Layout Analysis of Document Images

Nina S. T. Hirata
Instituto de Matemática e Estatística – Universidade de São Paulo
nina@ime.usp.br

Abstract

In order to extract information of interest from document images, their content must be recognized. To that end, layout analysis is an important step. In layout analysis one is concerned in finding page components such as text blocks, tables, formulas, diagrams, and determining their logical role. That could enable building of efficient representation of document content, with application possibilities such as determining the right reading order, establishing relationships between page components, or improving indexing in general. In this work, we revisit the layout analysis problem reviewing how it is being impacted by advances in the field of machine learning. Then, we conclude pointing some future works.

1. Introduction

The ever-growing use of digital technologies, accompanied with advances in data collection, transmission, processing, and storage, plus ubiquitous and continuous connectivity have promoted a fast growth of the amount of digital documents. Besides the so called born-digital documents[1], many paper documents are being digitized by scanning or photographing them. Thus, in some sense information has never been so widely available. However, many digital documents are stored as images. To ease searching of contents of interest, often the documents are associated to metadata, in a similar way of library catalog items. However, finding the information of interest still requires skimming or reading the selected material. Thus, although widely available, the information of interest in general is not readily accessible.

Thus, there is a growing interest in automating information extraction from document images [8]. This would enable indexing of documents in a more meaningful way, narrowing the amount of materials to be examined after a search query, sorting out documents in more efficient ways in offices, among other applications.

[1] Wikipedia: Born-digital

Document pages usually contain multiple types of components such as text blocks, figures, tables, mathematical expressions, among others. Detecting and recognizing them is usually the first step in document analysis, a problem known as page segmentation [16]. Figure 1 shows some typical page components in scientific papers. The arrangement of these components define what is usually called the physical or geometric layout. Distinct layouts can be designed to highlight important parts and establish a logical flow, aiming clear organization of the content. Layouts may also have aesthetic purposes.

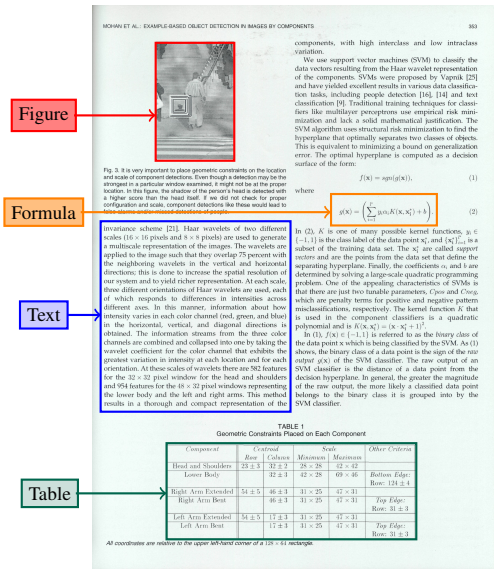

Figure 1. Examples of page components which are targets of the page segmentation task (image source: [2]; annotation by the author)

Layout analysis also involves identifying the logical role of each component. For instance, text blocks may correspond to a title, a paragraph of the body text, an entry in a table cell, or a footnote. Thus, logical layout analysis is an important step in document understanding [7]. Another important step is the representation of the overall content in an expressive and flexible manner, in such a way as

