AN OVERVIEW OF DOCUMENT IMAGE ANALYSIS SYSTEMS

Andrei Tigora[1]

ABSTRACT

This paper presents an overview of Document Image Analysis Systems, their composing modules, the approaches these modules use, as well as uses for these applications. One of the main goals is to present some of the most important technologies and methods behind the Document Image Analysis domain in order to evaluate the best approach when dealing with real-world documents. The other main goal is to ensure a foundation for those starting to build such complex software systems and to give an elaborate technical answer to the question: "How to make physical documents available to a large number of people?"

Keywords: Document image analysis, character recognition, OCR, image data extraction, image export

1. INTRODUCTION

Scanning physical pages and storing them in a digital format is a means of making physical data available to the digital world. It also solves the problems of storage, paper deterioration, accessibility and many others. However, what it does not do is make it faster to pinpoint the data that is relevant for a certain endeavor; giving the pixels structure and extracting the underlying information into a computer comprehensible representation is the job of Document Image Analysis (DIA) systems. When talking about DIA systems, the first thing that comes to mind is Optical Character Recognition [2][16], both because it has been in use for quite some time in the American postal system, and because it is something users have come to expect when interacting even with non-text files. However, OCR is no more than a small component of much larger applications, and represents a narrow view of what a DIA really is.

Document image analysis is generally aimed at synthetic images [1], images that contain symbolic objects, such as book pages, postal addresses on letters, engineering drawings, sheet music, maps and so on. However, it can also deal with naturally occurring patterns as well, such as fingerprint analysis [16]. The other category of images is represented by "natural" images, such as photographs, satellite images, X-rays - objects that may be captured with standard or non-standard cameras. A picture of a Chinese traffic sign may be part of both categories, depending on what is expected of it, but this should not represent a problem for the rest of the discussion.

2. PROCESSING STEPS

Firstly it should be noted that there is no such thing as a uniquely recognized taxonomy of the processing steps images undergo from the raw form to the computer comprehensible data. Nagy [1] proposed a taxonomy that is comprehensive and it catalogues procedures

[1] Engineer, Jinny Software Romania SRL 13 C Pictor Ion Negulici, Bucharest, Romania, andrei.tigora@jinnysoftware.com

based on the granularity of the entities they deal with. The five identified levels, starting from the lowest granularity, are as follows:

- Pixel level
- Primitive level
- Structure level
- Document level
- Corpus level

Nagy's taxonomy also differentiates between the nature of the input images, separating those whose contents is mostly text, from those that are made of mostly non-text graphics. A similar differentiation between character and graphics dominated images is also reflected in [2]. However, for some input images, depending on the desired output, processing steps from both categories might be necessary. For example, a scanned image of a circuit diagram will need both OCR processing and specialized line detection and a circuit identification component.

The classification used throughout this paper is loosely based on Nagy's taxonomy [1]. The next sections will present an overview of each level, with a focus on text documents.

3. PIXEL LEVEL PROCESSING

Pixel level processing deals with image to image, attempting to transform the given images into more appropriate versions for the following levels. Algorithms that fall within this category handle noise reduction, binarization, character segmentation, character scaling and vectorization.

Noise Reduction

Noise in images has multiple causes, such as degraded input images, imperfect capture devices, as well as improper use of the previously used devices, compression and transmission error. Before using the images, the optical abnormalities have to be compensated for.

Noise reduction aims at increasing the signal to noise ratio and it is done not only for images, but for all forms of signals. In the world of digital processing it plays an important role not only in digital image analysis, but also in medicine [20] and astronomy [23]. Noise reduction can be applied on bitonal [16], grayscale [21][22], as well as directly on color images [19][20]. Most noise reduction mechanisms make assumptions concerning the noise distribution pattern [18]. Gaussian distribution is most widely assumed, due to its simplicity [19][5], but there Poisson distribution [20] is also considered, as well as non-Gaussian distributions [5].

The approaches used for noise reduction nowadays are quite varied [5] and a lot more complex than those that were in use ten years ago [2]. The classic approach is that of morphological methods, which rely on sequences of erosion and dilation transformations combined with segmentation heuristics [63]. These are non-linear filters, and together with linear filters, such as those based on Gauss filters [17], fall into the spatial domain filtering

category. More recently, special attention has been given to transform domain filtering; among these, the best results appear to be produced by wavelet domain filters [21]. Although they perform better than other methods for natural images, they do not seem as successful with synthetic ones, which represent the focus of document image analysis systems.

