

Handbook of Digital Egyptology: Texts

UAH MOA 01

Handbook of Digital Egyptology: Texts

Edited by Carlos Gracia Zamacona & Jónatan Ortiz-García



Universidad
de Alcalá

EDITORIAL
UNIVERSIDAD DE ALCALÁ

Con el patrocinio de la Comunidad de Madrid, Proyecto *The Earlier Ancient Egyptian Mortuary Texts Variability* (www.mortexvar.com), Programa Atracción de Talento 1 (2018-T1/HUM-10215).



El contenido de este libro no podrá ser reproducido,
ni total ni parcialmente, sin el previo permiso escrito del editor.
Todos los derechos reservados.

- © De los textos: sus autores.
- © De las imágenes: sus autores.

© Editorial Universidad de Alcalá, 2021
Plaza de San Diego, s/n
28801 Alcalá de Henares
www.uah.es

I.S.B.N.: 978-84-18979-09-5
Depósito legal: M-34533-2021

Composición: Solana e Hijos, A. G., S.A.U.
Impresión y encuadernación: Solana e Hijos, A.G., S.A.U.
Impreso en España

TABLE OF CONTENTS

1. Digital Egyptology: A very short introduction CARLOS GRACIA ZAMACONA & JÓNATAN ORTIZ-GARCÍA <i>Universidad de Alcalá</i>	9
2. Formatting of ancient Egyptian hieroglyphic text MARK-JAN NEDERHOF <i>University of Saint Andrews</i>	17
3. Digital writing of hieroglyphic texts SERGE ROSMORDUC <i>CEDRIC/Vertigo/Conservatoire National des Arts et Métiers, Paris</i> .37	
4. Annotating texts on 3D (coffin) models RITA LUCARELLI <i>University of California Berkeley</i>	55
5. Uncovering Old Kingdom society arrangement: Detection of powerful dignitaries using complex network analysis VERONIKA DULÍKOVÁ & RADEK MAŘÍK <i>Charles University / Czech Technical University, Prague</i>	69
6. Compiling the lexicon of ancient Egyptian: State of the art SIMON SCHWEITZER <i>Berlin-Brandenburgische Akademie der Wissenschaften, Berlin</i>	103
7. How to compute a shape: Optical character recognition for hieratic BERNHARD BERMEITINGER, SVENJA A. GÜLDEN & TOBIAS KONRAD <i>University of St. Gallen / Akademie der Wissenschaften und der Literatur Mainz</i>	121
8. The State of the Affairs in Optical Character Recognition (OCR) for Coptic ELIESE-SOPHIA LINCKE <i>Humboldt-Universität zu Berlin</i>	139

9. The Polychrome Hieroglyph Research Project: A new method for recording, displaying and analysing the colours used in monumental inscriptions	
DAVID NUNN	
<i>Université Libre de Bruxelles</i>	165
Index	181

8. THE STATE OF THE AFFAIRS IN OPTICAL CHARACTER RECOGNITION (OCR) FOR COPTIC

ELIESE-SOPHIA LINCKE

Institute of Archaeology, Humboldt-Universität zu Berlin¹

ABSTRACT

Optical Character Recognition (OCR) forms part of text digitization workflows, useful especially where larger amounts of text are to be digitized. It is a process in which text is extracted from digital images (mostly scans or photographs), whereby the bitmap image of a page (in our case from the edition of an ancient work) is converted into machine-readable text. The result is a text file, either in plain text format (*.txt) or in an enriched xml format. Modern OCR software does not only match pixel-based shapes to Unicode characters, it uses Artificial Intelligence (AI) in order to learn a language model and make predictions about characters expected in context. This approach has proven very powerful with respect to different types of pattern and language recognition (e.g. speech recognition, word recognition, handwritten text recognition and, of course, OCR). For Coptic printed text, the models trained by the *Coptic OCR* project reach character accuracy rates of 98.5% to 99.6% (depending on the font). This article will describe a state-of-the-art OCR workflow for Coptic printed texts and give an overview of the tools used as well as of the underlying AI techniques.

KEYWORDS

Coptic – OCR (Optical Character Recognition) – digitization of textual cultural heritage – Artificial Intelligence – Neural Networks – Machine Learning.

¹ The author would like to thank Heike Behlmer, Marco Büchler, Kirill Bulert, Camilla Di Biase-Dyson, Frank Feder, Jürgen Knauth, So Miyagawa, Tobias Paul, Malte Rosenau, Caroline Sporleder, Ronnie Vuine and Jörg Wettlaufer for encouragement, hosting the project, their input in discussions and/or technical support.

