Northumbria
University
NEWCASTLE

A Connected Components Based Layout Analysis Approach for Educational Documents

Ruiying Liu[1], Shenbao Yu[1], Fan Yang[1], Yinghui Pan[2] and Yifeng Zeng[3]*
[1]*Department of Automation, Xiamen University*
[2]*College of Computer Science and Software Engineering, Shenzhen University*
[3]*Department of Computer and Information Sciences, Northumbria University*

***Abstract*—Layout analysis, which aims to detect and categorize areas of interest on document images, is an increasingly important part in document image processing. Existing researches have conducted layout analysis on various documents, but none has been proposed for documents yielded from teaching, i.e. exam papers and workbooks, which are worth studying. In this paper, we propose a novel layout analysis system to achieve two tasks for workbook pages and exam papers respectively. On one hand, we segment text and non-text areas of workbook pages. On the other hand, we extract regions of interest on exam papers. Our system is based on connected component (CC) analysis, specifically, it extracts geometric features and spatial information of CCs to recognize page elements. We carried out experiments on images collected from real-world scenarios, and promising results confirmed the applicability and effectiveness of our system.**

***Index Terms*—Layout Analysis, Connected Component Analysis, Digital Image Processing**

I. Introduction

It is inconvenient to store, retrieve and analyze paper documents. Now people are increasingly inclined toward paperless office, promoting rapid development of document image processing. Layout analysis, also known as page segmentation or page object detection, is a fundamental step of document image processing. It aims to automatically extract regions of interest on document images and classify them without text recognition or human supervision [1]. A great performance of layout analysis boosts accuracy and efficiency of subsequent processes, especially OCR (optical character recognition). In addition, layout information can help to build a logical relationship among different parts for a more comprehensive document understanding.

Several challenges of document images processing are as follows. First, the inter-class variance is small while the intra-class variance is large among different elements. For example, most document elements have the same black color, preventing distinguishing with color features. On the contrary, elements of the same class have great differences from each other due to arbitrary orientations, large scale changes and different fonts. Second, layout styles and kinds of page elements can vary a lot with categories of documents. For example, documents are either double column or single column; and some documents may contain hybrid elements. Third, there is still a huge lack of data for research. These challenges cause tremendous hardship to propose a universally general method. Currently most existing researches mainly focus on one specific kind of document, such as financial documents [2], scanned scientific papers [3], etc.

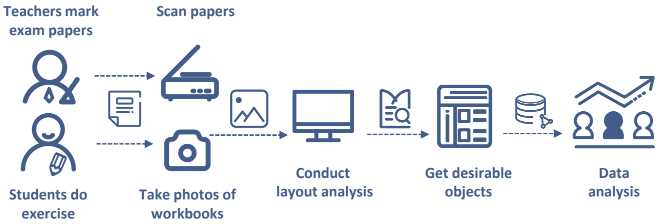


Fig. 1: Two application scenarios we envision for our layout analysis system: teachers scan the exam papers after marking and students take pictures of the workbooks after finishing practicing. These images will enter our layout analysis system, which can detect desirable objects for subsequent data analysis.

There is a large literature in layout analysis of various documents, but to the best of our knowledge, there is no work devoting to exam papers or workbook pages yielded from educational teaching though they are ubiquitous. We consider two practical application scenarios shown in Fig. 1. One scenario is that teachers mark exam papers and the other one is that students practice at home. It is easy to scan or take photos to get document images. A layout analysis system receives these scanned exam papers or workbook page photos and then outputs regions of interest with category labels. With the layout analysis results, people can further recognize the detected areas and mine data from them for many applications, for example, building student portraits to customize more efficient learning plans.