Binarization

Just like noise reduction, binarization [64] is relevant for multiple types of images, not only for document images. Through the process of binarization, the pixels of an image are assigned to two categories: foreground and background. This is perhaps the most radical transformation an image undergoes, due to the significant information loss; yet, for a text image, this simplification should not make the image lose any of its value, as characters will become foreground and the rest of the image background.

The most used algorithms are thresholding based [24][25][26]; thresholding methods compute a global or local value (threshold) [64][65] which is used a delimiter between black and white pixels. Binarization and noise reduction can be combined together in a single step in order to classify better what is foreground and what is background. [62][63] A different approach is used by dithering algorithms: halftone [27] and ordered [28]. These ones attempt to reproduce the color densities within a given boundary using only a predefined set of color values. The last type of binarization algorithms is error dispersion technique [29]; they aim to minimize the error representations across the image, through approximation and propagation of the color difference between neighboring pixels.

Deskew

Skew detection rather than deskewing itself is the main focus of the various algorithms used for skew correction. Image skew is the result of improper handling of documents during digitization process; depending on the device, and the nature of the document, improper placement may lead to the appearance of dark regions around the borders of the document, known as marginal noise [8]. Although deskewing does not concern itself with removing marginal noise, these two enhancements are usually performed one after the other. A correctly aligned document simplifies the character and text line detection algorithms [7], assuming that text itself has the expected orderly distribution.

Some approaches for skew estimation rely on projection profiles or Radon transform. These algorithms rotate the image within certain limits and evaluate the obtained projected profiles; the one with the largest variation corresponds to the skew angle [5][66]. Just like the projection profiles, Hough transform methods execute a sweep of possible angle values in the attempt to determine the image skew; they rely on the observation that most collinear pixels will be encountered along lines that a parallel to a text line's base line [4][7][31]. Filtering in the Hough space can improve results [69] but unfortunately neither of these methods produces good results when dealing with documents that also contain images, as they have different pixel distributions that affect the statistics.

Unlike the previous methods, nearest neighbor approach first determines character blocks and then, by grouping the ones that are closest. This is done under the assumption that they

must be part of the same text line. Once the neighbor groups are determined, the skew angle is computed by analyzing the variations between neighboring entities [5][30][67]. These methods also have a downside: they are very sensitive to noise, which can introduce artificial entities that disturb the neighborhood creation process. There are however methods which give a confidence factor for the detected rotation and can even correct bent pages caused by the shape of the document or lens distortions [68].

Character Segmentation

Character segmentation deals with identifying the particular regions in an image that correspond to a single character. Segmenting Latin script-based printed texts in high quality images poses little problems, as there are few - at most 3 but usually only one - connected components that compose a character. This is usually solved by determining the connected components and running a nearest neighbor algorithm to assign the various diacritics to the corresponding central entity.

However, with low quality images, characters tend to either merge or be split into multiple entities, with some fonts being more prone to a certain behavior than others. Such situations are common for other scripts, such as Arabic, where the characters are linked, or Chinese [56], where characters may be composed of several distinct components or scripts that few people know how to interpret, like Lanna which has touching and overlapping letters. [14] Therefore, some of the mechanism initially designed to solve the problem of low quality images are actually employed for character segmentation for those specific scripts.

Separating merged characters may be done by explicit character identification [32][33][35]. Yet this limits the segmentation to a particular script, making it less generic that desired. Another solution relies on vertical projection [33][34], the aim being to determine the coordinates with fewest pixel and split along that coordinate. Thinning [33][34] may also be employed, as it may result in the "natural" elimination of the merge points, but it can just as well overly segment the existing characters.

Merging separated entities into a single character usually comes down to evaluating the distances between the various components. The components may be represented by their bounding boxes or the pixels themselves or by Voronoi diagrams, Delaunay triangulation or 3D meshes. [70][71][72]

4. PRIMITIVE LEVEL PROCESSING

The aim of primitive level processing is to identify the nature of the elements that have become accessible following the preprocessing steps. Whereas for text documents the focus is the eventual recognition of characters, for non-textual documents the aim is recognizing basic geometric entities.[13] For scanned maps, the deduction of textual elements becomes harder, as the text has various orientations and is placed over variable colored background. Morphological operations together with statistical data help in identifying interesting regions. [15]

Character Recognition

The early work in character recognition was performed on Latin script, as it is the dominant script of the Western world. Though not yet perfect, in ideal circumstances, character recognition rates are close to 100% [36]. All methods rely to some extent on matching a candidate character against a set of features. The most basic approach is using a reference bitmap of the character, which is then compared to a scaled version of the candidate. More reliable reference sets are made of line strokes, stroke crosses, relative angles [1][37]. The purpose is finding some features that can uniquely and unambiguously classify a certain candidate object as belonging to exactly one group. The reference set of features is determined through machine learning, usually a neural network.