1. INTRODUCTION

With services such as retrodigitization of back issues of scientific journals or of monographs and collected volumes in Google Books, we have become used to the easy access to printed texts in digital form and other advantages that go along with them, in particular full-text searches. Commercial software such as Adobe Acrobat Pro integrates OCR software and makes it possible to recognize text in scanned documents with just a couple of clicks. The precondition of all this is OCR software (including a so-called OCR model) that has learned to recognize text in the respective script. For Coptic texts, automated text recognition has not been possible until recently because such OCR software simply had not been trained on Coptic.² Consequently, the digitization of Coptic texts was a fully manual, time-consuming, sometimes tedious undertaking.

Occasionally, Optical Character Recognition is used as a cover term for recognition of written language (printed, handwritten). But usually, it only refers to the recognition of printed texts as opposed to Handwritten Text Recognition (HTR, also HWR, Handwriting Recognition). Coptic texts are mostly published using professional print, sometimes also as a reproduction of typewritten manuscripts or even the editor's handwriting. More recently, computer fonts (first non-Unicode, later also Unicode) have replaced typesetting. Hitherto, Coptic OCR focused on typeset Coptic in order to help digitize the substantial back catalogue of editions of Coptic texts. Other types of fonts (typewriter and computer fonts) as well as Handwritten Text Recognition (handwriting of ancient scribe or modern editor) have not yet been tackled.³

OCR is an offline recognition task (whereas with HTR, there is also online text recognition, i.e. recognition during writing). In contrast to the fully automated mass digitization of large collections of modern text (e.g. newspapers), it will probably always result in a semi-automatic workflow, including human input for the correction of imperfect OCR results, due to the high expectations concerning accuracy in philological disciplines.

This article provides an overview of Optical Character Recognition (OCR) for printed Coptic texts from scholarly editions, its workflow, the state-of-the-art software and, very briefly, its computational foundations. It is based on a study run by the author of this paper during a fellowship at the CampusLab Digitization and Computational Analytics (DCA) at the Göttingen Centre for Digital Humanities in 2018.⁴ Since then, the work has continued under the project name *Coptic OCR*.

² There has been some pioneering work by Moheb S. Mekhael in the 2000s for which see Section 0 and Schroeder 2020: 330. This has, however, not had an impact on the digitization of Coptic texts.

³ With significant technical advances in the field of HTR, it is definitely not impossible to obtain good results not only for handwritten Coptic texts in editions but also for Coptic manuscripts.

⁴ Beforehand, there had been pilot studies by So Miyagawa at the University of Göttingen and by myself at the Humboldt-Universität zu Berlin. The advances in OCR for Coptic are supported by Kirill Bulert and

The project collected and prepared training data for several widely used Coptic fonts (see Fig. 3) and has trained OCR models for these fonts and evaluated them. The recognition rates (character accuracy rates, see Fig. 2) are promising and comparable to results achieved for fonts of other historical and minority scripts.⁵ When information on OCR for Coptic is given below, the underlying findings and results are from this project unless otherwise noted. Depending on future funding, the project will continue and expand its work.⁶

This introductory paper will orient the reader towards what is possible in Coptic OCR today, what data and tools are available and where to find more information.⁷ In order to do so, the paper is structured in the following way: Section 2 deals with the processing of data in OCR from two different perspectives, (1) the end-user perspective, that is the digitization workflow given that the necessary OCR tools are available, and (2) the training perspective, that is the steps and choices necessary to create OCR models. Section 3 provides an overview of software (and some remarks on hardware) that is useful for Coptic OCR and briefly sketches relevant features. Section 4 introduces the state of affairs with respect to digitization of Coptic texts, including some remarks on the relationship between automated and manual digitization procedures. A commented bibliography in Section 5 concludes the paper.

2. HOW OCR WORKS

2.1. OCR workflows

The workflow of automated digitization (OCR) is different from manual digitization. Fig. 1 presents the process in form of a flowchart. There are two different OCR workflows, depending on the objectives of the user. The first one is the end-user workflow with the aim to digitize text. The other one is the training workflow in which OCR models are built. This is actually the prerequisite for the end-user to be able to recognize text. Both workflows share the preparatory steps (“Preparation” and “Preprocessing”). Afterwards, they split up into two branches: (1) OCR in the narrower sense with “Recognition” and “Postprocessing” and (2) Machine Learning

Marco Büchler who provided the necessary technical infrastructure, for instance access to a high-performance server and a GitLab data repository hosted at the GWDG in Göttingen.

