

DESIGNING A FLEXIBLE DOCUMENT IMAGE ANALYSIS SYSTEM – PART 1: THE ARCHITECTURE

Andrei Tigora^{1*}

ABSTRACT:

Archives of manuscripts, old printed books, papers and so on, are only available on paper support. Making them easily accessible to a wider public involves not only scanning them, but also processing them using OCR applications to extract the actual contents and make it searchable.

KEYWORDS: OCR, Digitization, Document Image Analysis, Document Export, Retroconversion

1. INTRODUCTION

The aim of this project is to evaluate the possibility of designing a modular OCR system, made of independent components, which could be dynamically configured at runtime. Most modern systems have a monolithic design, leaving the user little to no possibility of interfering with the configuration of intermediary processing steps. At best, the user has the possibility of choosing the format of the end file (PDF, Doc or text file), and in some cases selecting the document areas that would undergo content extraction. Another configurable component of modern systems is the dictionary which might be used to improve the word matching rate. Sometimes, more than one dictionary may be specified, but this is nothing more than a convenience feature, as all the contents could be part of a single file.

This paper is the result of the 2 year's work done for the master thesis of the author [30]. Other papers published during this period, which are connected to the presented subject and can add relevant information, which was omitted here for the sake of consistency, are [26] and [27]. They go into the details of preprocessing and contain detailed related work.

2. DOMAIN OVERVIEW

History of OCR Systems

Developing automated systems for character recognition has attracted the attention and imagination of scientists for a very long time. However, the technology to achieve it only became available around the end of the 19th century and the beginning of the 20th century, with the invention of scanning devices. In the interbellum period, a couple of patents were awarded for OCR machines, one to Gustav Tauschek in Germany in 1929 and another to Paul Handel in the US in 1933 [10]. And although these never materialized

^{1*} corresponding author, Engineer, "Politehnica" University of Bucharest, 060042 Bucharest, Romania, andrei.tigora@cti.pub.ro,

in actual machines at the time, due to technological limitations, the idea behind them, of matching characters to templates, would form the basis of many OCR systems to come.

Eventually, technology did catch up, thanks to the innovations brought about by the Second World War, so when a commercial opportunity arose in the 1950s in the domain of character recognition, financing and research in this domain took off. The opportunity itself came in the form of a challenge launched by the finance sector, which was looking for a unified mechanism for automatically processing checks - an activity that after the war had become the most wide-spread printed document processing activity. Eventually, the winning solution was Magnetic Ink Recognition (MIR), but there were also some proposals that relied on OCR [8].

In 1955 Reader's Digest bought the first commercial OCR[12], which was used for processing printed sale reports and generating punched cards, which would then be used as input for the computer that handled the delivery of 15 to 20 million books every year. The system itself had been developed by David S. Shepard, based on a patent obtained in 1953, in his company, Intelligent Machines Research Corporation.

Another system was sold to Standard Oil, for reading data from credit cards, so as to ease the bill generation procedure. Among the systems sold by IMRC till the end of the 1950s, there were also a bill reading system for Ohio Bell Telephone Company and a system for scanning pages used by the US Air Force. In the following years, IBM and other companies obtained the right to create systems based on Shepards's patents.

By the mid-1960s, Reader's Digest and Radio Communication of America were collaborating on developing an OCR system for identifying series numbers on Reader's Digest advertisement coupons. To do that, the text was printed with an RCA Drum printer, using the OCR-A, a special kind of font specifically designed to simplify the OCR process. The system itself was composed of a reader connected directly to an RCA 301 computer, followed by a specialized document reader that could process up to 1500 documents a minute. The documents would be either fully processed or rejected. Eventually, the system became part of product line offered by RCA, which could be used for processing bills.

Perhaps the company with the longest tradition and highest impact in OCR usage is the post system. In the USA, the postal service has been using an automated system since 1965. The receiver's address is identified by OCR analysis, and then a bar code is printed on the envelope, which is a lot simpler to read and interpret by subsequent classifiers in the chain [22].

Related Technologies

Because character detection is not 100 percent accurate and it can be relatively expensive computationally speaking, alternative systems have been designed to deal with interpreting printed information. In the following sections, an overview of these systems is presented, along with their advantages and disadvantages.

ICR Systems

Intelligent Character Recognition (ICR) systems process non-cursive hand written characters, which are then converted into digital characters. These systems mainly focus on digit recognition, which has a far greater success rate than letter recognition. The accuracy of this solution is lower than that of OMR and it requires additional editing and checks [12].

