Extraction of Text Touching Graphics using SURF

Sheraz Ahmed*[†], Marcus Liwicki*, Andreas Dengel*[†]
* *German Research Center for AI (DFKI)*
Knowledge Management Department,
Kaiserslautern, Germany
{firstname.lastname}@dfki.de
[†] *Knowledge-Based Systems Group,*
Department of Computer Science, University of Kaiserslautern,
P.O. Box 3049, 67653 Kaiserslautern,Germany

***Abstract*—In this paper we propose a novel part-based method for the extraction of text touching graphic components. The Speeded Up Robust Features (SURF) are used to localize the text components and distinguish them from graphics. We introduce several post-processing steps to finally detect the text. We have tested our method on a publicly available data set of architectural floor plans and on real geographical maps. On floor plans we have located more than 95 % of the text components which were not identified as text beforehand because they were touching graphic components.**

***Keywords*-Text/graphics segmentation;SURF;Text extraction**

I. Introduction

Image analysis and image understanding is an important area of research. In image analysis, text/graphics segmentation is considered as an important step, e.g., in analysis of maps, technical drawings, floor plans, etc. The aim of this process is to extract two separate layers, one containing only graphical information, the other containing only textual information. The graphics recognition community has already put a lot of effort into research on text/graphics segmentation. In general, different methods have been proposed to perform text/graphics segmentation in different scenarios.

In document images, mostly parts of text overlap with graphics. To extract the overlapping text is an important challenge in text/graphics segmentation and is still an open issue. A part-based approach for text/graphics segmentation has been recently proposed by Partha et al. [1]. While the method in [1] seems to work well on map images, a direct application to technical drawings bears some complication.

The aim of this paper is to extract all the text components which are touching graphics especially for technical drawing images. We propose the use of the Speeded Up Robust Features (SURF) in order to detect key points of interest. Those key points are then compared to key points from a template database and assigned to the categories text and non-text. In our experiments on architectural drawings we have found that more than 95 % of the characters touching graphics where correctly extracted by our method.

The remainder of this paper is organized as follows. First, Section II summarizes the work related to text/graphics segmentation in general and extraction of touching text characters in particular. Then, Section III provides an overview of the method proposed in the paper. Subsequently, experimental results are described in IV. Finally, section V concludes the paper and gives an overview of future work.

II. Related Work

Several different methods have been proposed to perform text/graphics segmentation for different scenarios.

Initially, the focus was to retrieve only the text which is not touching graphics. [2] has proposed a method for block image segmentation and text extraction in mixed text/image documents. This method has shown promising results for document images. It did not focus the text touching graphics.

[3] proposed a method to extract text strings from mixed text/graphics images of technical drawings. This method is based on connected component analysis and works fine with non touching text. Text components which are touching graphics are marked as a graphical component rather than as text. Whereas, in most technical drawings and especially in map images, text and graphics overlay.

[4] performed a vector-based segmentation of text connected to graphics. This method also focuses on touching characters using heuristics. The focus of this method is engineering drawings.

[5] proposed a text/graphics separation method for overlapping text and graphics. They start with preprocessing to separate solid graphical components and remove all dashed lines. This method is applied on map images and shows promising results.

[6] used Mellin Fourier Transform too classify characters and symbols drawn on technical drawings. A Filtering technique is used to detect touching characters and symbols. Most of the touching characters are successfully extracted by this method. A major disadvantage of this method is that it is very time consuming.

[7] proposed an improvement for the method proposed in [3]. [7] introduces some additional filters to apply on connected components. Hough transform is used to extract text touching graphics and to group characters into strings.

978-0-7695-4661-2/12 $26.00 © 2012 IEEE
DOI 10.1109/DAS.2012.39

This method improved the results, but still some touching characters which are either in start or end of word were marked as graphical components.

Furthermore, [8] improved the approach of [7]. They use color information to separate touching text from the graphics. After separation of text/graphics Hough transform is used to remove the lines from the image. Finally, pyramid segmentation is used for grouping the characters into words. This method can be used where text and graphics are occuring in different colors.

[9] used color information coupled with a graph representation. A basic assumption of the method is that the text is not touching graphical components.