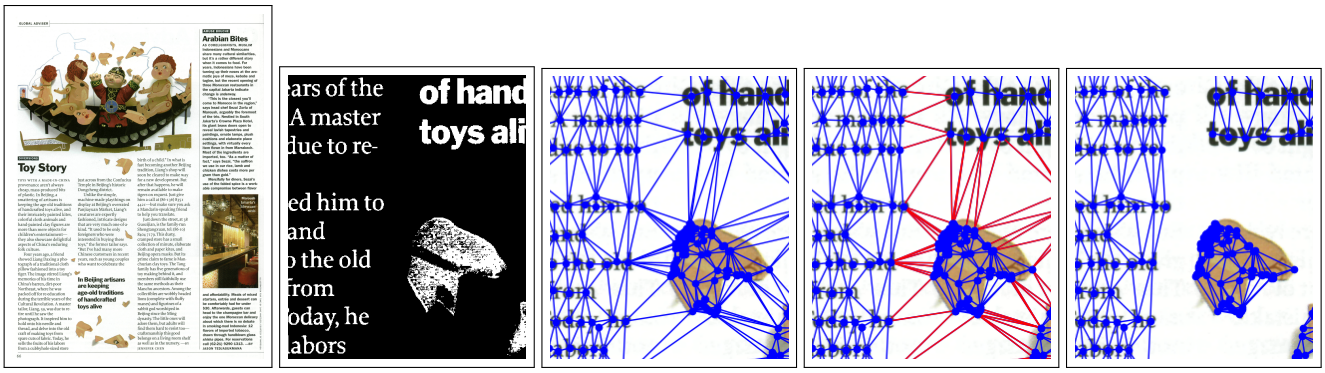

Figure 2. From left to right: full page, binarized region, nodes (connected components) and edges of the region adjacency graph, edges to be removed (in red), resulting subgraphs (page components).

to enable determining the right reading order, understanding the structure of the text (such as sections and subsections), relating figures or tables with the text block that references them, facilitating localization of specific information, among other uses.

In this work we revisit the problem of document image layout analysis, focusing on bottom-up approaches for page segmentation. This is revisited in Section 2, and then we turn our attention in Section 3 to some machine learning (ML) based methods employed within the framework of bottom-up approaches. In Section 4, we discuss what changes are happening after the emergence of deep learning techniques. In particular, we believe the boundary between physical and logical layout analysis is disappearing. From this view, in Section 5 we list potential future works toward page content representation learning.

2. Heuristic-based page segmentation

Document image processing has a long history. The fact that documents in general have a white background and dark foreground make them naturally close to binary images. Thus, it is not surprising that in early days of computation, document images were among the most processed type of images. OCR (optical character recognition) is, perhaps, one of the first widely successful applications in the field [13].

Early approaches for document image processing and analysis heavily relied on heuristics. Document binarization, noise filtering, character recognition, or skew estimation are commonly tackled problems [25]. Regarding page segmentation, existing approaches are categorized into top-down, bottom-up or hybrid approaches [3]. Bottom-up approaches refer to those that start with a super-segmentation of the image and then cluster the low-level granular segments into larger regions corresponding to the page components of interest.

In general terms, bottom-up approaches can be abstracted into the following main steps: (i) Partitioning the image into primitives (supersegmentation); (ii) Description of the primitives (e.g., by means of a set of features); (iii) Definition of adjacency relationships between primitives; (iv) Definition of a similarity measure between adjacent primitives; (v) Grouping of adjacent primitives based on similarity; (vi) Labeling of the resulting regions (i.e., assigning page component labels). Those that are familiarized with clustering-based or region-growing approaches for semantic image segmentation will easily notice the similarities. Perhaps one difference is in the appearance of image content. While semantic segmentation usually deals with natural images of objects or scenes, documents are man-made entities. Components in document images in general present sharp boundaries, while those in natural scenes do not.

An example of page segmentation process that follows the bottom-up approach is shown in Figure 2. The process considers as primitive the connected components. Connected components are computed after document binarization. This can be seen in the second image in the figure. Then, Delaunay triangulation [10] is applied to build an adjacency graph, as illustrated in the third image. In this graph, connected components correspond to the nodes and an edge indicates that the two connected components it is linking are adjacent each other. A similarity measure between two connected components can be computed based on their set of features. Assuming that closely located connected components with similar features are part of a same page component, the idea is to remove edges that are linking non-similar connected components. This is illustrated as red edges in the fourth image in the figure. After removing the red edges, those that remain define subgraphs that correspond to the page components.

The processing pipeline described above involve many steps that require parameter adjustment. For instance, binarization may require adjustments or pre-processing depend-

ing on image characteristics; for connected components one can assign color, texture, shape or other type of features, while for edges one can assign length, angle or other topological information. Edge classification can be then based on these vertex and edge features. Besides deciding which feature to compute and how similarity between two neighbor components is measured, at the end we also need to decide which edges should be kept and which one should be removed. For instance, typical spacing between characters or between lines or paragraphs can be used to decide if connected components are part of a same word or paragraph.

When a processing pipeline depends on specificities of each image, optimal parameter values need to be defined for each image or families of images. Manual tuning of parameters do not scale for large amount and large variety of document images. Thus, it is only natural that methods to automate such processing started to be developed.