OMR Systems

An OMR (Optical Mark Recognition) detects the existence of a mark, not its shape. The technology is usually used for automatically processing forms that contain check-boxes that the respondent has to fill in. Although such a system is incapable of recognizing actual characters, limiting to some extent its flexibility, it displays the greatest processing speed of all file processing technologies. These systems are most often used for standardized testing, so they have to be able to differentiate between various grayscale intensities, which might mean the difference between a checked check-box and one whose content has been erased [12].

MICR Systems

Magnetic Ink Character Recognition is a special technology for identifying characters used by the financial industry to facilitate the processing of checks. Most US and UK checks include an MICR identification code that uses a special font E-138. Supported for recognizing this font is included in most modern OCR systems, without making use of the magnetic ink. However, sometimes, the background of check can have rather complicated designs that would interfere with the recognition mechanism, so there is a need to use the magnetic component of the ink to guarantee high recognition rates [9].

OCR Today

The main area of interest for using OCR systems nowadays is represented by the digitization of paper archives, which are of interest to the public due to their historic importance. The aim is indexing the contents of all the papers and allowing the public to search the contents by key words.

To evaluate the performances of an OCR system, two measurements are used: confidence and accuracy. Confidence indicates to what extent a system can rely on an identification it has performed and consists of assigning a (confidence) score to the proposed identification. Confidence starts from character level, then it can be aggregated over all characters within a page, a document or a set of documents. The accuracy score, regardless of its value, cannot indicate if the identification was correct or not. This can only be done by a human supervisor, who can manually determine the accuracy of the identification: how many characters (or possibly) words were correctly identified within the analyzed sample.

Following tests performed on a set of papers dating from the 19th and 20th centuries as part of a big digitization project, the accuracy for letters ranged between 71 and 98 percent [9]. According to this analysis, for some documents, approximately 30% of all characters are misclassified. If they were to be uniformly spread among all words, searching the document for keywords would become basically impossible. It is therefore quite likely that for older sources, as the quality of the documents drops, the number of recognition errors increases.

Apart from the accuracy of character identification, papers also pose the problem of segmenting and grouping text into logical units. By correctly determining the reading order of the contents of a paper page, titles and subtitles could be identified and provide an extra indexing mechanism.

The problems that appear in the final steps of document processing may have various causes, each of which can be dealt with by applying the appropriate correction technique. The most important factor is the quality of the physical document in itself, thus, while scanning the image it is important to use a resolution that allows to capture the relevant information without using too much storage space - usually, 300dpi is a good enough resolution. The image format is just as important, because it would be a shame to lose part of the information by storing the image using a lossy format.

What follows is a preprocessing step, which attempts to improve various features of the image, of which, one of the most important is foreground-background contrast enhancement [15]. Most often, this simply consists of binarization, but there are also other approaches which would rather use the image in its grayscale version. The next step attempts to determine the element positioning within the page, with a possible skew correction, if the document was not correctly aligned. Then comes the edge detection, process which might suffer due to a potential skew correction that might have interrupted existing edges. The edges or more precisely features of the edges are now used for character detection, with various degrees of accuracy. Characters are grouped into words based on closeness, confidence and matches in existing dictionaries [15]. Depending on the used OCR, the performance of this step may be improved by adding additional dictionaries or perhaps completing an existing one with toponyms or technical terms relevant to the contents of being analyzed. A step that is not actually part of the document analysis is training the OCR system itself [11]. Although it does not produce results directly, the impact it has on creating a database of relevant features for future documents might greatly improve the performance of the OCR system.

Unfortunately, despite all the advancements in the domain and the increasing processing power of modern computers, automatic processing cannot equal human performance; the intervention of human users is sometimes critical for correcting the output of such processing systems. Human intervention ranges from correcting the final text - most frequently - to setting custom parameters for various steps in order to respond to the features of the image being processed.

3. PROPOSED SYSTEM DESCRIPTION

The proposed solution has the main purpose of serving both as a functional (and good quality) Document Image Analysis System, and as an educational tool for students researching this field.

Application Structure

The system has a modular structure, each functionality being implemented by a standalone Executable.