The layout analysis methods can be basically classified into three types: bottom-up, top-down and hybrid methods [4]. Bottom-up methods start from small elements like pixels and then gather homogeneous elements into zones of corresponding classes. Top-down methods [5], [6] start from the entire page or few big regions, and then constantly divide regions into smaller zones based on texture features or homogeneous rules until no zone can be divided. Top-down methods are easy to be implemented, but are incapable of documents with complex layouts. By contrast, bottom-up methods are able to handle more kinds of documents including namely non-Manhattan layout pages and so on, but they need higher computational cost as an exchange. Hybrid methods integrate these two methods, and one of the most representative methods is CC analysis [7], [8], [9], [10], [11], [12]: CCs are detected from the the entire images first, and then researchers analyze these CCs to acquire areas of interest. Hybrid methods combine the benefits of bottom-up and top-down methods, they can handle a variety of documents with relatively fast speed.

We extend CC analysis to process workbooks and exam papers with rule-based heuristics. There are four basic steps: binarization, morphological operations, contour extraction [13] and CC analysis. First, binarization segments the input image into foreground contents and background, generating a binary image. Second, morphological operations transform the binary images to underline desirable objects or filter out useless parts. Then we trace contours [13] to obtain CCs (there is a one-to-one correspondence between contours and CCs). We analyze geometric features and spatial relations of CCs to achieve layout analysis. In our system, we use three binarization algorithm and special morphological operations to get better binary maps for detection. It worth noting that we always add a Gaussian blur filter in advance of binarization to reduce noises. Our system performed quite well on real data, strongly proving its applicability and effectiveness.

Our main contributions are as follows:

1) Facing unusual challenges, we propose a novel layout analysis system for exam papers and workbook pages. To the best of our knowledge, this is the first work processing these documents which has great significance to education.
2) Our system only uses basic digital image processing techniques to trade off efficiency and accuracy. It does not require any complicated model or expensive computational cost, so our system is convenient to be implemented in practice.
3) We collected several exam papers and workbook pages from real-world scenarios for experiments. The layout analysis results were satisfactory with only a little errors occurring, which is acceptable, and the validity of our proposed system was verified.

The rest of this paper is organised as follows: Section II and Section III introduce processes of layout analysis on workbook pages and exam papers in detail respectively. Section IV describes experiments we conducted and shows the results. Finally, Section V draws a conclusion and gives some possible improvements.

II. Text and Non-text Segmentation of Workbook Pages

First, we illustrate the framework for segmenting text and non-text areas of workbook pages. Although text segmentation has been widely studied, it is still challenging considering that workbooks mix Chinese characters, English alphabets and mathematical symbols. In addition, the layouts are more complex, making it impossible to utilize alignment information [11]. Existing researches segment with size information at first [14], [11] to get coarse segmentation, and then further extract texts from small CCs. We use the two-stage framework by convention and our workflow is shown as Fig. 2. We add some special morphological transformations for better segmentation, and use only Harris corners for identification of small CCs.

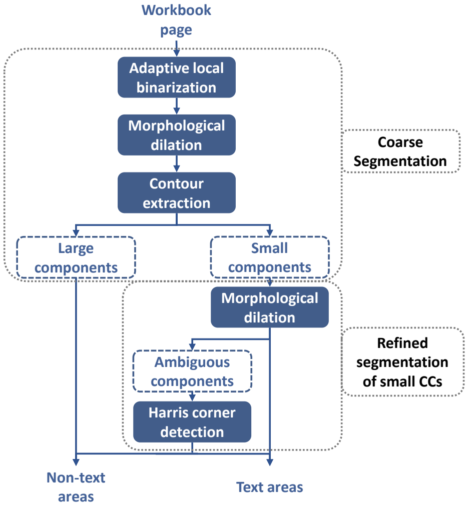


Fig. 2: The workflow of segmenting text and non-text areas on workbooks. We divide CCs based on size information first. Large CCs are classified into non-text areas while small CCs are further identified.

