

OCR-D & OCR4all: Two Complementary Approaches for Improved OCR of Historical Sources

Konstantin Baierer¹, Andreas Büttner², Elisabeth Engl³, Lena Hinrichsen³ and Christian Reul²

¹Berlin State Library, Germany

²University of Würzburg, Germany

³Herzog August Library Wolfenbüttel, Germany

Abstract

In order to leverage recent advances within the field of computer science for tasks ranging from full text search to various forms of quantitative analysis, an important desideratum in many areas of historical research is the availability of sources in the form of machine-actionable text. Here we present two complementary and free-to-use approaches, OCR-D and OCR4all. They deal with the conversion of scanned historical documents into such machine-actionable text and allow to tackle a variety of different use cases with diverse user requirements regarding both quantity and quality.¹

Keywords

OCR and transcription of old texts, Computer Vision applied to historical image collections, Mass digitization, Full text digitization, Handwritten Text Recognition (HTR), OCR-D, OCR4all

1. Introduction

We often have to trade off accuracy against volume when it comes to digitization efforts under practical constraints. Therefore, we need tools that are flexible enough to suit the characteristics of our sources, are able to fulfil the requirements of our research regarding quality, and provide enough throughput to compile necessary collections of data in time.

There are various open-source tools for every step in the OCR workflow (cf. Fig.1), including open-source OCR engines, such as Tesseract [1], Kraken [2], Calamari [3], and OCRopus [4], and proprietary software, like Transkribus [5] or ABBYY FineReader. In these and other state-of-the-art OCR services, recurrent neural networks (RNN) ensure both a low Character Error Rate (CER) and reliable performance. However, an open-source-service not only including engines but also covering the whole OCR-workflow, has been a desideratum. With the projects OCR4all, which develops an interactive platform for individual workflows, and OCR-D, focusing on mass digitization, we cover these problems and offer solutions for historians and other scholars.

¹This work was partially supported by the German Research Foundation (DFG), grant nos. 409784275 and 460665940.

HistoInformatics 2021 – 6th International Workshop on Computational History, September 30, 2021, online

✉ konstantin.baierer@sbb.spk-berlin.de (K. Baierer); andreas.buettner@uni-wuerzburg.de (A. Büttner); engl@hab.de (E. Engl); hinrichsen@hab.de (L. Hinrichsen); christian.reul@uni-wuerzburg.de (C. Reul)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).


 CEUR Workshop Proceedings (CEUR-WS.org)



Figure 1: Main steps of a typical OCR workflow, from left to right: preprocessing, binarization, segmentation / layout analysis, recognition, and postcorrection.

2. OCR-D

The OCR-D project [6] is tasked with developing software and workflows to augment the digitization of historical prints with OCR and layout recognition. It aims to make the German printed cultural heritage, in particular those works collected in the *Bibliography of Books Printed in the German Speaking Countries* (VD16–18)¹, accessible as high-quality research data for humanities researchers and the general public alike. For historians, being able to make new connections among masses of primary sources is particularly attractive, although this approach is very limited unless full text is available.

After a first phase (2015–2017) of exploration of the state of the art, engineering requirements and developing a functional model for the tasks ahead, a second phase of funding (2018–2020) brought researchers and developers from across Germany together to prototype solutions for particular areas in the domain where tools and methods were still underdeveloped. In the current phase 3 of OCR-D (2021–2024), these prototype workflows will be made production-ready, to be deployed in various implementation scenarios.

The main focus of OCR-D is on flexibility and interoperability. Since the prints to be recognized vary wildly in terms of age, state of preservation, language, script, layout etc., it is essential that OCR-D workflows are composable of single-task tools, which in compound cover all possible steps of an OCR process, to tailor workflows exactly to the print at hand. With currently over fifty powerful tools, OCR-D offers users several possibilities for most OCR steps. OCR-D processing is based on the METS file format while information on layout and recognized text is stored in the PAGE-XML format,[7] with optional conversions to further formats like ALTO-XML, another standard used in libraries. To ensure an optimal integration of OCR-D in mass digitization, the software can be run via the CLI or Python API.