3. Machine learning based methods

Supervised machine learning (ML) algorithms are powerful techniques for prediction. For instance, in the bottom-up processing example described above, one may use machine learning algorithms to decide whether an edge should be kept or removed. To that end, we need to provide training data, which consists of examples of edges to be kept and to be removed. Given sufficient amount of training data, ML algorithms in general reach good performance in a variety of application contexts. It should be noted that training data must resemble characteristics of the images to be processed later. A strong advantage of ML-based methods is the possibility of easily adapting it to documents with distinct characteristics. All is there to be done is to prepare training data and retrain the algorithm.

Many machine learning algorithms require objects to be represented by means of feature vectors. For instance, if one is interested in recognizing characters, each character image must be represented by a set of features (e.g., geometric and shape features such as normalized density, aspect ratio, curvature information, existence of extremities, among others [29]). The main idea of features is that they should be highly discriminative, so that features of objects in distinct classes are far from each other, while those in the same class are close each other. Machine learning algorithms, after properly trained, find the frontiers in the feature space that optimally separate distinct class objects.

Thus for a long period (around between 1990 and 2010) feature extraction and selection methods [12, 14], as well as metrics that better capture similarities between feature representation of the objects, received considerable attention.

Regarding document images, text/non-text segmentation of document pages was important to enable OCR systems to focus only on regions with text [5]. Besides text detection, methods for recognizing other types of page components such as tables [28], diagrams, formulas [1], photos, among others, started to emerge as computational power increased and more data became available [6].

For the bottom-up pipeline described in the previous section, machine learning algorithms can be employed in multiple steps such as in image binarization, pre-classification of connected components regarding the page component to which they belong, classification of edges so as to decide which one should be kept and which should be removed, and labeling of predicted page components [23].

In [15, 23], connected components are classified using a convolutional neural network (CNN). Instead of using hand-engineered features to describe connected components, a small patch of the original image centered on a connected component is cropped and sent to a CNN. The CNN is trained to predict the page component label corresponding to each connected component. Figure 3 shows examples of image patches corresponding to distinct connected components. Note that since connected components may have distinct scales, they are re-scaled to fit a 8×8 square and a surrounding area that, when equally re-scaled, results in 40×40 image is cropped.

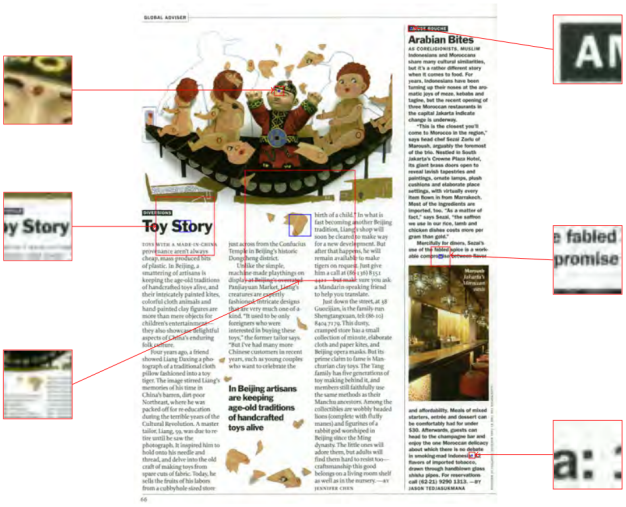

Figure 3. Examples of patches corresponding to distinct connected components in the page image. All patches are re-scaled to be of size 40×40 such that the connected component is confined into a 8×8 square region in the center.

For edge classification, a standard neural network that uses edge features (length and angle with respect to the x-axis) plus features (geometric features, positional coordinates, and CNN classification scores) of the two connected components in the extremity of the edge is used [23]. After edges are removed, the resulting connected subgraphs are assumed to be a page component. To determine the page component class, the CNN classification scores of all connected components that belongs to the page component are

taken into account to train a weighted combination.

An advantage of ML based steps in the pipeline is the possibility of retraining it for each family of documents, as well as improving each of the steps when more data is available. In fact, when data is available, machine learning makes adaptation to new scenarios simpler.

4. Deep learning based methods

Nowadays, the approaches used in the Computer Vision are mostly based on deep learning techniques [11]. Needless to say, this is also happening in the document analysis field. One important characteristic of deep learning techniques, and here we refer to deep neural networks, is their ability to learn feature extraction from raw data. This is a turning point in machine learning based methods. Now, most of the effort for designing image analysis methods are devoted to improving network models, generating training data, and optimizing training strategies and use of available data.