[1] used the SIFT features for extraction of text touching graphics. The SVM classifier is used to extract the non touching text. To extract the touching characters SIFT features of template and image are compared. To accommodate rotation of characters shape models are used. The proposed method is tested on geographical maps images and is capable of extracting most of the touching characters.

[10] introduced an approach that is based on the sparse representation framework and two appropriately chosen discriminative dictionaries. Using each dictionary, a sparse representation of one signal type and a non-sparse representation of the other signal type are generated. Finally, text/graphics segmentation is performed by promoting the sparse representation of an input image in these two dictionaries. This method extracts some of the touching text as well.

[11] proposed a method for text/graphics segmentation in architectural floor plans. This method extends the method proposed by [7], by providing a mechanism to calculate different thresholds dynamically. This method has good accuracy and is able to extract most of the overlapping text, but can only be used for architectural floor plan images.

III. Methodology

To extract touching characters from the image, Speeded Up Robust Features (SURF) are used. The SURF is a robust, translation, rotation, and scale invariant representation method. It extracts the key points/points of interest from an image. Then each key point is represented by a 128 bit discriminative descriptor. Figure 1 shows key points extracted by SURF for some text and non text images. Each of the extracted key points contains the information about the x, y, location of the point, laplacian value, size of the feature, direction, value of hessian, and its descriptor. More details on SURF can be found in [12].

SURF has been successfully applied to object recognition [12] [13]. The main idea is to apply SURF on the images to locate the touching text from images. [1] has already used SIFT features for extraction of touching text using character templates. Features of characters template are used to localize touching characters. Using only text templates leads to many false positives. In contrast, we have used both text and non-text features to localize touching characters, as well as to reduce the number of false positive. Furthermore, the computation of SURF is significantly faster than SIFT.

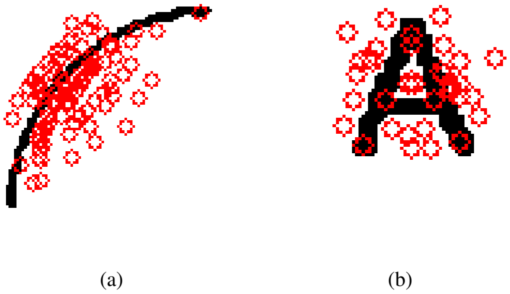


Figure 1: SURF features for text and non text component

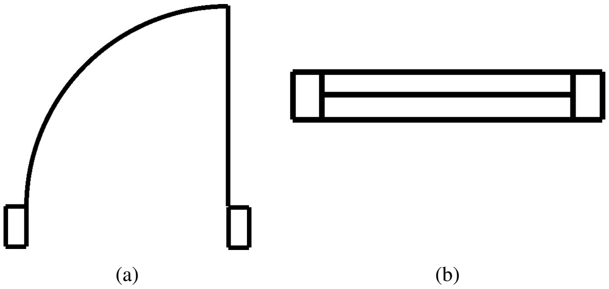


Figure 2: Example of non text component

In the methodology presented in this paper first, non-touching text components are extracted from the image. Based on the nature of image, either [7] or [11] can be used for extraction of non-touching characters. These extracted text components serve as templates for the localization of text components which touch graphics. If the number of text components extracted by [7] or [11] are very few, then reference templates of typical fonts are also used as templates. To find the font size used in the image, average height(Avg_{height}) and average width(Avg_{width}) are computed from the extracted text components. In addition to alphabet templates we also store some templates for graphical elements if available, e.g., lines, arcs, and objects,(see figure 2). These are referred as graphic template.

In next step, SURF is applied on every reference text template and all the key points and their respective descriptors are stored as reference text features. Similarly, SURF is applied on the graphics template, and extracted the key points and their descriptors are stored as reference graphics features, respectively. To reduce number of false positive, all the text descriptors are compared with graphics descriptors. Similar descriptors are removed from both reference text features and reference graphics features.

After removing ambiguous descriptors from reference features, SURF is applied on the entire graphic image where touching text needs to be localized. This results in list of key points and their respective descriptors for the graphics image. The graphic image was obtained as an output of applying [7] or [11] method on the original image. The descriptors of graphics image (containing touching characters) are compared to the reference text and graphics key points descriptors mentioned above. Finally, nearest neighbor approach is used to compare these descriptors.