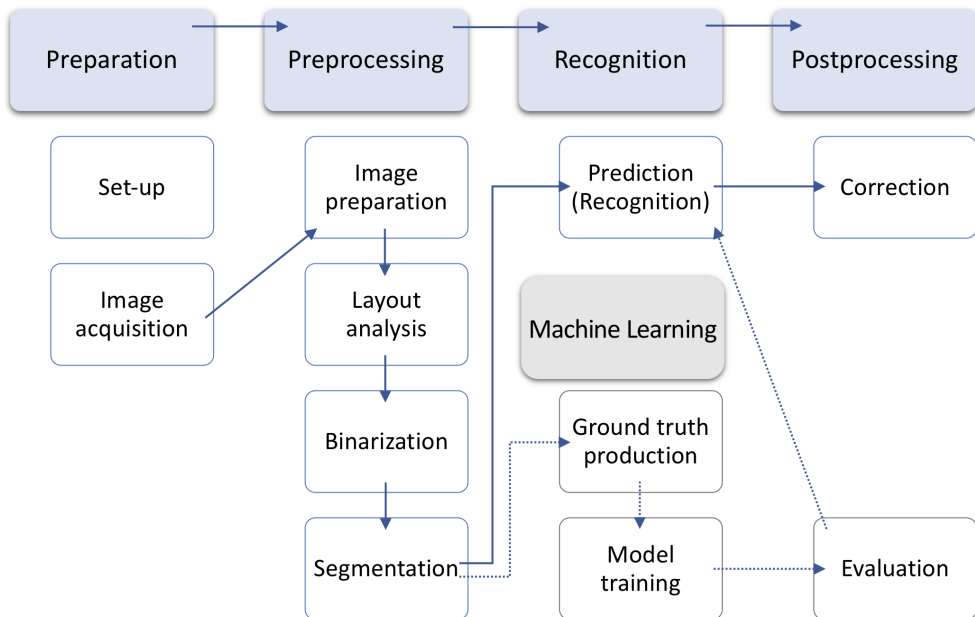
⁵ First results are published in Lincke et al. 2019.

⁶ Training data and OCR models are currently available in the GitLab repository of *Coptic OCR*, DOI: 21.11101/0000-0007-C9D1-A; PID: 21.11101/0000-0007-C9D1-A.

⁷ Not included in this introduction is a description of the output formats that have been developed for OCR output (PAGE XML, ALTO XML, hOCR) since these are closely linked to the question of metadata standards that cannot be addressed in this concise paper. All OCR software mentioned in this paper also outputs plain text files (*.txt).

which includes the training of OCR models, that is, the adaptation of a Neural Net so that it can perform a recognition task, as well as their evaluation.

FIG. 1. OCR WORKFLOW



2.2. Preparation

The OCR workflow differs from manual text input in that it requires preparatory steps before the actual digitization (recognition). Firstly, there is the acquisition of digital images (preferably scans) and secondly, the installation of the preprocessing tools and OCR software (technical set-up). Image acquisition is comparably easy, especially since a considerable number of editions of Coptic texts are available in online archives (e.g. archive.org) or on other websites so that scanning can be skipped. The high accuracy rates of Coptic OCR reached by means of training on existing scans show that their quality is generally sufficient. Image acquisition will therefore not be considered any further in this paper.⁸ The technical set-up will be addressed in Section 3.

⁸ Robertson 2019 gives some insights in how far image quality and enhancement influences the output quality of OCR.

2.3. Preprocessing

Depending on the quality of the print itself and the quality of the digital image of the print, and other than in manual text digitization, preprocessing usually is the most time-consuming part of the workflow because it requires the most human intervention. This is one of the reasons why this implementation of OCR is only semi-automatic.

What is summarized in the workflow under the term image preparation consists of several steps. Not all of them are compulsory depending on the properties of the digital images that are processed. Image preparation includes:

- rotation – adjustment of the orientation,
- deskewing – balancing the orientation so that the text lines are strictly horizontal,
- splitting – split 2-on-1 scans so that there is one image file per page,
- despeckling – noise removal, i.e. removal of stains and blotches on the digital image,
- dewarping (usually necessary only when historical prints are warped),
- content selection – selection of text to be OCRed (deselecting/deleting translation, critical apparatus, margins etc.),
- reformatting – from pdf format into tiff or png

Layout analysis is not mandatory and only necessary when the content of the whole scanned page is of interest or shall be preserved (no content selection during image preparation). We have included it here for future reference when other than in the current workflow the recognition process shall not be limited to the Coptic text part but include the non-Coptic content of the document (e.g. commentary, critical apparatus, translation and so on) in a multi-lingual OCR process.

In the final step, Segmentation, the text image is cut up so that as a result there is one image file per text line.

OCR software generally excludes image preparation, sometimes also the other steps of preprocessing. This is then outsourced to software specifically designed for this purpose (e.g., ScanTailor and Larex in Section 3).

2.4. Prediction

Prediction is the text or character recognition in the narrower sense. This is a fully automated step in the process. The Neural Network in the OCR model extracts features from the forms (characters) consisting of pixels on an image and classifies them, that means, assigns (“predicts”) a Unicode character. Some basic information on the underlying AI can be found in Section 2.8.