In the context of document processing, in fact a CNN has been employed for recognizing zip code numerals from images many years ago [18]. However, CNNs became popular only after AlexNet model [17] won the ILSVRC[2] (*ImageNet Large Scale Visual Recognition Challenge*) in 2012.

In layout analysis, a review paper published in 2017 [9] does not mention use of deep learning techniques. On the other hand, we find various papers being published in recent years [3, 4, 19, 21, 22, 31], indicating that the document image processing community has adopted deep learning techniques.

Among works that employ deep learning techniques for layout analysis, many tackle page segmentation. For instance, works [19] and [31] employ semantic segmentation networks (U-Net [27] or similar ones) for dense pixel-level classification. Authors of works [21] and [4] employ object detection networks assuming the page components to be detected are delimited by rectangles. In [24], a U-Net is also employed for dense pixel classification, but with a modified loss function that includes a regression term regarding the bounding box of page components. Some recent works [21, 22] combine visual and textual features. While visual features are extracted with CNNs, textual features are extracted with language models such as BERT. In this type of approach, different strategies for combining the multimodal features can be explored.

In summary, page component segmentation is now performed in an end-to-end fashion. In general, some post-processing is required to fix imperfections on the boundaries or to remove overlap between two detected components. To enable training of large networks, new and large annotated datasets are being made publicly available [20, 26, 30], which will possibly push new developments. We note, however, that new datasets mostly consist of documents with rectangular or Manhattan layouts, those where the page components can be delimited by bounding boxes. We also observe a larger variety of page components, including logical categories, being considered in page segmentation. We understand this as an indication that the separation between physical and logic layout analysis is becoming less clear, and both will be fused eventually.

[2] ILSVRC

5. Concluding remarks

Machine learning based techniques can successfully segment text regions and are being employed to segment various categories of page components such as tables, formulas, diagrams, pictures, among others. In particular, deep learning techniques enabled end-to-end processing for this type of tasks, and combination of visual and textual features is enabling fine grained logical level classification of page components (for instance, discriminating text blocks as titles, footnote, paragraph, etc). As future work, there is room for the development of methods for the segmentation of non-rectangular page components as well as improving boundary precision of detected page components. Moreover, with more recent architectures such as transformers, there will be opportunities for developing new multimodal feature fusion methods to refine classification of page components regarding their logical categories. In addition, graph neural networks seems to be a natural approach to capture spatial and hierarchical relationships between page components. This, allied with multi-modal and attention mechanisms, have potential to establish semantic connections between page components (for instance, a figure and a paragraph that describes it). All these ideas are important to advance toward learning expressive and flexible page content representation schemes in a data-driven fashion.

Acknowledgements: This work was supported in part by the FAPESP (The São Paulo Research Foundation), grants 2017/25835-9 and 2015/22308-2.

References

[1] D. Anitei, J. A. Sánchez, J. M. Fuentes, R. Paredes, and J. M. Benedí. ICDAR 2021 Competition on Mathematical Formula Detection. In *Document Analysis and Recognition – ICDAR 2021: 16th International Conference, Proceedings, Part IV*, pages 783–795. Springer-Verlag, 2021. 3

[2] A. Antonacopoulos, D. Bridson, C. Papadopoulos, and S. Pletschacher. A realistic dataset for performance evaluation of document layout analysis. In *10th International Conference on Document Analysis and Recognition*, pages 296–300. IEEE, 2009. 1

[3] G. M. Binmakhashen and S. A. Mahmoud. Document layout analysis: A comprehensive survey. *ACM Comput. Surv.*, 52(6), 2019. 2, 4

[4] S. Biswas, P. Riba, J. Lladós, and U. Pal. Beyond document object detection: Instance-level segmentation of complex layouts. *Int. J. Doc. Anal. Recognit.*, 24(3):269–281, 2021. 4

[5] S. S. Bukhari, M. I. A. Al Azawi, F. Shafait, and T. M. Breuel. Document image segmentation using discriminative learning over connected components. In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, pages 183–190. ACM, 2010. 3