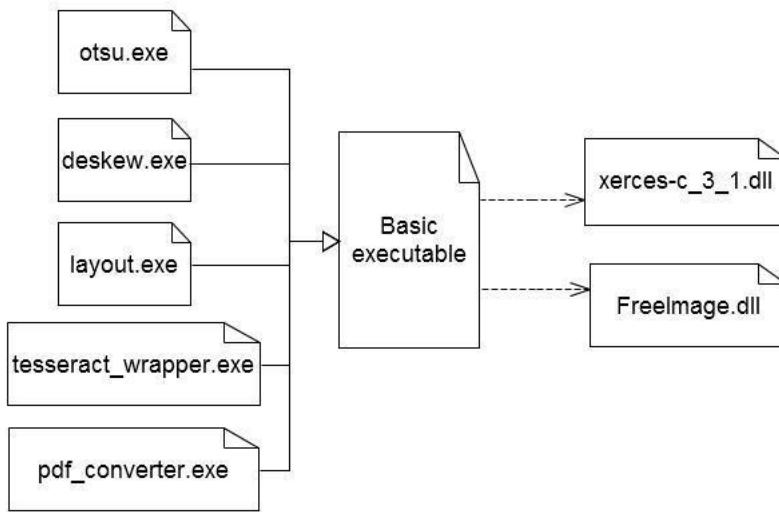


Figure 1. Component Diagram

All executables that compose the system interact more or less with images, so they need access to the FreeImage library. For example, while `otsu.exe` creates a new bitonal image based on the initial image, `pdf_converter.exe` on the other hand will use the library to make interrogations related to image features.

Another aspect common to all these executables is communication through XML files. XML files are used both for setting configuration and input parameters as well as reporting the results of the processing. Therefore, all executables must have the capacity to parse XML files.

Development

Typical Code Structure

All executables have as unique input argument the name of an xml file, which contains the parameters with which the program will run. The XML format is standard, and imposed by an XML Schema specially defined for this purpose. After the file is parsed,

application specific structures are built and initialized, storing the parameters that are to be used in the processing.

When the processing of the XML file is done, the initially populated structures are checked, to guard against invalid parameters. Aside from the processing result itself, whether it is a modified image or another file type, an auxiliary xml file will also be generated specifying if the processing was successful or not.

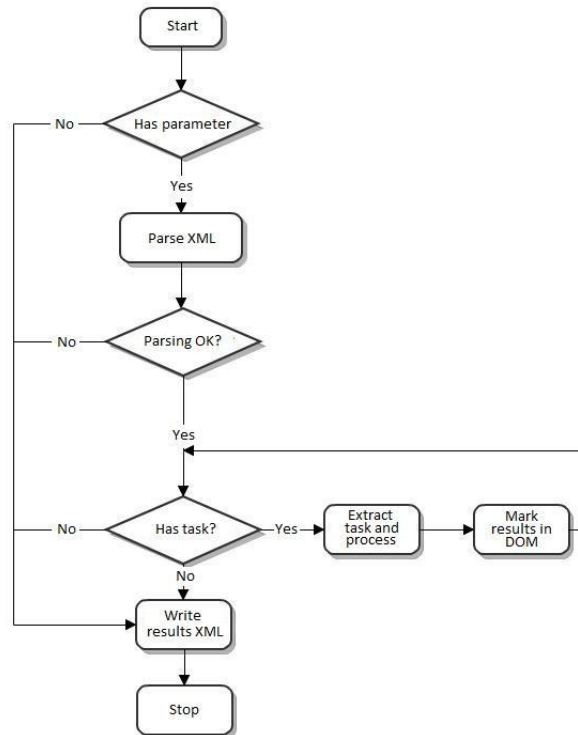


Figure 2. Code Execution Diagram

External Dependencies

Because some of the aspects of the system were auxiliary to the project, external libraries were used to solve the difficulties that arise.

FreeImage is a library developed under the GNU GPL license which offers support for the most popular image formats. The library offers minimal support for pixel manipulations, which reduces the memory costs of using it.

Xerces is an Apache library for manipulating XML files. The parsers included in the library are XML API compliant. The Xerces project was initially developed at IBM, both the Java and the C++ implementation. Eventually, Perl was added to the two languages, and due to attention it received, it ended up being top level Apache Foundation project.

As mentioned earlier, for the OCR processing itself, the system uses Tesseract. As it deals with images itself, Tesseract also uses a third party image processing library, and in this case it is ImageMagick; this library has a more generous interface, with more functionality, most of which however are not necessary to Tesseract.

Also, conversion from the METS/ALTO format to PDF is done through a third party binary, which also needs to access images. Luckily, it too uses ImageMagick, so there is no need for an additional library.

4. COMPARISON TO OTHER SYSTEMS

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, or specific document layout, for archives of documents whose format repeats over and over again [14][4]. 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 [19].