Character classification was traditionally performed on a per character basis, determining which features best fit the proposed character. The fitness function is usually relatively simple, computing the candidate's deviation from the reference pattern in terms as distance from the reference's list of features. However, experience over the years has shown that this approach has a high degree of ambiguity, which can only be solved by using higher level information, such as language patterns. By taking into account linguistic information, the classification can become more precise, with the use of Generalized Hidden Markov Models or Bayesian Networks, thus indicating what sequence of characters is more likely to occur.

To some extent it can be said that the field has somehow matured, with developers and researchers alike acknowledging the performances of current OCR products [11][36][41].

Most of the current research in character recognition is targeted towards Arabic and Chinese scripts, as well as handwritten documents [2]. Handwritten and Arabic script based texts are similar in that for both cases characters that form a word are linked to one another. Also, while the shape of Arabic characters varies based on context, the shape of characters for handwritten texts tends to display larger variation not only between documents written by different people, but also within the same document.

For Chinese based scripts, and others inspired by it, the main difficulty is represented by the large number of available characters, which makes them unsuitable for evaluation using approaches employed for Latin based scripts. Yet, there are some mechanisms that respond well to the demands imposed by this script, such as Tesseract [37]. Overall though, most algorithms developed for analyzing Latin script can be applied, with minor modifications/customizations for any type of script [39][40].

5. STRUCTURE LEVEL PROCESSING

For this level of processing, the aim is to give "meaning" to the groups of entities, such as identifying words, reconstructing text lines and interpreting tables.[10]

The problem of word segmentation must once again receive different treatment depending on the used script. Modern Latin based scripts use space as a means of separating words

and in some special cases other symbols, such as hyphens. Original Latin monumental inscription though, did not separate words; the same applies for Chinese language.

For Latin script based texts, separating words is a matter of evaluating distances between consecutive characters. Yet, for noisy document images, the distances between characters will be affected by the presence of unexpected dark pixels, so segmenting words requires some sort of linguistic model information. This is also holds true for Chinese texts that do not use any kind of separator, and relies on the readers' knowledge to decide where to split the words.

Using linguistic information means that segmentation becomes an OCR type processing. The oldest approach dates from the 1960s, when N-grams were proposed for error correction [1] and later extended to word segmentation [55], as an alternative to the high requirements of using a full dictionary [57]. Both the N-grams and dictionaries may be obtained from preexisting repositories [55], or generated from "relevant" segmented sources [57]. The actual segmentation evaluates candidate sequences, choosing the one that has the highest probability, and in the case of varying length patterns slightly favoring longer sequences, as they are less frequent. Linguistic information can be generated on the spot from the scanned document and then used in the segmentation for the further documents, or correction in post-processing. [74]

6. DOCUMENT LEVEL PROCESSING

Modern document level processing is centered on document layout analysis, which is grouping elements into logical sequences, and to some extent, identifying outstanding elements of the text, such as authors, headers, etc. For layout analysis it is assumed that entities such as words (or even text lines), tables and images have already been identified, along with their bounding boxes. Information concerning the nature of the contents of the bounding boxes is not compulsory. Grouping the different elements is a matter of observing distance conventions that are usually respected while writing texts, finding interpreting white spaces, lines [75] and any form separators [76]. Methods for detecting font characteristics accurately give valuable information [70]. Authors evaluate these distances relative to text size [77], as the blank columns separating two text columns need to be large enough to unambiguously split two lines that have the same vertical coordinates [42]. For top-down approaches, so called X-Y cut algorithms are employed, based on X-Y trees, that recursively split/cut regions in the image by one of the two axes, to eventually end up with a [45]

As previously stated, layout analysis does not limit itself to grouping elements; it also performs text labeling. This can be achieved both through bottom-up and top-down approaches. The labeling can rely exclusively on block page distribution [44], or be enhanced with OCR information [45].

Other important issues are those of detecting tables, graphs, pictures and text regions within a document image. The algorithms should be able to cut out each element from the image without including parts from other objects, even if this means cutting in an irregular manner [78][71].

One approach for detecting table regions is to identify the graphic elements that compose the table: line segments and intersections [58]. This tends to be relatively resilient to noise, but it is cannot be applied to tables that do not have graphic components. Other solutions attempt to analyze word distributions within a certain region and make decisions based on their positions relative to other words.