2.5. Postprocessing

Postprocessing, that is the correction of the OCR result, is essentially the same as in a manual digitization workflow and also a reason for the classification of this workflow as semi-automatic. However, it can be computer-assisted, e.g. by a line-by-line display of the editable OCR result and a snippet of the respective line from the input image. Such a display facilitates the mapping of input image and OCR output and thereby makes it easier to spot errors in the OCR output. In more advanced workflows of postprocessing (not yet implemented for Coptic), specialized tools⁹ also make suggestions for corrections based on dictionaries or word lists.

2.6. Model training and evaluation

OCR programs use training data as input to train Neural Networks (for which see Section 2.8). Training data consists of the segmented images of text lines produced in preprocessing and the so-called Ground Truth, transcriptions of these text lines. The result of training, the trained Neural Net, is called a (OCR) model. In our case, models for Coptic were trained in the OCR programs OCRopus and Calamari.

As for the amount of training data, Springmann recommends a minimum of 300 text lines for training with OCRopus and additional lines in further training cycles if necessary.¹⁰ This includes 10% of training data reserved for evaluation. Reul et al. suggest 300 lines of Ground Truth for training with Calamari (plus 100 for testing).¹¹ Due to the demands on the quality and representativeness of the data (for which see Section 2.7), the number of training lines for Coptic models exceeded these recommendations.

During training, for each line of text, the Neural Network makes a prediction about the text (string of characters) that an image file (of a line of text) depicts. This prediction is then compared to the man-made Ground Truth and the result is fed back into the Neural Network in order to stimulate its adaptation and to improve the prediction. The OCR program proceeds line by line, several times for every line of training data. Every turn is called an iteration.

For the software OCRopus, the recommended number of iterations is at least 30,000.¹² The training process of Calamari, another more advanced software, is slightly different because cross-fold-training is implemented. Cross-fold-training means that the training data is split into equal batches (the default is five batches) and as many models are trained as there are batches. Because there is a random element in the

⁹ For instance, PoCoTo for which see Section 3.6.

¹⁰ Springmann 2015.

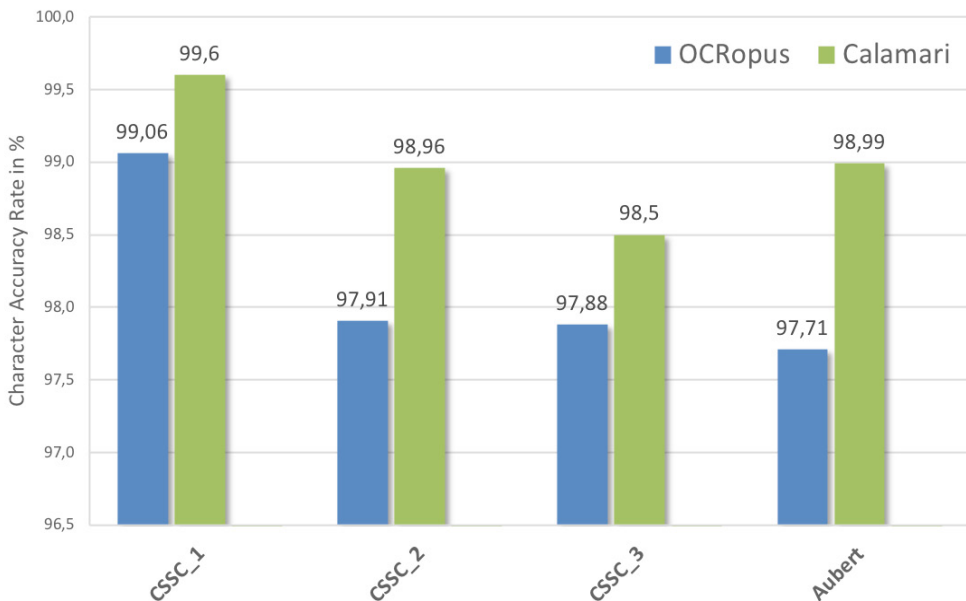
¹¹ Reul et al. 2017b.

¹² Springmann 2015.

training process two models will never be identical, in other words, they will make slightly different predictions. The benefit of this is, that if one model is mistaken about one character, it is very unlikely that other models will be too, so this observation can be used to eliminate prediction errors. Models not only predict characters, they also give a confidence value (in percent), i.e., an estimate of how certain they are that one or another character is correct. These confidence values are used in what is called confidence voting. If the models disagree in their predictions, the confidence values are added up for each candidate and preference is given to the character with the highest confidence value.¹³ This combined use of multiple well-trained models is what makes OCR outputs produced by Calamari superior to results obtained with OCRopus.

OCRopus and Calamari each include an evaluation mechanism that outputs the character accuracy rate of an OCR model (or a batch of OCR models) for the test sample that the Neural Network has not seen during training, i.e. that has not been part of the training data. Fig. 2 shows a comparison of the best character accuracy rates reached by Calamari and OCRopus respectively, based on training with identical training data.¹⁴

FIG. 2. CHARACTER ACCURACY RATES OF OCR MODELS FOR COPTIC FONTS COMPARED



¹³ Note that this is different from a simple majority vote (e.g. 3 : 2). For a more detailed explanation and examples, cf. Reul et al. 2018.