If a key point's nearest neighbor is a text reference key point and the distance between the descriptors is less than $Dist_{text}$, it is marked as a text key point. Similarly, if a key point's nearest neighbor is a graphics reference key point and the distance between descriptors is less than $Dist_{graph}$, it is marked a graphics key point. $Dist_{text}$ and $Dist_{graph}$ are distance thresholds that are computed empirically after investigating the behavior on one reference image. To finally mark a key point as text or graphic, a majority voting is applied based on the neighboring key points. If a key point has more graphic key points as neighbors, it is finally marked as a graphic key point, otherwise it is marked as a text key point.

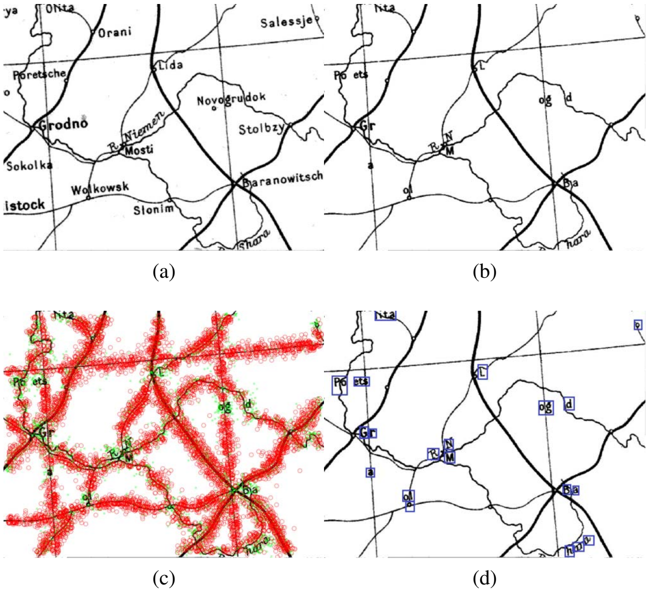
For extracting the text from the marked text key points, the bounding box of size Avg_{height} and Avg_{width} is constructed on the detected regions, and if this bounding box contain any black component it is marked as touching text.

Figure 3a[1] shows the floor plan image where all of the non-touching characters are removed using the method of [11]. After applying the nearest neighbor approach, the key points as illustrated in Fig. 3b, are extracted. Note that the red circles denote graphics key points and green circles denote text. Finally, in Fig. 3c the resulting bounding boxes are shown. As shown, they only mark the text area which was touching a diagonal line.

To investigate the behavior of proposed method, we have also applied this method on floor plans where only thick walls were removed and no text extraction is performed. This results in an image, where all of the remaining graphics as well as all text components are present. The results of our method are shown in Figs. 3g, 3h, and 3i, respectively.

IV. Evaluation

Our system is evaluated using a data set of original floor plans collected over a period of more than ten years. This data set was primarily introduced for floor plan analysis in [14] and contains 90 floor plan images. [11] has performed text/graphics segmentation evaluation on this data set. From the results of evaluation in [11], 327 characters out of 21, 737 where those which were difficult to read. Among these 327 characters, 199 characters overlay with graphics.



(a) (b)

(c) (d)

Figure 4: Map image(a) After removal of isolated characters(b) text and graphics keypoints marked after comparison

Touching characters	Number	Percentage (%)
Total	199	100
Retrieved	190	95.48
Missing	9	4.52

Table I: Touching text extraction results

Analysis of table I reveals that our system finds 95% text components which were touching graphics. If these results are combined with results of text/graphics segmentation method in [11], the overall recall of text/graphics segmentation method by [11] increases significantly.