In [59] authors attempt to identify tables based on row line sparsity, which is how large the white spaces on a particular line are. However, this assumes that the table cells have little content, which may not always be the case. An alignment based approach is proposed in [60], first identifying narrow text columns and then grouping them together into tables. A hybrid implementation is described in [61]; this attempts combining non-tabular information such as header and trailer position, white spaces/lines separating table cells as well as background color variations.

7. CORPUS LEVEL PROCESSING

Once a document image has been completely analyzed and enhanced with information, the interactions that are possible for an application specific digital version of the scanned document should become available to the user. These include content indexing and search, as well as validation and document update. When cataloging papers, humans first react to names, such as the title of the publication, so the expectation was that machines would do the same thing. Unfortunately, the documents that really require automating indexing are usually old and low quality, which leads to poor performances of OCR [36]. Therefore, other approaches have been developed such as those based on document layout. The algorithm is first trained on a set of images to learn the existing layouts, then for the classification phase, for each new image the layout is determined [71] and a fitness score is computed based on how well the polygons in the image fit the ones on the reference [43][46]. This layout recognition approach can also be used as a retrieval mechanism, which can prove to be a more robust mechanism than OCR.

However, information extraction mechanism can rely on elements as low level as pixels. The authors of [47] describe an image identification algorithm that uses pixel distributions measurements to identify particular images. A slightly higher level approach analyzes character shapes, more precisely the vertical segments of characters, to construct image feature sets, than are then used for identifying a particular document [9].

8. PERFORMANCE EVALUATION

Evaluating the performance of a particular algorithm should not be a matter of debate, yet, in the field of Document Image Analysis it often is. The first problem arises from the modular, hierarchical structure of Document Analysis Systems. In this context, a binarization algorithm cannot be evaluated simply by comparing its output to the ground truth image. A image sharpness indicator is a useful starting point [73]; a more reliable indicator would be the correct identification of characters and other features by the higher level analyzers. Even so, an advanced OCR analyzer might in fact hide an average performance from the lower levels, so a comparison should be performed using alternative modules.

Choosing the reference images is also an important component. While there is a general consensus that using a high number of files is desirable [1], the community is split when it comes to what are those files that should be used. This problem exists from binarization, all the way up to the high level processing algorithms. The reference output, the so-called "ground truth", is the great issue. For example, when binarizing artistic images and photographs it is highly unlikely to have a uniquely satisfying binary image, whereas for synthetic images, the binary reference is usually unique - unless the noise in the image is for some reason more relevant that the actual contents. Despite its unique nature, another problem arises, that of generating the ground truth information. On one hand, one may start from the ground truth and distort it in various ways so that the input that is processed is the distorted version of the original ground truth. The alternative would be to generate the ground truth from documents that are usually processed by program or platform. As expected, both approaches have their drawbacks; the first one is relatively fast and allows the designers great control over the kind of distortions they expect to test, but the distorted images tend to be an "unnatural". Therefore, algorithms that are evaluated against such inputs are being fine tuned for situations that are never encountered in real documents. The latter solution on the other hand, while it does provide realistic input, there is no rapid means of quickly generating the ground truth - the ground truth must be generated by hand by humans, making the approach non-scalable for current testing demands [38].

9. ANALYSIS SYSTEM APPROACHES

Ideally, a single software package should be able to extract the information from any type of document it is presented, but the truth is that most times highly specialized software is used to solve a specific problem. For optimum results, analysis systems, as well as their components, tend to have a limited applicability. This may involve a limited series of processing than can be performed, imposing a restriction on the quality of the documents, limited language support, meaning that only documents written in a particular language may be processed, or specific document layout, for archives of documents whose format repeats over and over again [12][48]. Yet, even when the input material is limited from the point of view of the contents, the required processing might be very complex, due to high variety of other parameters such as paper color, non-standard text layout, and wide ranging handwriting styles as is the case of [53].

Document image analysis systems may be limited in scope to things such as extracting names from documents [4], or processing only tables, graphs and images [3]. This is simplifications rely on the assumption that only certain information is worth being used for document indexing.

The system described in [6] circumvents character recognition altogether, extracting "word images" that are then matched with user inputted keywords through word-to-word matching.

Some systems simply cannot compensate for the low quality of the documents that are processed, so the designers developed systems that rely on user feedback. As stated in [36], without human intervention, the results tend to degrade significantly, so human intervention is compulsory for good results. The systems proposed in [7][12] uses initial human input to create templates that are afterwards used to extract information for similarly constructed

documents, with no other user intervention expected. More common though, is allowing human users to review the proposed results of the analysis and correct them accordingly [12][51].