While the main focus of OCR-D is mass digitization of VD prints, we strive for transparency and openness in the development process and aspire to include the wider OCR/HTR (Handwritten Text Recognition) community in our developments. Besides an active chat,² we have regular calls that are open to the public, both meetings on the technical details of software development, specification engineering and Ground Truth as well as low-threshold events for new or potential users without a technical background.³

¹<https://vd16.de>, <http://vd17.de>, <http://vd18.de>.

²<https://gitter.im/OCR-D/Lobby>

³<https://ocr-d.de/en/platforms>

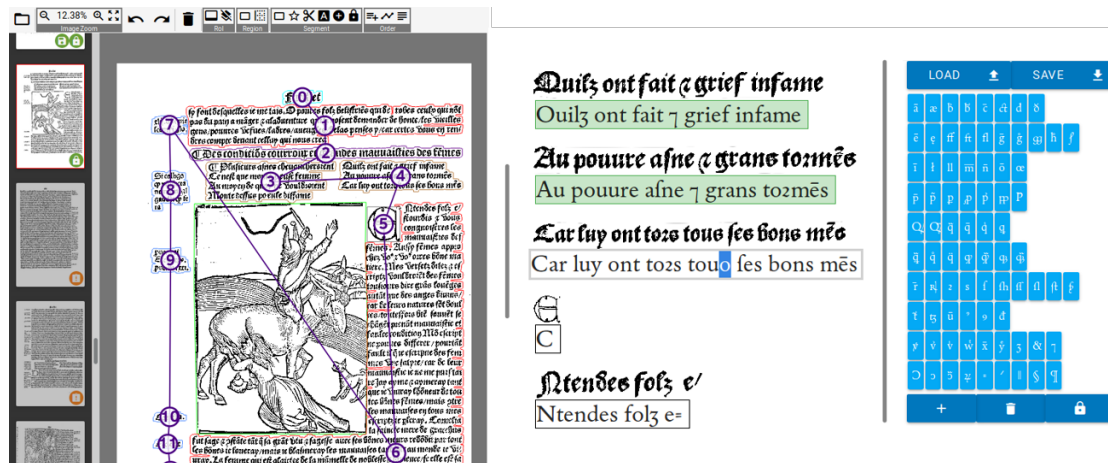


Figure 2: Open-source tool LAREX (Layout Analysis and Region Extraction).

3. OCR4all

The open-source software OCR4all⁴ [8] allows users even without any technical background to process printed and handwritten historical documents independently and with great quality via a GUI (Graphical User Interface). Further cornerstones are flexibility, interactivity, and configurability. Depending on the material at hand and the quality requirements of the user, the best way to approach the OCR can vary considerably, from a fully automated pass through to a highly interactive workflow with several manual correction steps in-between. OCR4all supports this via the so-called “process flow”, which allows to combine modules that can run fully automatically into a pipeline.

To ensure a reliable and low-threshold installation process as well as platform-independence, the main application and all of its submodules are encapsulated in a single container which is then provided as a Docker or VirtualBox image. Combined with the underlying client/server architecture this allows for a flexible deployment and usage either on a single user’s local PC or as a centrally hosted web application, enabling users remote access via their browsers.

Currently, the preprocessing and line segmentation steps are performed by OCRopus [4] while Calamari [3] takes care of the OCR/HTR related steps of recognition, training, and evaluation. Another key component is LAREX [9] (cf. Figure 2) which visualizes and allows to correct most (intermediate) results that are produced during the entire OCR workflow and are stored in the PAGE format. Among others this includes region and line coordinates, semantic region types (heading, marginalia, ...) the reading order, and the result of the text recognition.

Manual corrections are not limited to enhancing existing results but can also be fed back into the workflow to improve the output of the fully automated steps: By correcting the OCR/HTR output on a line-by-line basis the corrections can then be used to train a new work-specific model that is better suited to the work at hand. Applying the obtained model to further lines

⁴<http://ocr4all.org>

leads to a lower CER, allowing for an even faster correction and consequently a more efficient production of training material. This so-called “iterative training approach” can be repeated until a satisfactory CER is reached or the entire book is transcribed.