¹⁴ The models are trained for specific fonts labelled CSSC_1 ... Aubert, see the next section (Section 2.7).

2.7. Training Data

An OCR model is only as good as the training data used in its creation and it can only make predictions based on what it has learned in training. For this reason, training data is an important factor in the subsequent performance of an OCR model. In other words, in order to perform well, the composition of the training data depends on the needs of the later application of the OCR models.

This is why digitization projects that train their own OCR models make differing decisions with respect to model training. For instance, when digitizing very early printed books (e.g. incunabula from the late 15th century A.D.), the *Kallimachos* centre,¹⁵ trains a specific model for each book due to the range of variation in early printing technique.¹⁶ Springmann and Lüdeling, working on OCR models for the RIDGES corpus of herbal texts dating between 1487 and 1870, separately trained mixed models, i.e. multi-font models (Antiqua and Fraktur fonts) for each book in their training corpus.¹⁷ Some of their models also performed well on other books (>90% character accuracy, sometimes even >95%) but most did not.¹⁸ By contrast, commercial OCR software for modern texts (e.g. ABBYY FineReader) usually performs well on a large variety of fonts, regardless of the document.

Scholars working on OCR for Coptic have chosen a different approach and decided to train font-specific models.¹⁹ This is based in the observation that a Coptic font has generally been used over a longer period and in a certain number of editions so that it makes sense to create font-specific models that can be used to OCR all books printed in this font. Usually only a single font is used for the reproduction of a (literary) Coptic text (unlike, for example, in the herbal texts of the RIDGES corpus where more than one font is used within the same text). Of course, critical apparatus, commentary, translation etc. are printed in other (Latin) fonts. However, since the primary aim of OCR for Coptic is to digitize Coptic texts for further processing, not to create a digital reproduction of an already existing edition, these components have been disregarded for the time being.

Fig. 3 shows the fonts chosen by the *Coptic OCR* project for model training in Calamari and OCRopus.²⁰ Each font is represented in screenshots from two different editions in order to demonstrate the variation within the

¹⁵ See below, Section 3, “Larex” and “OCR4all”.

¹⁶ Reul et al. 2017b.

¹⁷ Springmann – Lüdeling 2017.

¹⁸ Springmann – Lüdeling 2017: Fig. 6.

¹⁹ Lincke et al. 2019; Miyagawa et al. 2017 and Miyagawa et al. 2019 also trained a particular font.

²⁰ Models created by the *Coptic OCR* project, available in the project repository: <<https://vcs.etrap.eu/Coptic-OCR/datasets/>>, DOI: 21.11101/0000-0007-C9D1-A [accessed: 7/15/2020].

the writing system of a language.²³ In addition to the core set of Coptic lower case (minuscule) characters,²⁴ it includes:

- upper case (majuscule) and ornamental (ekthetic) forms of the core characters, cf. Fig. 3, font Aubert, right side
- characters found only in some dialects or chronolects, e.g., Bohairic ⲛ (U+03E6) or Akhmimic ϣ (U+2CC9), see Fig. 3, font CSSC_1, with Sahidic on the left and Bohairic on the right side
- Coptic numerals, e.g. Ⲉ (Coptic Epact Digit Six, U+102E6)
- a great variety of combining diacritics:²⁵ trema ̈ (Combining Diaeresis, U+0308), macron ̄ (Combining Macron, U+0304), ̅ (Coptic Combining Ni Above, U+2CEF), more complex diacritics forming sequences like ̅̅̅ (Combining Macron Left Half, U+FE24, Combining Conjoining Macron, U+FE26, Combining Conjoining Macron, U+FE25), ̆̆ (Combining Double Circumflex Above, U+1DCD), ̇̇ (Combining Double Tilde, U+0360) and more
- abbreviations (ϣ, Kai, U+2CE4), symbols like Ⲡ (Tau Ro, U+2CE8), ⲡ (Ki Ro, U+2CE9), Ⲣ (Shima Sima, U+2CEA) and others, cf. Fig. 3, CSSC_1 left side
- Coptic punctuation, e.g. ⋅ (Middle Dot, U+00B7), ∴ (Three Dot Punctuation, U+2056), ∴ (Four Dot Punctuation, U+2058), ˘ (Coptic Morphological Divider, U+2CFF) and more
- editorial signs:
 - brackets, in addition to the generally known ones also ⌈ (U+27E6) and ⌋ (U+27E7), † (U+2E22) and ‡ (U+2E23), † (U+23A8) and ‡ (U+23AC) and others;
 - footnote signs such as superscript Latin lower case letters and superscript Arabic numerals;
 - underdots for damaged characters (̣, Combining Dot Below, U+0323), cf. CSSC_2 right side and CSSC_3 and Aubert left side for a variety of editorial signs;
 - Arabic and/or Latin numerals for page, paragraph and/or line numbering;
 - other signs indicating page/folio breaks in the original manuscript or a switch between manuscripts attesting the respective passage, e.g. Ⓞ (U+2299), □ (U+2311)