Other systems allow the user to gain full control of their capabilities [49], which could be useful whenever the batch mode processing does not yield the expected results. A more radical approach is to leave the entire document analysis to the end user [50], taking advantage of a large community of scientists that are interested in those documents and can collaborate on enhancing them. Some degree of automated processing could be integrated nonetheless, having been tried in [51]. However, in order to eliminate the effort of duplication, a single shared model of the documents should be used [52]. Crowd sourcing is also used for [54], in order to review the results of the automated processing and improve the quality of the end documents.

10. CONCLUSIONS

There is no universal solution to all problems, due to high variety of input material.

However, finding optimal solutions for specific document input categories and a specific (pre)processing stage is possible. In order to take advantage of a "best" solution in a "most suitable" processing environment/phase an intelligent Document Image Analysis System is needed, which would allow choosing both the algorithm/method to solve a specific problem and the parameter settings most suitable for the task.

11. REFERENCES

1. G. Nagy, "20 Years of Document Image Analysis in PAMI", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 1, January 2000.
2. R.r Kasturi, L. O'gorman, V. Govindaraju, "Document image analysis: A primer", *Sadhana*, Vol. 27, No. 1, February 2002.
3. X. Lu , S. Kataria , W. J. Brouwer , J. Z. Wang , P. Mitra , C. Lee Giles, "Automated analysis of images in documents for intelligent document search", *International Journal on Document Analysis and Recognition*, Vol. 12. No. 2, June 2009.
4. L. Likforman-Sulem, P. Vaillant, A. B. de la Jacopière, "Automatic name extraction from degraded document images", *Pattern Analysis and Applications*, Vol. 9, No. 2-3, October 2006.
5. T. Saba, G. Sulong, A. Rehman, "Document image analysis: issues, comparison of methods and remaining problems", *Artificial Intelligent Review*, Vol. 35, No. 2, February 2011.
6. C. B. Jeong, S.H. Kim , "A Document Image Preprocessing System for Keyword Spotting", *Proceedings of the 7th international Conference on Digital Libraries: international collaboration and cross-fertilization*, December 2004.
7. C. Antonio Peanho, H. Stagni, F. S. C. da Silva, "Semantic information extraction from images of complex documents", *Applied Intelligence*, Vol. 37, No. 4, December 2012.
8. Z. Hu, X. Lin, H. Yan, "Document image retrieval based on multi-density features", *Frontiers of Electrical and Electronic Engineering in China*, Vol. 2, No. 2, 2007.
9. C. L. Tan, W. Huang, S. Y. Sung, Z. Yu, Y. Xu, "Text Retrieval from Document Images Based on Word Shape Analysis", *Applied Intelligence*, Vol. 18, No. 3, May-June 2003.

10. J.-Y. Ramel, S. Busson, M. L. Demonet, "AGORA: the interactive document image analysis tool of the BVH project", *Second International Conference on Document Image Analysis for Libraries 2006*, 27-28 April 2006.
11. A. Antonacopoulos, D. Karatzas, "Document Image Analysis for World War II Personal Records", *Proceedings of First International Workshop on Document Image Analysis for Libraries*, 2004.
12. J. He, A. C. Downton, "User-assisted archive document image analysis for digital library construction", *Proceedings of Seventh International Conference on Document Analysis and Recognition*, 3-6 August 2003.
13. E.E. Regentova, S. Latifi, D. Chen, K. Taghva, D. Yao, "Document analysis by processing JBIG-encoded images", *International Journal of Document Analysis and Recognition*, Vol. 7, No. 4, September 2005.
14. S. Pravesjit, A. Thammano, "Segmentation of Historical Lanna Handwritten Manuscripts", *Intelligent Systems (IS), 2012 6th IEEE International Conference*, 6-8 September 2012.
15. S. Biswas, A. K. Das, "Text extraction from scanned land map images", *2012 International Conference on Informatics, Electronics & Vision (ICIEV)*, 18-19 May 2012.
16. L. O. Gorman, R. Kasturi, "Document Image Analysis", *Los Alamitos: IEEE CS Press*, 1995.
17. J. Lee, R.-H. Park, S. Chang, "Noise reduction using multiscale bilateral decomposition for digital color images", *Signal, Image and Video Processing*, August 2012.
18. H.-Y. Lim, D.-S. Kang, "Efficient noise reduction in images using directional modified sigma filter", *The Journal of Supercomputing*, December 2012.
19. J. Harikiran, R. Usha Rani, "Color Image Restoration Method for Gaussian Noise Removal", *Information and Communication Technologies Communications in Computer and Information Science*, Vol. 101, 2010, pp 554-560.
20. S. Okawa, Y. Endo, Y. Hoshi, Y. Yamada, "Reduction of Poisson noise in measured time-resolved data for time-domain diffuse optical tomography", *Medical & biological engineering & computing*, 50(1), 2012, pp 69–78.
21. A. D. E. Stefano, P. R. White, W. B. Collis, "Selection of Thresholding Scheme for Image Noise Reduction on Wavelet", 2004, pp 225–233.
22. M. Nachtegael, S. Schulte, D. Weken, D. Van Der, V. De Witte, E. E. Kerre, "Do Fuzzy Techniques Offer an Added Value for Noise Reduction in Images?" *Advanced Concepts for Intelligent Vision Systems*, 2005, pp. 658–665.
23. S. Harmeling, B. Scholkopf, H. C. Burger, "Removing noise from astronomical images using a pixel-specific noise model", *IEEE International Conference on Computational Photography*, 2011, pp 1-8.
24. N. Otsu, "A threshold selection method from gray-level histograms", *IEEE Transactions on Systems, Man and Cybernetics*, 1979.
25. M. Sezgin, B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation", *Journal of Electronic Imaging*, 2004.
26. J. Zhang, J. Hu, "Image segmentation based on 2D Otsu method with histogram analysis", *Proceedings of the International Conference on Computer Science and Software Engineering. IEEE Computer Society*, Washington, DC, USA, 2008.
27. A. R. Ulichney, "Halftone Characterization in the Frequency Domain", *Imaging Science and Technology 47th Annual Conference*, 1994.
28. A. R. Ulichney, "The void-and-cluster method for dither array generation", *SPIE*, Vol. 1913, 1993.
29. R. W. Floyd, L. Steinberg, "An adaptive algorithm for spatial grey scale", *Proceedings of the Society of Information Display*, Vol. 17, pp. 75–77, 1976.