A. Coarse Segmentation

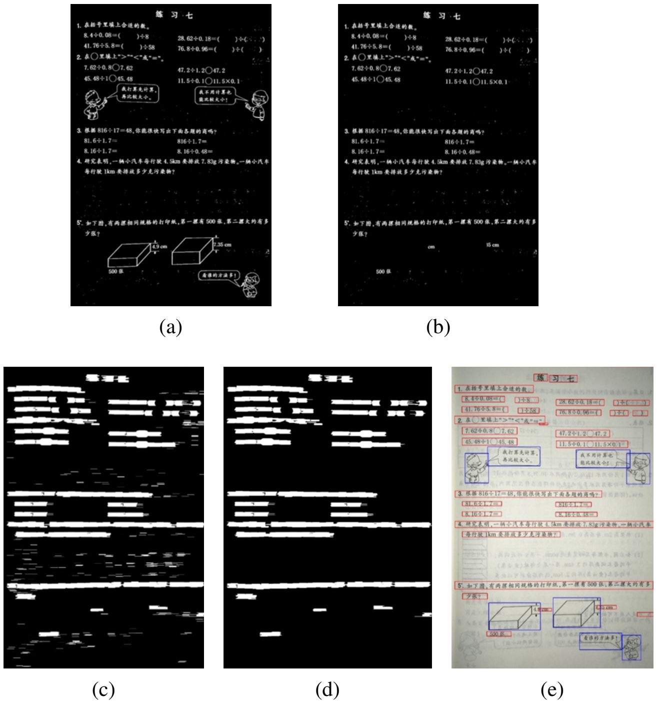
As mentioned in the previous section, the workbook pages are photos, so the performance of binary transformation may suffer from uneven illumination. Thus we use an adaptive local binarization approach [15] which is robust to illumination changes.

To better distinguish small texts between large graphics, existing studies usually fill contours by different means: filling the whole regions according to different rules [11], [12], [9] or performing hole-filled morphological closing [10]. However, there are two problems in our scenario that filling is inapplicable to. First, Chinese characters usually consist of several components like radicals, so multiple CCs will be generated just for one character. Another problem is that some foreground pixels may be lost due to low quality of document images, hence a page element may be broken up into several CCs. Too many small CC pieces make difficulty on merging [16] and reduce computation efficiency. We use a morphological dilation operation with the structural element of 5×5 square to alleviate the two problems at the same time.

Fig. 3a shows a dilated binary image. We extract contours from dilated binary images and divide CCs by sizes. All large CCs are classified as non-text areas while small CCs as shown in Fig. 3b need refined identification next.

B. Refined Segmentation of Small CCs

The majority of small CCs are body texts apart from a few noises and small non-text elements. To simplify the identification, firstly we perform morphological dilation again but with a different structural element: we use a short dash of width $\tilde{h}$, the average height of all small CCs. CCs are



(a) (b)

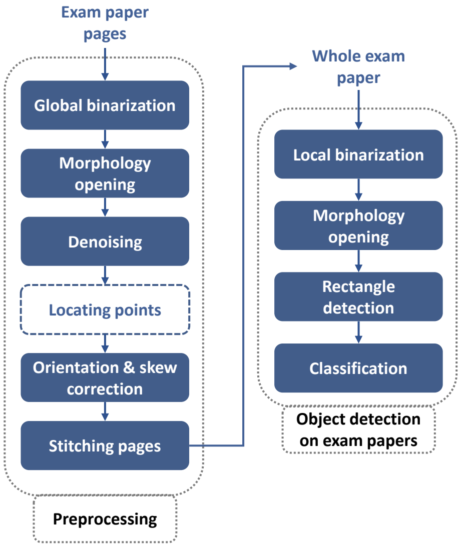
(c) (d) (e)

Fig. 3: The procedure of segmenting text and non-text areas on workbook pages. (a) shows all CCs and (b) shows small CCs only. (c) is transformed from (b) through morphological dilation, and then noises are removed on (d). The segmentation result is shown as (e) where blue and red bounding boxes label non-text areas and text lines respectively.