²³ Thea Sommerschild recently mentioned in her talk “PYTHIA: a deep neural network model for the automatic restoration of ancient Greek inscriptions” at the Digital Classicist London Seminar (05 June 2020) that, for the encoding of Ancient Greek inscriptions in her corpus, 147 characters are required, <<https://youtu.be/nKSfzHYmLtQ>> [accessed: 7/2/2020].

²⁴ Character names according to the Unicode Standard but see also Kasser 1991.

²⁵ ◦ (U+25CC) merely serves as a placeholder for Coptic letters that could combine with respective diacritics.

Not all characters witnessed in Coptic printed texts are already represented by a code point in the Unicode standard (esp. punctuation, e.g. ∙). In these cases, placeholders (currency symbols such as €, \$ and ¥) were used in the transcriptions.²⁶

In order to include the character set of a font as completely as possible in the training, the pages for training were not chosen randomly, but in such a way that even very rare characters are represented.

Another deliberate decision with regard to the robustness of the OCR models was not to simplify the data, i.e. not to clean up the data by removing diacritical and editorial characters (brackets, characters indicating lacunae and the like). This has two advantages: less time spent on image preparation and an OCR output that is closer to the “original” printed text. As a result, later users of the digitized texts can decide for themselves whether they want to remove diacritics etc. or not. The transcriptions were also not normalized, neither in terms of orthography nor in terms of character set (e.g. by homogenizing diacritics, transcribing characters such as tildes as the more common macrons). This is despite the fact that the choice of Unicode characters as equivalents to printed glyphs can be considered a normalization process in itself. The transcriptions were made with the aim of not invalidating differences made in the printed version of the texts. This includes, for example, the transcription of capital letters where they are used in the respective editions, although Coptic manuscripts do not distinguish between lower and upper case and although this does not comply with the transcription guidelines of the Kellia network.²⁷

Coptic texts as represented in editions (as well as in manuscripts) also differ with respect to line length and column layout (one column vs. two columns). This may be relevant for OCR model training as modern OCR software not only considers one individual character at a time, but also takes into account a whole string of characters, i.e., the adjacent characters, both to the left and to the right (see Section 2.8). Therefore, texts with differences in line lengths and column number were included in the training data (cf. Fig. 3).

2.8. Artificial Intelligence in Optical Character Recognition

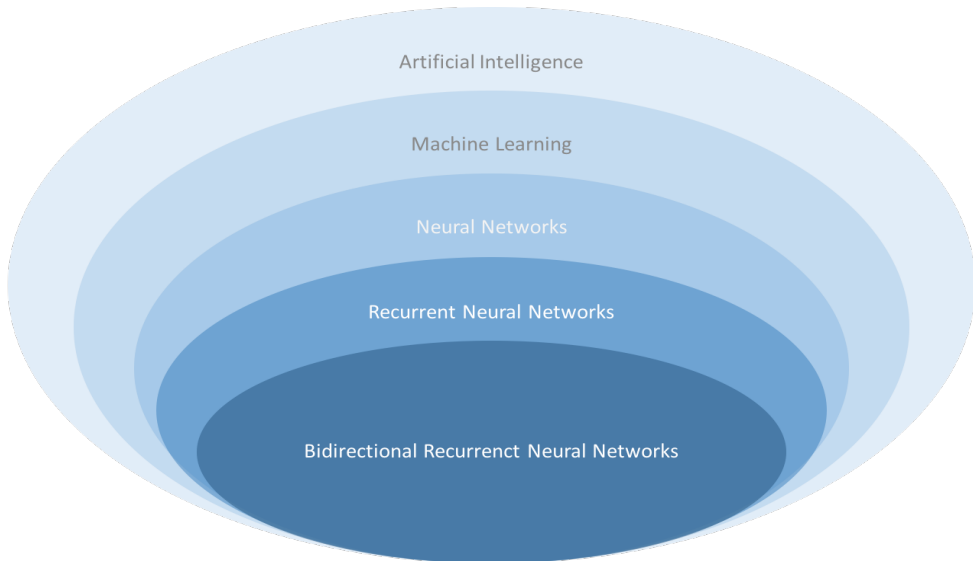
What makes current OCR software so successful with respect even to difficult recognition tasks (diacritics, non-standard orthography, relatively small amounts of training data and so on) is its use of forms of Artificial Intelligence. The technical details are complex and outdate quickly – basically, with a new generation of OCR every couple of years. Therefore, it is not the aim of this section to give a comprehensive description

²⁶ In order to ensure future exchangeability of the data sets, the Private Use Area, which is provided by the Unicode standard for the individualization of character sets, was not used, cf. also Schroeder et al. 2020.