30. Y. Lu , C. L. Tan, "A nearest-neighbor chain based approach to skew estimation in document images", *Pattern Recognition Letters,* Vol. 24, pp. 2315–2323, 2003.
31. Y. M. Alginahi, "A survey on Arabic character segmentation"*, International Journal on Document Analysis and Recognition*, 2012.
32. J. Wang, J. Jean, "Segmentation of Merged Characters by Neural Networks and Shortest-Path", *Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing*, pp. 762–769, 1993.
33. Y. M. Alginahi, "A survey on Arabic character segmentation"*, International Journal on Document Analysis and Recognition*, 2012.
34. A. Zahour, B. Taconet, L. Likforman-Sulem, W. Boussellaa, "Overlapping and multi-touching text-line segmentation by Block Covering analysis", *Pattern Analysis and Applications*, 2008.
35. A. Nomura, K. Michishita, S. Uchida, M. Suzuki, "Detection and Segmentation of Touching Characters in Mathematical Expressions", *Seventh International Conference on Document Analysis and Recognition*, pp. 126–130, 2003.
36. R. Holley, "How Good Can It Get? Analysing and Improving OCR Accuracy in Large Scale Historic Newspaper Digitisation Programs"*, D-Lib Magazine*, Vol. 15, Issue 3, pp. 1–13, 2009.
37. R. Smith, "An Overview of the Tesseract OCR Engine", *Ninth International Conference on Document Analysis and Recognition*, Vol 2, pp. 629–633, 2007.
38. J. Callan, P. Kantor, D. Grossman, "Information Retrieval and OCR: From Converting Content to Grasping Meaning", *ACM SIGIR Forum*, Vol. 36, Issue 2, 58–61, 2002.
39. F. Hedayati, J. Chong, K. Keutzer, "Recognition of Tibetan Wood Block Prints with Generalized Hidden Markov and Kernelized Modified Quadratic Distance Function", *Proceedings of the 2011 Joint Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data*, 2011.
40. M. K. Jindal, M. Kumar, M., R. K. Sharma, "Offline Handwritten Gurmukhi Character Recognition: Study of Different Feature-Classifier Combination", *Proceeding of the Workshop on Document Analysis and Recognition Pages*, pp. 94–99, 2012.
41. H. Déjean, J. Meunier, "Structuring Documents According to Their Table of Contents", *ACM symposium on Document Engineering*, pp. 2–9, 2005.
42. P. E. Mitchell, H. Yan, "Newspaper Document Analysis featuring Connected Line Segmentation", *Proceedings of the Pan-Sydney area workshop on Visual information processing*, Vol. 11, pp. 77–81, 2001.
43. L. Golebiowski, "Automated Layout Recognition", *1st ACM workshop on Hardcopy document processing*, pp. 41–45, 2004.
44. B. Rosenfeld, R. Feldman, Y. Aumann, "Structural Extraction from Visual Layout of Documents", *Proceedings of Eleventh International Conference on Information and Knowledge Management*, pp. 203–210, 2002.
45. A. Takasu, K. Aihara, "Information Extraction from Scanned Documents by Stochastic Page Layout Analysis", *Proceedings of the 2008 ACM symposium on Applied computing*, pp. 447, 2008.
46. L. Lecerf, D. Maupertuis, "Scalable Indexing for Layout Based Document Retrieval and Ranking", *Proceedings of the 2010 ACM Symposium on Applied Computing*, pp. 28–32, 2010.
47. Z. Hu, X. Lin, H. Yan, "Document Image Retrieval Based on Multi-Density Features", *Frontiers of Electrical and Electronic Engineering in China*, Vol. 2, Issue 2, 2007.
48. A. B. S. Almeida, R. D. Lins, G. D. F. Pereira e Silva, Thanatos: "Automatically Retrieving Information from Death Certificates in Brazil", *Proceedings of the 2011 Workshop on Historical Document Imaging and Processing*, pp. 146–153, 2011.