widened through this transformation. Specially, the laterally adjacent text components are merged into text lines, and the text areas stand out from other two kinds of small CCs, hence a simplistic approach is sufficient to separate them out. As shown in Fig. 3c and 3d, noises are easily to be removed based on their small sizes. Tiny non-text CCs and short text lines are similar in terms of size, but the outer contours of non-text components are intuitively more smooth than that of text strings. Therefore we calculate Harris corners [17] of every contour curve to distinguish them. If a curve contains more corners over certain threshold, the corresponding CC will be consider as text. Finally we obtain text and non-text areas as shown in Fig. 3e.

III. Extract Regions of Interest on Exam Papers

Different from the segmentation task for workbooks, we detect specific regions of exam papers to extract desirable information, including student information, exam information and question information. Fig. 4 shows the flow diagram. The input is two side pages, we correct them respectively and stitch them together into the whole exam paper for final detection. There are two parts to the entire process: preprocessing and detecting regions of interest on exam papers.



Exam paper pages

Global binarization

Morphology opening

Denoising

Locating points

Orientation & skew correction

Stitching pages

Preprocessing

Whole exam paper

Local binarization

Morphology opening

Rectangle detection

Classification

Object detection on exam papers

Fig. 4: The workflow of layout analysis of exam papers. We first perform preprocessing, and then detect desirable regions on exam papers.

A. Preprocessing

Our system accomplishes two main tasks during preprocessing: correction and stitching. Correction is a routine operation in document image processing including rotating images to correct orientations and eliminating probable skews or tilts. Normally the input of our system is two images of both sides of an exam paper (one side is scanned at a time) rather than a complete paper image, so we need to stitch them together.

Many efforts have been done to detect orientation and skew by projection profiles [18], [4]. This method highly relies on alignment of page components and rich texts, but exam papers have irregular layouts and no enough alignment information. It is noticed that scanned exam papers all have specific elements: there are two small black solid rectangles fixed on the upper edge of every page for locating when scanning. We call them *locating points* and try to use them for page correction.

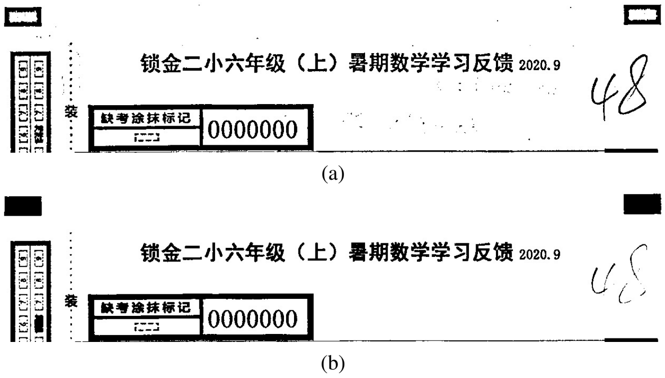

锁金二小六年级（上）暑期数学学习反馈 2020.9
缺考涂抹标记
0000000
48
(a)
锁金二小六年级（上）暑期数学学习反馈 2020.9
缺考涂抹标记
0000000
(b)

Fig. 5: Comparison between the results of the Gaussian adaptive binarization (a) and the Otsu's method (b). We get complete locating points on the top of (b).

We perform the classic global thresholding named the Otsu's method [19] in binarization. Though local adaptive thresholding algorithms usually perform better, we found that local methods cannot preserve complete locating points. As shown in Fig. 5, locating points become hollow rectangles after local binarization while the Otsu's method does not damage them. A possible explanation for this might be that local methods calculate thresholds based on the neighbor areas, and the inner areas of locating points are all black, thus quite low thresholds for inner pixels are calculated by mistake.