²⁷ Schroeder et al. 2019.

but only to name some basic principles. Fig. 4 shows how the concepts briefly discussed below are interrelated, namely as elements in a matryoshka-like class inclusion.

FIG. 4. CONCEPTS OF ARTIFICIAL INTELLIGENCE IMPORTANT FOR OCR



Artificial Intelligence (AI) refers to systems that are not fully determined by their developers but learn by taking data into account. Narrow AI – other than general AI – focuses on specific tasks. Machine Learning (ML) techniques that achieve impressive results in recognition tasks (such as OCR) are part of narrow AI. Training an OCR model is a supervised (Machine) Learning task. Supervision refers to the fact that the computer is given Ground Truth, i.e. a man-made correct output that serves as reference for the expected output of the computer. The machine learns by means of (Artificial) Neural Networks. The term Neural Network (NN) is a metaphor in which the source domain is biology (neuron = nerve cell, neural network = network of neurons) and the target domain is computer science (Machine Learning). The metaphor relies on an input–activation–output analogy. Put simply, in OCR the input consists of black and white pixels, and the output is a Unicode code point (a Unicode character). Between input and output nodes (artificial neurons) is one or several so-called hidden layers (also artificial neurons). The activations of the individual neurons determine if and how information is forwarded to the next layer of neurons and ultimately to the output neurons. The network learns by comparing its output against the expected output (Ground Truth) and by feeding the result of the comparison back into the network (backpropagation). Out of the different types

of Neural Networks, Recurrent Neural Networks (RNN) are particularly important for OCR, more specifically, Bidirectional Recurrent Neural Networks (BRNN) with Long Short-Term Memory architecture. BRNNs do not only consider one individual input character at a time, but also take the context into account, i.e. the adjacent characters, both to the left and to the right (bidirectionality).²⁸ In this sense, it learns a kind of language model, in the form of the probability distributions of how characters follow each other. Its Long Short-Term Memory then makes it possible to look back over a larger sequence of characters, which is why Robertson says about OCR software making use of these techniques: “Strictly speaking, then, they do not perform optical *character* recognition, but rather optical *line* recognition: the context of a character in its line becomes pertinent information.”²⁹

3. OCR TOOLS (SOFTWARE AND HARDWARE)

There are a number of tools (i.e. software) that have successfully been tested for OCR of Coptic texts. Those working on OCR for Coptic texts have generally followed suggestions made by the CIS OCR group, using ScanTailor and OCRopus.³⁰ In the time since this group first proposed a workflow and tool set that proved useful when tested for Coptic, there have been some updates. In particular, the *Kallimachos* Centre of the University of Würzburg,³¹ partially building on existing work, have refined tools or developed new ones with respect to Layout Analysis (Larex), performance of text recognition (Calamari) and usability. Their tools and suggestions have successfully been adapted and used for Coptic by the *Coptic OCR* project. Their latest release, OCR4all, “also explicitly *focuses users with no technical background* and combines different tools in one consistent interface.”³² This is a significant step forward in the user experience, considering that before none of the trainable OCR software offered a “turn-key” solution.³³ None of them even had a Graphical User Interface.

It should also be added that, since the developers usually code on Linux, the tools presented in the following also run best on Linux. Most of them should also be

²⁸ Modern OCR software such as OCRopus uses Bidirectional Recurrent Neural Networks (BRNN) with Long Short-Term Memory (LSTM) architecture. For technical details, see Schuster – Paliwal 1997 for BRNN in general, Hochreiter – Schmidhuber 1997 for LSTM in general; Graves et al. 2008 for the application of BRNN and LSTM in text recognition; Breuel et al. 2013 for the implementation of these techniques in OCRopus.

²⁹ Robertson 2019: 123.

³⁰ Centrum für Informations- und Sprachverarbeitung, Ludwig-Maximilians-Universität München <<https://www.cis.uni-muenchen.de/dighum/cisocrgroup/index.html>> [accessed: 7/14/2020]. The workflow and tools used by the CIS OCR group are best described in Springmann 2015. As for applications to Coptic, cf. Lincke et al. 2019; Miyagawa et al. 2019.

³¹ Zentrum für Philologie und Digitalität “Kallimachos”, Julius-Maximilians-Universität Würzburg <<https://www.uni-wuerzburg.de/zpd/startseite/>> [accessed: 7/14/2020].

³² <<https://www.uni-wuerzburg.de/en/zpd/ocr4all/>> [accessed: 7/7/2020]; cf. Reul et al. 2019 and Section 3.5.