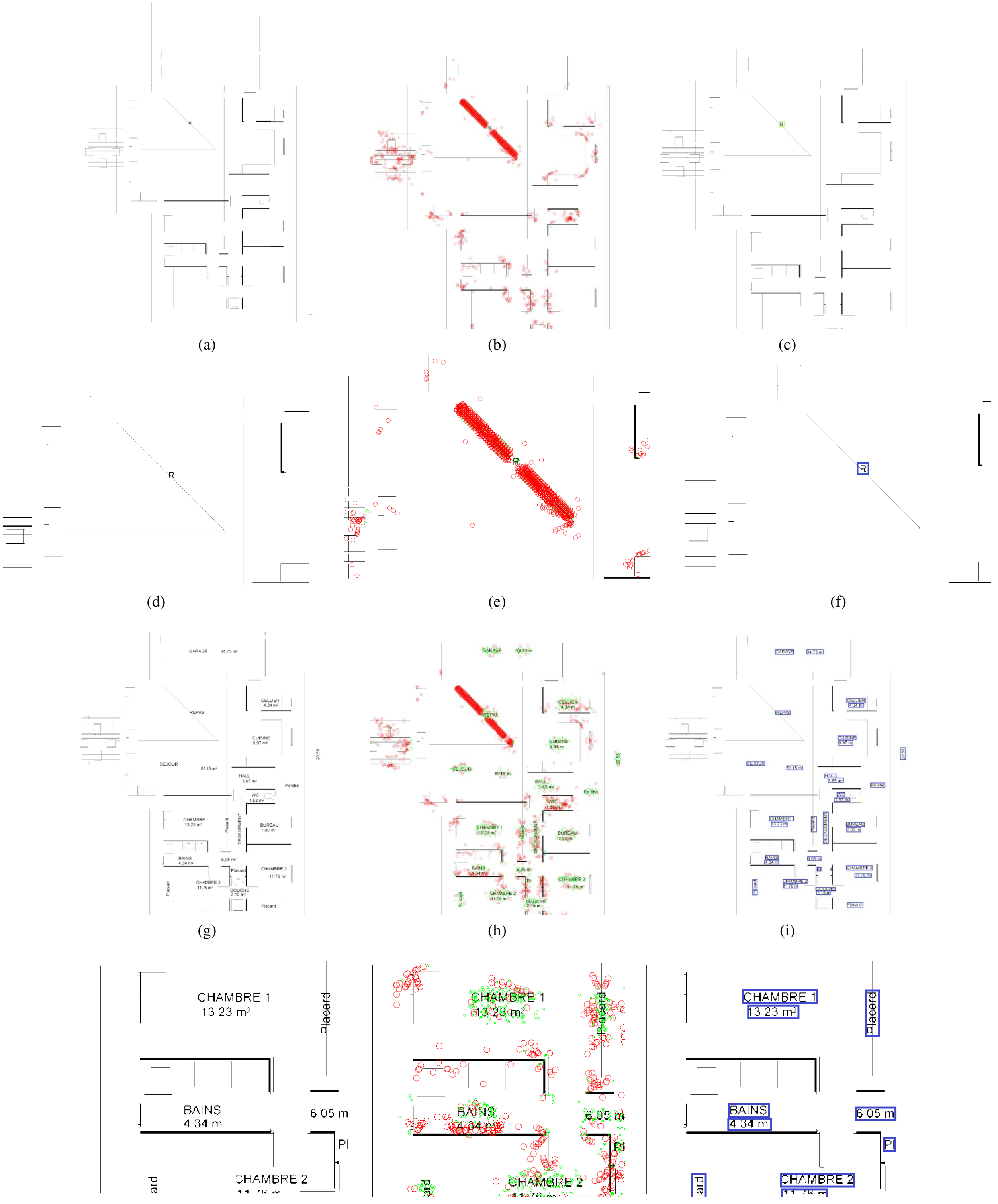
Figure 4 shows the results of our method on map image. Isolated characters are removed using [7]. In Figure 4c it can be seen that all of the touching characters are marked with green key points. It is difficult to judge if the false positives in the map image are errors or not because on the location where false positives are detected there are holes which are very similar to the character ”O”.

V. Conclusion and Future work

In this paper a part-based method for extracting text components touching graphics is proposed. The method extracts all SURF keypoints of a questioned image and compares them with the keypoints of reference templates from characters and non-characters.