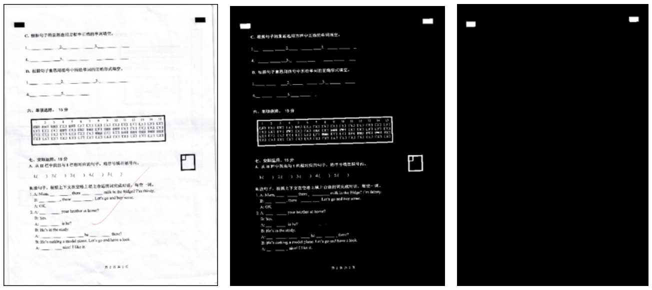

(a) (b) (c)

Fig. 6: Process of detecting locating points. (a) is the original image. (b) is the binary image of (a). After morphological erosion, we got (c) where there are only locating points left.

After binarization, we get Fig. 6b, it is too computationally cumbersome to extra only two locating points by analyzing all the components. Inspired by multiresolution morphology method [20] and its improved version [10], we found that components with thinner lines or strokes are easier to be removed by morphological opening, and the threshold of thickness depends on the size of structural elements. The locating points can be seen as extremely thick dashes, so we perform opening operation with a large structural element to extract them. We set the structural element a square with the side length of $min/90$, where min denotes the minimum between the height and width of the whole image. Finally, we get locating points left only shown as Fig. 6c. To remove possible noises, we conduct a simple post-process using some heuristic rules based on positional relationship and spacial information. For example, locating points are paired and aligned with each other, also they are very close to the edge of pages.

Then it is quite easy to correct pages with the coordinates of locating points. When pages are in the correct orientations, locating points are on the upper side. Based on this, we can rotate pages to right orientations. To eliminate skew, we calculate the slope of the straight line which connects locating points to get skew angle. Then stitching the corrected pages together, we get a complete exam paper for detection.

B. Desirable Region Detection on Exam Papers

Exam paper images may suffer from auxiliary degradations when printing and scanning, here we determine to use local binarization to generate more refined binary images. We take the Gaussian thresholding method which calculates Gaussian-weighted sum of neighbour area as the threshold to for each pixel.

On exam papers, desirable information is surrounded by similar thick rectangles. First, we try to extract thick CCs. Similar with the process of extracting locating points above, we transform images by opening to filter out all thin lines with the structural element of which the side is $min/30$. Then we identify rectangular shapes from thick CCs.

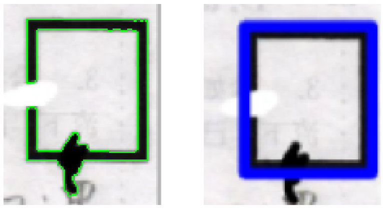

(a) (b)

Fig. 7: The example of an incomplete bold box with the contour drawn in green on (a). The whole rectangular region was extracted successfully through our rectangle detection, and the bounding box was drawn on (b).

Most methods detect rectangles based on line analysis. Research used the Hough transform [21] or edge detection with gradient [22] to extract straight lines, and then find out shapes which satisfy artificial geometric conditions, such as that two pairs of opposite sides are parallel, to recognize rectangles. But the contours of components are not always perfect due to smudges or fading on scanned papers, complicating detection on sides of a CC. So we convert to mine vertex relationships for rectangle detection.

We combine polygonal approximation [23] and corner analysis [24], [25] for rectangle detection. Polygonal approximation is to select several key points of contour to be vertices of its approximate polygon. But contours of incomplete rectangles may not be rectangular as illustrated in Fig. 7a, where the green curve represents the detected contour. To get the accurate rectangular regions, we firstly find out vertices of right angles from contour points, denoting as kps, and then calculate the convex hull h of the kps by algorithm proposed by Sklansky [26]. Finally, we find approximate polygon of h to get p. If p has four sides, which means that we find a contour with four right angles, the corresponding CC will considered as rectangle.

Ultimately, we can classify these boxes based on their geometric features and coordinates to get layout analysis results. For example, the exam information is on the upper left of page, and the student information is always on the left and the corresponding rectangular CC has a much small aspect ratio.