Document image analysis systems may be limited in scope to things such as extracting names from documents [18], or processing only tables, graphs and images [16]. These simplifications rely on the assumption that only certain information is worth being used for document indexing.

The system described in [7] 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 [11], without human intervention, the results tend to degrade significantly, so human intervention is compulsory for good results. The systems proposed in [6][14] use 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 [14][24].

Other systems allow the user to gain full control of their capabilities [17], 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 [25], 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 [24]. However, in order to eliminate the effort

of duplication, a single shared model of the documents should be used [21]. Crowd sourcing is also used for [13], in order to review the results of the automated processing and improve the quality of the end documents.

There are currently several large scale projects for digitization of national archives in countries such as Australia, Finland and the USA. Apart from the hardware support for scanning the documents, these also enjoy performant software dedicated to information extraction. In the following sections, a couple of these systems are described, and their architectural choices are compared to those made for the current project.

OCRopus

OCRopus is an open source software distributed under the Apache 2.0 license for document analysis and OCR processing. The last version of the system is 1.0, launched in April 2014.

Apart from the critical components that are written in C++, the system also has scripting support in languages such as python and LUA. As a result, the OCRopus components can be accessed either by linking the C++ dynamic libraries or through scripts [1].

The functionality of the system may also be extended by writing classes that implement existing interfaces or extend classes. Any class that implements one of these interfaces can afterwards be used in OCRopus as if it were a native component. Among the interfaces, there are some dedicated to image enhancement, page segmentation and classification interfaces [1].

Like the system that was presented up until now, OCRopus is also easily extensible; however, adding new components is conditioned here by implementing the interfaces, which are not I/O XML files, but C++ ones. OCR opus can be configured for either performance or flexibility, which is not possible in our proposed system, which was mainly designed having flexibility in mind.

OCRopus enjoys relative success, and the developers, once they become accustomed to the components, which are quite numerous, may create fast and relatively well performing image processing components. The disadvantage though is the fact that it requires getting accustomed with the components, which may be too big of a hurdle for developers that are not fully committed to this solution. On the other hand, using XML files to communicate with other components is a lot easier, as the technology is relatively familiar, and the developers may choose to implement their solutions using any library they find useful.

METAe

METAe was a research project founded by the European Commission that took place between 2000 and 2003. The aim was to create a software that would allow the conversion and archival of historic documents such as books, scientific publications, magazines and newspapers.

The METAe system simplifies the digitization of documents because it detects structural elements in printed documents without prior training. One of the more advanced processing that the system performs is the hierarchical analysis of the components, which leads to cataloguing the text zones as titles, subtitles, articles etc. and their grouping into logic sequences [2].

The results of the image processing may be saved in different formats typical for archives of digital documents. METS and PDF are some of the most popular in that respect [2]. The current system only outputs PDF as its final format, but given that the METS/ALTO is generated as an intermediary format, the possibilities of expressing the end result are basically limitless.

The developers of the system were aware that the implemented algorithms are not perfect and would not be able to cover all cases. As a result, the system allows humans to make low level modifications. This is also possible in the proposed system, through some of the graphical interfaces or by directly modifying the intermediary files.

METAe is hardly flexible, with all components being compiled into the final system; although the components can be configured through the use of carefully selected parameters, developing a new component implies recompiling the entire project.

5. CONCLUSION

The world of Document Image Analysis Systems is getting larger and larger. Unfortunately most of the proposed applications lack a full configuration for the processing flow and for the components involved. The system presented in this paper may be a solution which is easy to configure, fully customizable while also providing both robust everyday usage and good quality results.

REFERENCES

- [1] How to extend ocrpus in c++. <http://code.google.com/p/ocrpus/wiki/CxxProgramming>.
- [2] The METAe engine. <http://meta-e.aib.uni-linz.ac.at/metaengine/engine.html>.
- [3] 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.
- [4] B. S. Almeida, R. D. Lins, and 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, pages 146-153, 2011.
- [5] A. Boiangiu, A. C. Spataru, A. I. Dvornic, and C. C. Cananau. Normalized text font resemblance method aimed at document image page clustering. WSEAS Transactions on Computers, July 2008.
- [6] F. S. C. da Silva C. Antonio Peanho, H. Stagni. Semantic information extraction from images of complex documents. Applied Intelligence, 37(4), December 2012.