49. R. D. Lins, G. D. F. Pereira e Silva, A. D. A. Formiga, “Enhancing a Platform to Process Historical Documents”, *Proceedings of the 2011 Workshop on Historical Document Imaging and Processing*, Vol. 0, pp. 169–176.
50. P. Tranouez, S. Nicolas, V. Dovgalecs, A. Burnett, L. Heutte, Y. Liang, R. Guest, R., “DocExplore: Overcoming Cultural and Physical Barriers to Access Ancient Documents”, *Proceedings of the 2012 ACM symposium on Document engineering*, pp. 205–208, 2012.
51. A. H. Toselli, E. Vidal, A. Juan, “Interactive Layout Analysis and Transcription Systems for Historic Handwritten Documents Categories and Subject Descriptors”, *Proceedings of the 10th ACM Symposium on Document Engineering*, pp. 219–222, 2010.
52. R. Sanderson, B. Albritton, R. Schwemmer, H. Van De Sompel, “SharedCanvas: A Collaborative Model for Medieval Manuscript Layout Dissemination”, *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*, pp. 175–184, 2011.
53. E. Matthaiou, E. Kavallieratou, “An information extraction system from patient historical documents”, *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pp. 787, 2012.
54. T. Ishihara, T. Itoko, D. Sato, A. Tzadok, H. Takagi, “Transforming Japanese Archives into Accessible Digital Books Categories and Subject Descriptors”, *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries*, pp. 91–100, 2012.
55. V. Zhikov, H. Takamura, “An Efficient Algorithm for Unsupervised Word Segmentation with Branching Entropy and MDL”, *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pp. 832–842, 2010.
56. A. Chen, “Chinese Word Segmentation Using Minimal Linguistic Knowledge”, *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pp. 148–151, 2003.
57. P. Simon, S. Hsieh, L. Prevot, C.-R. Huang, “Rethinking Chinese Word Segmentation: Tokenization, Character Classification, or Wordbreak Identification”, *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pp. 69–72, 2007.
58. L. A. Pereira Neves, J. M. de Carvalho, F. Bortolozzi, “A Table-form Extraction with Artefact Removal”, *Proceedings of the 2007 ACM Symposium on Applied Computing*, pp. 622–626, 2007.
59. Y. Liu, P. Mitra, C. L. Giles, “Identifying table boundaries in digital documents via sparse line detection”, *Proceeding of the 17th ACM conference on Information and knowledge mining*, pp. 1311, 2008.
60. F. Shafait, R. Smith, “Table detection in heterogeneous documents”, *Proceedings of the 8th IAPR International Workshop on Document Analysis Systems*, pp. 65–72, 2010.
61. G. Harit, P. Art, “Table Detection in Document Images Using Header and Trailer Patterns”, *Proceedings of the Eighth Indian Conference on Computer Vision*, Graphics and Image Processing, 2012.
62. C. A. Boiangiu, A. I. Dvornic. “Methods of Bitonal Image Conversion for Modern and Classic Documents”, *WSEAS Transactions on Computers*, Issue 7, Volume 7, pp. 1081 – 1090, July 2008.
63. C. A. Boiangiu, A. I. Dvornic, D. C. Cananau, “Binarization for Digitization Projects Using Hybrid Foreground-Reconstruction”, *Proceedings of the 2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, August 27-29, pp.141-144.
64. C. A. Boiangiu, A. Olteanu, A. V. Stefanescu, D. Rosner, N. Tapus, M. Andreica, “Local Thresholding Algorithm Based on Variable Window Size Statistics”, *Proceedings CSCS-18, The 18-th International Conference on Control Systems and Computer Science*, May 24-27 2011, Bucharest, Romania, Volume 2, Pp. 647-652.