4. Conclusion and Outlook

The goal is to offer a wide variety for each individual step and let the users choose freely among them, based on the material at hand, available resources (for example OCR/HTR models), or simply personal preference.

To enable this in an extendable and future-proof way, a flexible and interoperable access to more OCR solutions for individual steps of the workflow is required. As this is exactly what the OCR-D project offers, combining both approaches is an obvious next step and will be tackled within the DFG-funded project “OCR4all-libraries” which is part of the third OCR-D phase. The goal of the project is to incorporate the OCR-D solution into OCR4all and make it available via the GUI, offering non-technical users a low-threshold access. Experienced users can also profit from this, e.g. by making use of LAREX as a visual explanation component to compare different workflows and configurations or to solve problems. Combined with the integrated options to produce Ground Truth, train more specialized models, and manually correct the OCR output, if desired, this allows a flexible usage in different scenarios.

Furthermore, employing the same components in OCR4all and OCR-D entails that data resulting from smaller projects targeted at specific sources can later be used to improve the results of mass digitization efforts. Models optimized for the recognition of a single work can be reused for other volumes or books from the same hand or printer, thus providing a scalable workflow. The annotations created by transcribing with OCR4all provide a valuable contribution for the training of mixed models which feature a better robustness with regards to variations in font, condition of the source document, and the quality of the digital images.

In the future, we are looking forward to further cooperation with projects aimed at the digitization of historical text documents. This will give us an opportunity to identify requirements not yet covered by existing solutions, e.g. of encoding special kinds of textual data or information on the digital provenance to provide a reliable basis for source criticism, thereby allowing us to keep improving the OCR of historical sources.

References

- [1] R. Smith, An overview of the tesseract ocr engine, in: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), volume 2, 2007, pp. 629–633. doi:10.1109/ICDAR.2007.4376991.
- [2] B. Kiessling, Kraken - an Universal Text Recognizer for the Humanities, DH 2019 Digital Humanities (2019). doi:10.34894/Z9G2EX.
- [3] C. Wick, C. Reul, F. Puppe, Calamari - A High-Performance Tensorflow-based Deep Learning Package for Optical Character Recognition, Digital Humanities Quarterly 14 (2020). URL: <http://www.digitalhumanities.org/dhq/vol/14/2/000451/000451.html>.

- [4] T. M. Breuel, The OCRopus open source OCR system, in: Document Recognition and Retrieval XV, volume 6815, International Society for Optics and Photonics, 2008, p. 68150F.
- [5] P. Kahle, S. Colutto, G. Hackl, G. Mühlberger, Transkribus - a service platform for transcription, recognition and retrieval of historical documents, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), volume 04, 2017, pp. 19–24. doi:10.1109/ICDAR.2017.307.
- [6] E. Engl, OCR-D kompakt: Ergebnisse und stand der forschung in der förderinitiative, Bibliothek Forschung und Praxis 44 (2020) 218–230. doi:doi:10.1515/bfp-2020-0024.
- [7] S. Pletschacher, A. Antonacopoulos, The PAGE (page analysis and ground-truth elements) format framework, in: 2010 20th International Conference on Pattern Recognition, IEEE, 2010, pp. 257–260. doi:10.1109/ICPR.2010.72.
- [8] C. Reul, D. Christ, A. Hartelt, N. Balbach, M. Wehner, U. Springmann, C. Wick, C. Grundig, A. Büttner, F. Puppe, OCR4all—An Open-Source Tool Providing a (Semi-) Automatic OCR Workflow for Historical Printings, Applied Sciences 9 (2019) 4853. doi:10.3390/app9224853.
- [9] C. Reul, U. Springmann, F. Puppe, LAREX: A semi-automatic open-source tool for layout analysis and region extraction on early printed books, in: Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage, DATECH2017, ACM, New York, NY, USA, 2017, pp. 137–142. doi:10.1145/3078081.3078097.