[6] C. Clark and S. K. Divvala. Looking beyond text: Extracting figures, tables and captions from computer science papers. In *AAAI Workshop: Scholarly Big Data*, 2015. 3

[7] A. Dengel and F. Shafait. *Analysis of the Logical Layout of Documents*, pages 177–222. Springer London, 2014. 1

[8] David S. Doermann and Karl Tombre, editors. *Handbook of Document Image Processing and Recognition*. Springer, 2014. 1

[9] S. Eskenazi, P. Gomez-Krämer, and J.-M. Ogier. A comprehensive survey of mostly textual document segmentation algorithms since 2008. *Pattern Recognition*, 64:1–14, 2017. 4

[10] S. Fortune. *Voronoi Diagrams and Delaunay Triangulations*, page 377–388. CRC Press, Inc., USA, 1997. 2

[11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. 4

[12] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, 2003. 3

[13] S. Impedovo, L. Ottaviano, and S. Occhinegro. Optical character recognition – a survey. *International Journal of Pattern Recognition and Artificial Intelligence*, 5(01n02):1–24, 1991. 2

[14] A. Jain and D. Zongker. Feature Selection: Evaluation, Application, and Small Sample Performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):153–158, 1997. 3

[15] F. D. Julca-Aguilar, A. L. L. M. Maia, and N. S. T. Hirata. Text/non-text classification of connected components in document images. In *30th Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 450–455. IEEE, 2017. 3

[16] K. Kise. Page segmentation techniques in document analysis. In *Handbook of Document Image Processing and Recognition*, pages 135–175. Springer, 2014. 1

[17] Aa Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 4

[18] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989. 4

[19] J. Lee, H. Hayashi, W. Ohyama, and S. Uchida. Page segmentation using a convolutional neural network with trainable co-occurrence features. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 1023–1028, 2019. 4

[20] M. Li, Y. Xu, L. Cui, S. Huang, F. Wei, Z. Li, and M. Zhou. DocBank: A Benchmark Dataset for Document Layout Analysis, 2020. 4

[21] S. Li, X. Ma, S. Pan, J. Hu, L. Shi, and Q. Wang. VTLayout: Fusion of Visual and Text Features for Document Layout Analysis. In *Trends in Artificial Intelligence: 18th Pacific Rim International Conference on Artificial Intelligence (PRICAI)*, page 308–322, 2021. 4

[22] S. Luo, Y. Ding, S. Long, J. Poon, and S. C. Han. Doc-GCN: Heterogeneous Graph Convolutional Networks for Document Layout Analysis. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING*, pages 2906–2916. International Committee on Computational Linguistics, 2022. 4

[23] A. L. L. M. Maia, F. D. Julca-Aguilar, and N. S. T. Hirata. A machine learning approach for graph-based page segmentation. In *31st Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 424–431. IEEE, 2018. 3

[24] L. Markewich, H. Zhang, Y. Xing, N. Lambert-Shirzad, Z. Jiang, R. K.-W. Lee, Z. Li, and S.-B. Ko. Segmentation for document layout analysis: Not dead yet. *Int. J. Doc. Anal. Recognit.*, 25(2):67–77, 2022. 4

[25] L. O'Gorman and R. Kasturi, editors. *Document Image Analysis*. IEEE Computer Society Press, Washington, DC, USA, 1995. 2

[26] B. Pfitzmann, C. Auer, M. Dolfi, A. S. Nassar, and P. Staar. DocLayNet: A Large Human-Annotated Dataset for Document-Layout Segmentation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, page 3743–3751. Association for Computing Machinery, 2022. 4

[27] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015. 4

[28] Faisal Shafait and Ray Smith. Table detection in heterogeneous documents. In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, pages 65–72. Association for Computing Machinery, 2010. 3

[29] O. D. Trier, A. K. Jain, and T. Taxt. Feature extraction methods for character recognition – a survey. *Pattern Recognition*, 29(4):641–662, 1996. 3

[30] X. Zhong, J. Tang, and A. Jimeno-Yepes. PubLayNet: largest dataset ever for document layout analysis. *CoRR*, abs/1908.07836, 2019. 4

[31] Y. Zou and J. Ma. Deep learning based semantic page segmentation of document images in chinese and english. In *Intelligent Computing Theories and Application: 17th International Conference*, pages 484–498. Springer-Verlag, 2021. 4