65. C. A. Boiangiu, A. Olteanu, A. V. Stefanescu, D. Rosner, A. I. Egner, "Local Thresholding Image Binarization using Variable-Window Standard Deviation Response", *Annals of DAAAM for 2010, Proceedings of the 21st International DAAAM Symposium*, 20-23 October 2010, Zadar, Croatia, pp. 133-134.
66. B. Raducanu, C. A. Boiangiu, A. Olteanu, A. Ştefănescu, F. Pop, I. Bucur, "Skew Detection Using the Radon Transform", *Proceedings CSCS-18, The 18-th International Conference on Control Systems and Computer Science*, May 24-27 2011, Bucharest, Romania, Volume 2, Pp. 653-657.
67. D. Rosner, C. A. Boiangiu, A. Ştefănescu, N. Ţăpuş, A. Olteanu, "Text Line Processing for High-Confidence Skew Detection in Image Documents", *ICCP 2010 Proceedings*, Cluj-Napoca, Romania, August 26-28 2010, pp. 129-132.
68. C. A. Boiangiu, D. Rosner, A. Olteanu, A. V. Stefanescu, A. D. B. Moldoveanu, "Confidence Measure for Skew Detection in Photographed Documents", *Annals of DAAAM for 2010, Proceedings of the 21st International DAAAM Symposium*, 20-23 October 2010, Zadar, Croatia, pp. 129-130.
69. C. A. Boiangiu, B. Raducanu, "Effects of Data Filtering Techniques in Line Detection", *Annals of DAAAM for 2008, Proceedings of the 19th International DAAAM Symposium*, pp. 0125–0126.
70. C. A. Boiangiu, A. C. Spataru, A. I. Dvornic, D. C. Cananau, "Automatic Text Clustering and Classification Based on Font Geometrical Characteristics", *Proceedings of the 9th WSEAS International Conference on Automation and Information, WSEAS Press*, pp. 468 – 473, Bucharest, Romania, June 24-26, 2008.
71. C. A. Boiangiu, D. C. Cananau, B. Raducanu, I. Bucur, "A Hierarchical Clustering Method Aimed at Document Layout Understanding and Analysis", *International Journal of Mathematical Models and Methods in Applied Sciences*, Issue 1, Volume 2, 2008, Pp. 413-422.
72. C. A. Boiangiu, B. Raducanu, "3D Mesh Simplification Techniques for Image-Page Clusters Detection", *WSEAS Transactions on Information Science, Applications*, Issue 7, Volume 5, pp. 1200 – 1209, July 2008.
73. C. A. Boiangiu, A. V. Stefanescu, D. Rosner, A. Olteanu, A. Morar, "Automatic Slanted Edge Target Validation in Large Scale Digitization Projects", *Proceedings of the 21st International DAAAM Symposium*, 20-23 October 2010, pp. 131-132.
74. C. A. Boiangiu, D. C. Cananau, S. Petrescu, A. Moldoveanu, "OCR Post Processing Based on Character Pattern Matching", *20th DAAAM World Symposium*, pp. 141-144 Austria Center Vienna (ACV), 25-28th of November 2009.
75. C. A. Boiangiu, B. Raducanu, "Line Detection Techniques for Automatic Content Conversion Systems", *WSEAS Transactions on Information Science*, Applications, Issue 7, Volume 5, pp. 1200 – 1209, July 2008.
76. C. A. Boiangiu, D. C. Cananau, A. C. Spataru, "Detection of Arbitrary-Form Separators Based on Filtered Delaunay Triangulation", *Proceedings of the 9th WSEAS International Conference on Automation and Information*, WSEAS Press, pp. 442 – 445, Bucharest, Romania, June 24-26, 2008.
77. C. A. Boiangiu, D. C. Cananau, A. I. Dvornic, "White-Space Detection Techniques Based on Neighborhood Distance Measurements", *Annals of DAAAM for 2008, Proceedings of the 19th International DAAAM Symposium*, pp. 131 – 132.
78. C. A. Boiangiu, M. Zaharescu, I. Bucur, "Building Non-Overlapping Polygons for Image Document Layout Analysis Results", *The Proceedings of Journal ISOM*, Vol. 6 No. 2 / December 2012, pp. 428-436.