IV. Experiment Results

The experiments are conducted on exam papers and workbook pages respectively. Our system was implemented with C# language. The computer for experiments is equipped with

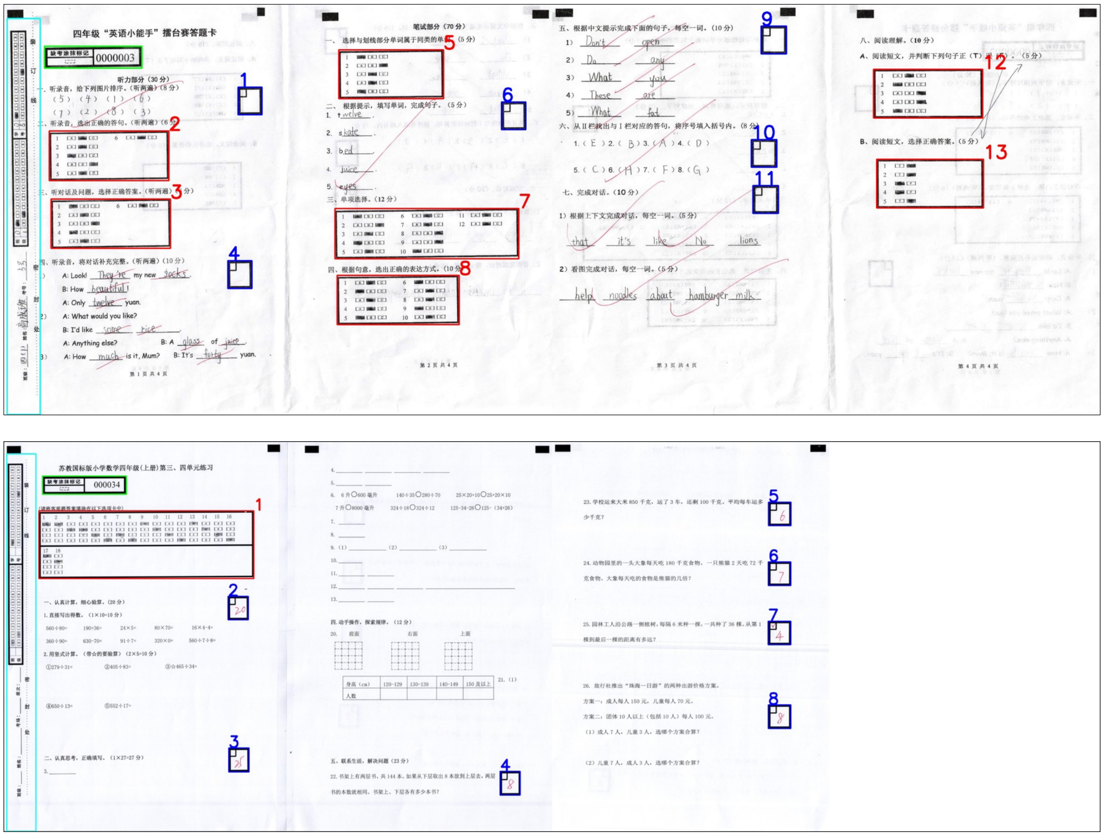


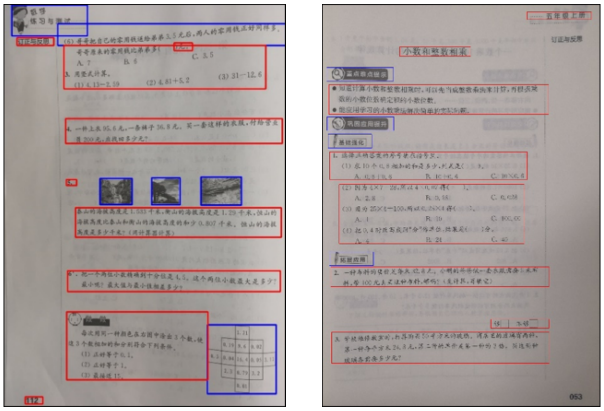
Fig. 8: Desirable regions belonging to different categories are drawn on the preprocessed images with different color bounding boxes: cyan boxes for student information region, green for exam paper information, red for answer sheets of objective questions and blue for score boxes of subjective questions. The numbers in blue and red around boxes are question numbers.

Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz with 16GB RAM. In particular, we implemented basic image process such as binarization, morphological operation and contour tracing with the awesome OpenCV library.

TABLE I: Layout analysis results of exam papers

	Exam Information	**Student Information**	**Subjective Questions**	**Objective Questions**
Pre(%)	0.9970	0.9985	0.9896	0.9916

Firstly, we study the performance of our proposed system on exam papers. We have collected 662 marked papers from elementary and junior high schools, which cover multiple subjects including Mathematics, English, Chinese, Physics and Biology. Preprocess managed to get all complete corrected papers, then the experimental task is to detect four objects on papers: exam information regions, student information regions, score boxes of subjective questions and answer sheets of objective questions. The detection precision of each object was counted. Table I shows satisfactory results: the precisions for all objects are higher than 0.98. Fig. 8 shows some examples of detected regions on exam papers. We draw bounding boxes in different colors which distinguish categories. In addition, we numbered objective and subjective questions with blue and red figures respectively.



(a)

(b)

Fig. 9: Some results of workbook image segmentation. Objects in red boxes are text regions detected by our system, and blue are non-text areas.

For experiment on workbook page detection, we took photos of 118 pages from primary school Math workbooks. Our system successfully detected all text and non-text areas of 111 pages. Fig. 9 shows some examples of detection results. Some misclassification occurs in Fig. 9a, where the first text line was classified as non-text because it is very slanted. What's more, we regard decorated text lines as illustrations, but at bottom of Fig. 9a, the decorated subtitle is included in the text area. Despite a bit of mistakes, our system performed well enough to be put into practical usage.

V. Conclusion and Discussion

We proposed a novel system to undertake layout analysis of exam papers and workbook pages based on CC analysis. Our system achieved two different tasks respectively on these two documents: text and non-text segmentation for workbook pages and desirable region detection for exam papers. Images collected from real-world scenarios were used in the experiments, and the satisfactory results proved the practical value of our system. Besides, our systems is implemented with basic digital image processing techniques, so it is lightweight and easy to be put into practical applications.

There are several possible improvements. First, some detected non-text areas containing texts, and we plan to do further extraction. Second, the handwriting detection is an interesting yet challenging task, we want to achieve it in exam paper processing. Third, our system depends on simple heuristics and it is not flexible enough, more sophisticated machine learning algorithms can help to explore a versatile and generic layout analysis system.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China (Grants No.61772442, 61836005 and 61806089).

References

[1] A. Droby, B. K. Barakat, and J. El-Sana, "Asar 2018 competition page layout analysis using fully convolutional networks," in *2018 IEEE 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR)*, 2018, pp. 161–164.

[2] R. Juge, I. Bentabet, and S. Ferradans, "The fintoc-2019 shared task: Financial document structure extraction," in *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*, 2019, pp. 51–57.

[3] X. Zhong, J. Tang, and A. J. Yepes, "Publaynet: largest dataset ever for document layout analysis," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 2019, pp. 1015–1022.

[4] G. M. Binmakhashen and S. A. Mahmoud, "Document layout analysis: a comprehensive survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 6, pp. 1–36, 2019.

[5] K. Chen, F. Yin, and C.-L. Liu, "Hybrid page segmentation with efficient whitespace rectangles extraction and grouping," in *2013 12th International Conference on Document Analysis and Recognition*, 2013, pp. 958–962.

