-the-art page segmentation methods.

	Non-text	Text	Overall
Vienna UT	**94.58**	94.35	**94.47**
Fraunhofer	75.15	**95.04**	93.14
FineReader	71.75	93.09	91.90
Tesseract	74.23	92.50	91.04
DICE	66.22	92.21	90.09
REGIM-ENIS	67.13	91.73	87.82
OCRopus	51.08	84.18	78.35

Table II
F-SCORES OF THE PAGE SEGMENTATION COMPETITION 2009 [1] COMPARED TO OUR METHOD (VIENNA UT).

V. CONCLUSION

Text classification and layout analysis of paper fragments was presented in this paper. The challenge of document analysis on paper fragments, is their varying content and the fact that methods must be capable of dealing with sparse and noisy data.

Compared to the current state-of-the-art in text classification, we employ local grayscale features which can handle noisy text, since they do not suffer from poor binarization results.

The proposed text classification method was evaluated on degraded document snippets, where it achieved a precision of 0.92. The evaluation on the PRImA dataset demonstrated not only that the methodology presented is competitive with state-of-the-art layout analysis methods but also pointed out that it can be easily adopted to other document layout analysis issues.

Currently, the system cannot deal with documents that possess text written in multiple directions (i.e. vertical axis labels, annotations) since the characters are grouped by means of LPPs. Hence, in order to improve the layout analysis, a local text orientation estimation should be performed.

ACKNOWLEDGMENT

The authors would like to thank the Fraunhofer-Institute for Production Systems and Design Technology (IPK), Berlin for supporting the work.

REFERENCES

[1] A. Antonacopoulos, S. Pletschacher, D. Bridson, and C. Papadopoulos. ICDAR 2009 Page Segmentation Competition. In *ICDAR*, pages 1370 –1374, jul. 2009.

[2] Itay Bar-Yosef, Nate Hagbi, Klara Kedem, and Itshak Dinstein. Line Segmentation for degraded handwritten historical documents. In *ICDAR*, pages 1161–1165. IEEE Computer Society, 2009.

[3] Abdel Belaïd. Recognition of table of contents for electronic library consulting. *IJDAR*, 4(1):35–45, 2001.

[4] S.S. Bukhari, F. Shafait, and T.M. Breuel. Script-Independent Handwritten Textlines Segmentation Using Active Contours. In *ICDAR*, pages 446 –450, jul. 2009.

[5] Sukalpa Chanda, Katrin Franke, and Umapada Pal. Document-Zone Classification in Torn Documents. In *ICFHR*, pages 25–30, 2010.

[6] Markus Diem, Florian Kleber, and Robert Sablatnig. Document Analysis Applied to Fragments: Feature Set for the Reconstruction of Torn Documents. In *DAS*, pages 393–400, 2010.

[7] R. Kandan, Nirup Kumar Reddy, K. R. Arvind, and A. G. Ramakrishnan. A Robust Two Level Classification Algorithm for Text Localization in Documents. In *ISVC (2)*, pages 96–105, 2007.

[8] K. Kuhnke, L. Simoncini, and Zsolt Miklós Kovács-Vajna. A system for machine-written and hand-written character distinction. In *ICDAR*, pages 811 – 814, 1995.

[9] Krystian Mikolajczyk and Cordelia Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[10] Bertram Nickolay and Jan Schneider. *Virtuelle Rekonstruktion "vorvernichteter" Stasi-Unterlagen. Technologische Machbarkeit und Finanzierbarkeit - Folgerungen für Wissenschaft, Kriminaltechnik und Publizistik*, volume 21, pages 11–28. Berlin, 2007.

[11] J. Sadri and M. Cheriet. A New Approach for Skew Correction of Documents Based on Particle Swarm Optimization. In *ICDAR*, pages 1066 –1070, jul. 2009.

[12] Bolan Su, Shijian Lu, and Chew Lim Tan. Binarization of historical document images using the local maximum and minimum. In *DAS*, pages 159–166, 2010.

[13] Godfried Toussaint. Solving Geometric Problems with the Rotating Calipers. In *In Proceedings IEEE MELECON*, pages 10–17, 1983.

[14] Yefeng Zheng, Huiping Li, and David S. Doermann. Machine Printed Text and Handwriting Identification in Noisy Document Images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(3):337–353, 2003.