- [7] S.H. Kim C. B. Jeong. A document image preprocessing system for keyword spotting. Proceedings of the 7th international Conference on Digital Libraries: international collaboration and cross-fertilization, December 2004.
- [8] The Association for Automatic Identification and Capture Technologies. Optical character recognition (OCR). [http:// www.aimglobal.org/ technologies/ othertechnologies/ ocr.pdf](http://www.aimglobal.org/technologies/othertechnologies/ocr.pdf).
- [9] TROY Group. Micr basics handbook. [http:// www.troygroup.com/ support/ documents/ 50-70300-001_CMICRBasicsHandbook_000.pdf](http://www.troygroup.com/support/documents/50-70300-001_CMICRBasicsHandbook_000.pdf), Accessed: October 2004.
- [10] P. W. Handel. Statistical machine, June 1933.
- [11] R. Holley. How good can it get? Analyzing and improving OCR accuracy in large scale historic newspaper digitization programs. D-Lib Magazine, 15(3):1-13, 2009.
- [12] Phoenix Software International. Optical character recognition - what you need to know. [http:// www.phoenixsoftware.com/ pdf/ ocrdataentry.pdf](http://www.phoenixsoftware.com/pdf/ocrdataentry.pdf), Accessed: January 2009.
- [13] T. Ishihara, T. Itoko, D. Sato, A. Tzadok, and 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, pages 91-100, 2012.
- [14] C. Downton J. He. User-assisted archive document image analysis for digital library construction. Proceedings of Seventh International Conference on Document Analysis and Recognition, August 2003.
- [15] E. Klijn. The current state-of-art in newspaper digitization. D-Lib Magazine, January-February 2008.
- [16] L. Likforman-Sulem, P. Vaillant, and A. B. de la Jacopi re. Automatic name extraction from degraded document images. Pattern Analysis and Applications, 9(2-3), October 2006.
- [17] R. D. Lins, G. D. F. Pereira e Silva, and A. D. A. Formiga. Enhancing a platform to process historical documents. Proceedings of the 2011 Workshop on Historical Document Imaging and Processing, 0:169-176.
- [18] X. Lu, S. Kataria, W. J. Brouwer, J. Z. Wang, P. Mitra, and C. Lee Giles. Automated analysis of images in documents for intelligent document search. International Journal on Document Analysis and Recognition, 12(2), June 2009.
- [19] E. Mattha  ou and E. Kavallieratou. An information extraction system from patient histor-ical documents. Proceedings of the 27th Annual ACM Symposium on Applied Computing, page 787, 2012.
- [20] N. Otsu. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man and Cybernetics, 1979.
- [21] R. Sanderson, B. Albritton, R. Schwemmer, and 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, pages 175-184, 2011.
- [22] United States Postal Service. How a letter travels. [http:// about.usps.com/ publications/ pub100/ pub100_078.htm](http://about.usps.com/publications/pub100/pub100_078.htm).
- [23] R. Smith. An overview of the Tesseract OCR engine. Ninth International Conference on Document Analysis and Recognition, 2:629-633, 2007.

- [24] H. Toselli, E. Vidal, and 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, pages 219-222, 2010.
- [25] P. Tranouez, S. Nicolas, V. Dovgalecs, A. Burnett, L. Heutte, Y. Liang, and R. Guest. Docexplore: Overcoming cultural and physical barriers to access ancient documents. Pro-ceedings of the 2012 ACM symposium on Document engineering, pages 205-208, 2012.
- [26] Tigora, M. Zaharescu, "A Document Image Analysis System for Educational Purposes", Journal of Information Systems & Operations Management, 05/2013, 7(1), pp. 116-175.
- [27] Tigora, "An Overview of Document Image Analysis System", Journal of Information Systems & Operations Management, 12/2013; 7(2), pp. 378-390.
- [28] A. Boianuiu, A. V. Stefanescu, "Target Validation and Image Color Calibration", International Journal of Circuits, Systems and Signal Processing, Volume 8, 2014, pp. 195-202.
- [29] A. Boianuiu, A. I. Dvornic. "Bitonal Image Creation for Automatic Content Conversion". Proceedings of the 9th WSEAS International Conference on Automation and Information, WSEAS Press, pp. 454 - 459, Bucharest, Romania, June 24-26, 2008.
- [30] Tigora, "Document Image Analysis System", Master Thesis, Unpublished Work, Bucharest, Romania, 2013.
- [31] A. Boianuiu, I. Bucur, A. Tigora - „The Image Binarization Problem Revisited: Perspectives and Approaches", The Proceedings of Journal ISOM Vol. 6 No. 2 / December 2012, pp. 419-427.