In our experiments on real floor plan images we have observed that more than 95 % of the characters were correctly detected. In fact these characters were actually the

[1] Note that in these figures, a zoomed version of an interesting zone in the figure is always shown below (e.g., in Fig. 3d for Fig. 3a)



(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

(i)

(j)

(k)

(l)

Figure 3: Example of localization of text points on floor plan image without external walls. Floor plan without thick walls and touching text(a), extracted text and graphics key points(b), detected text locations(c), (d)(e)(f) are zoomed versions of (a)(b)(c) respectively. Floor plan with non touching and touching characters without thick walls(g), extracted text and graphics key points (h), detected text locations(I)

problematic characters in the previous text/graphics segmentation method [11]. Therefore, we propose to use the part-based strategy as a post processing method for text/graphics segmentation methods existing in literature, e.g., [7], [11], and [3]. This method increases the overall recall of the existing methods, as remaining touching characters can be found. Note that it can also be used to increase precision of existing methods as it can locate graphical elements. We will investigate this behavior on large data sets in future. Another idea is to use the part-based method as a text/graphics segmentation method alone.

Acknowledgment

We would like to thank Joseph Llados and the members of the CVC for providing us the reference data set. We would also like to thank Seiichi Uchida for healthy discussions. This work was financially supported by the ADIWA project.

References

[1] P. Roy, U. Pal, and J. Lladós, "Touching Text Character Localization in Graphical Documents Using SIFT," in *Graphics Recognition. Achievements, Challenges, and Evolution*, ser. Lecture Notes in Computer Science, J.-M. Ogier, W. Liu, and J. Lladós, Eds. Berlin, Heidelberg: Springer Berlin / Heidelberg, 2010, vol. 6020, ch. 18, pp. 199–211.

[2] F. M. Wahl, K. Y. Wong, and R. G. Casey, "Block segmentation and text extraction in mixed text/image documents," *Computer Graphics and Image Processing*, vol. 20, no. 4, pp. 375 – 390, 1982.

[3] L. Fletcher and R. Kasturi, "A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 910–918, 1988.

[4] D. Dori and L. Wenyin, "Vector-based segmentation of text connected to graphics in engineering drawings," in *Advances in Structural and Syntactical Pattern Recognition*, ser. Lecture Notes in Computer Science, P. Perner, P. Wang, and A. Rosenfeld, Eds. Springer Berlin / Heidelberg, 1996, vol. 1121, pp. 322–331.

[5] R. Cao and C. L. Tan, "Separation of overlapping text from graphics," in *Proceedings. Sixth International Conference on Document Analysis and Recognition, 2001.*, 2001, pp. 44 –48.

[6] S. Adam, J.-M. Ogier, and C. Cariou, "Multi-scaled and multi oriented character recognition: an original strategy," in *Fifth International Conference on Document Analysis and Recognition, ICDAR 1999*. IEEE Computer Society, 1999, pp. 45–48.

[7] K. Tombre, S. Tabbone, L. Plissier, B. Lamiroy, and P. Dosch, "Text/graphics separation revisited," in *Document Analysis Systems V*, ser. Lecture Notes in Computer Science, D. Lopresti, J. Hu, and R. Kashi, Eds. Springer Berlin / Heidelberg, 2002, vol. 2423, pp. 615–620.

[8] P. P. Roy, J. Llados, and U. Pal, "Text/Graphics Separation in Color Maps," *International Conference on Computing: Theory and Applications*, vol. 0, pp. 545–551, 2007.

[9] R. Raveaux, J.-C. Burie, and J.-M. Ogier, "A colour text/graphics separation based on a graph representation," in *ICPR*, 2008, pp. 1–4.

[10] T. V. Hoang and S. Tabbone, "Text extraction from graphical document images using sparse representation," in *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, ser. DAS '10. New York, NY, USA: ACM, 2010, pp. 143–150.

[11] S. Ahmed, M. Weber, M. Liwicki, and A. Dengel, "Text / Graphics Segmentation in Architectural Floor Plans," in *11th International Conference on Document Analysis and Recognition.*, 2011.

[12] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Comput. Vis. Image Underst.*, vol. 110, pp. 346–359, June 2008.

[13] D.-N. Ta, W.-C. Chen, N. Gelfand, and K. Pulli, "Surftrac: Efficient tracking and continuous object recognition using local feature descriptors," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition,*, vol. 0, pp. 2937–2944, 2009.

[14] S. Macé, H. Locteau, E. Valveny, and S. Tabbone, "A system to detect rooms in architectural floor plan images," in *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, ser. DAS '10. New York, NY, USA: ACM, 2010, pp. 167–174.