[6] G. Lazzara, R. Levillain, T. Géraud, Y. Jacquelet, J. Marquegnies, and A. Crépin-Leblond, "The scribo module of the olena platform: a free software framework for document image analysis," in *2011 International Conference on Document Analysis and Recognition*, 2011, pp. 252–258.

[7] L. A. 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, no. 6, pp. 910–918, 1988.

[8] S. Bhowmik and R. Sarkar, "Classification of text regions in a document image by analyzing the properties of connected components," in *2020 IEEE Applied Signal Processing Conference (ASPCON)*. IEEE, 2020, pp. 36–40.

[9] S. Bhowmik, S. Kundu, and R. Sarkar, "Binyas: a complex document layout analysis system," *Multimedia Tools and Applications*, vol. 80, no. 6, pp. 8471–8504, 2021.

[10] S. S. Bukhari, F. Shafait, and T. M. Breuel, "Improved document image segmentation algorithm using multiresolution morphology," in *Document recognition and retrieval XVIII*, vol. 7874, 2011, p. 78740D.

[11] N. Vasilopoulos and E. Kavallieratou, "Unified layout analysis and text localization framework," *Journal of Electronic Imaging*, vol. 26, no. 1, p. 013009, 2017.

[12] N. Vasilopoulos, Y. Wasfi, and E. Kavallieratou, "Automatic text extraction from arabic newspapers," in *International Conference Image Analysis and Recognition*, 2018, pp. 505–510.

[13] S. Suzuki *et al.*, "Topological structural analysis of digitized binary images by border following," *Computer vision, graphics, and image processing*, vol. 30, no. 1, pp. 32–46, 1985.

[14] T.-A. Tran, I.-S. Na, and S.-H. Kim, "Separation of text and non-text in document layout analysis using a recursive filter," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 9, no. 10, pp. 4072–4091, 2015.

[15] D. Bradley and G. Roth, "Adaptive thresholding using the integral image," *Journal of graphics tools*, vol. 12, no. 2, pp. 13–21, 2007.

[16] H. Zhu and Y. Zou, "A cross-connected components-based layout analysis algorithm for chinese business card," in *2008 3rd IEEE Conference on Industrial Electronics and Applications*, 2008, pp. 2530–2534.

[17] C. G. Harris, M. Stephens *et al.*, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15, no. 50, 1988, pp. 10–5244.

[18] D. S. Le, G. R. Thoma, and H. Wechsler, "Automated page orientation and skew angle detection for binary document images," *Pattern Recognition*, vol. 27, no. 10, pp. 1325–1344, 1994.

[19] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[20] D. S. Bloomberg, "Multiresolution morphological approach to document image analysis," in *Proc. of the International Conference on Document Analysis and Recognition, Saint-Malo, France*, 1991.

[21] C. R. Jung and R. Schramm, "Rectangle detection based on a windowed hough transform," in *Proceedings. 17th Brazilian Symposium on Computer Graphics and Image Processing*, 2004, pp. 113–120.

[22] Y. Jin, X. Guan, and H. Zhang, "Gradient-direction-based rectangles and triangles traffic signs detection algorithm in natural scenes," in *2019 Chinese Automation Congress (CAC)*, 2019, pp. 1115–1120.

[23] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica: the international journal for geographic information and geovisualization*, vol. 10, no. 2, pp. 112–122, 1973.

[24] T. Wenzel, T.-W. Chou, S. Brueggert, and J. Denzler, "From corners to rectangles—directional road sign detection using learned corner representations," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 1039–1044.

[25] S. Wu, L. Gou, H. Xiong, and X. Li, "A graph-based approach for rectangle detection using corners," in *Proceedings of the International Conference on Video and Image Processing*, 2017, pp. 15–19.

[26] J. Sklansky, "Finding the convex hull of a simple polygon," *Pattern Recognition Letters*, vol. 1, no. 2, pp. 79–83, 1982.