³³ See below, Section 3.4.

compatible with Windows and/or Mac but this often requires adaptation during the installation or even compilation from source code.

The software chosen by the *Coptic OCR* project and its predecessors are by no means the only options. Other OCR projects in Ancient Studies as well as in other disciplines have found other solutions and adapted the workflows to their needs. Ancient Greek might be case with characteristics similar to those of Coptic (in particular: diacritics, philological markup). Information on the OCR implementation for ancient Greek can be found in a current paper of Robertson and Boschetti.³⁴

While most of the software that will be presented in the following are open source and available at no charge, the requirements for hardware are not equally low. Indeed, hardware requirements of modern OCR software can exceed those of a simple office PC. This is particularly true for the Machine Learning taking place during model training and to a lesser extent also for the recognition process. Since these requirements mostly pertain to the architecture of the Graphical Processing Unit (GPU, graphics card), a state-of-the-art gaming PC is well equipped for the task, including for model training. Since information on hardware requirements outdates quickly, it should be researched as the need arises. For occasional use or testing, instead of acquiring hardware, server capacities can be rented or provided by the data processing service center of a research institution.

In the following overview of tools, a set of icons will be used as visual clues pertaining to workflow and user-friendliness:

- workflow phases (preprocessing, prediction, postprocessing):  (all phases),  (pre- and postprocessing),  (preprocessing),  (preprocessing and prediction),  (prediction),  (postprocessing)
- Graphical User Interface (GUI): 
- documentation (manual, tutorial, Wiki): 
- material for Coptic available: 

Footnotes to each tool refer to a publication by its developers, the respective project website as well as the data repository (usually hosted on GitHub).

3.1. ScanTailor

ScanTailor was developed by Joseph Artsimovich and is currently maintained by Nate Craun.³⁵ It is a software facilitating image preparation. There are several

³⁴ Robertson – Boschetti 2017.

³⁵ ScanTailor homepage <<http://scantailor.org>> (doesn't seem to be updated anymore); ScanTailor code repository on GitHub <<https://github.com/scantailor/scantailor>>; for tutorials, see the ScanTailor Wiki <<https://github.com/scantailor/scantailor/wiki>>; [all accessed: 7/14/2020].

independent development branches forked from ScanTailor’s source code which may continue to be maintained, even if the original ScanTailor is no longer continued.

While image preparation in general can be done with ordinary image processing software like Photoshop or Gimp (to name just a few), ScanTailor has been designed with a focus on image preparation specifically for OCR and contains only operations that are useful for this purpose. And it allows for batch processing, which means the execution of an operation for a part of or all image files loaded. The functions of ScanTailor include: rotation, deskewing, splitting, despeckling, dewarping, content selection (selecting text to be OCRed, i.e. deselecting translation, critical apparatus, margins etc.), adjustment of the line thickness (increasing or decreasing the line strength of characters in the text), binarization (conversion into a binary image that only consists of black and white pixels), adjustment of the output resolution.

3.2. Larex »»

Larex (Layout Analysis and Region EXtraction) is a development of members of the above mentioned *Kallimachos* Centre. It is designed for different tasks in relation to OCR of Early Modern books.³⁶ In particular, its “segmentation” module allows for a refined segmentation of pages with a complex layout into functional regions, e.g. paragraph of running text, image, heading, marginalia, page numbers and others. Also, the areas on a page to be OCRed can be selected while the rest of the page (e.g. images or footnotes) will be disregarded in the recognition process.

The “text” module can be used as a tool for transcription, i.e. in the production of training data (Ground Truth) or in the postprocessing phase when OCR results need to be corrected.

3.3. Calamari »»

At present, Calamari is the best OCR software for which models trained on Coptic texts are available.³⁷ It has been developed in the same lab as Larex and integrates the latest developments of Neural Networks for language recognition tasks.³⁸ The integration of techniques like cross-fold-training and confidence voting (cf. Section 2.6) make its predictions better than those of other trainable OCR software. Another advantage of Calamari – although really only relevant for model training

³⁶ Reul et al. 2017a; Reul et al. 2017b; <<http://www.is.informatik.uni-wuerzburg.de/open-source-tools/larex/>>; <<https://github.com/OCR4all/LAREX>> [accessed: 7/14/2020].

³⁷ See footnote 6.

³⁸ Wick et al. 2020.

– is its exploitation of the Graphical Processor Unit (GPU) which makes training significantly faster.³⁹ Calamari doesn't integrate preprocessing but is designed in a way to use files preprocessed by means of OCRopus.

3.4. OCRopus (OCRopy) »» ♀

OCRopus, later also called OCRopy, is an OCR software developed by Thomas M. Breuel.⁴⁰ Integrating Neural Nets with LSTM architecture, it is an open source trainable software that produces results that can compete with the best proprietary OCR software, ABBYY FineReader. Since ABBYY FineReader is not available for training on new scripts, OCRopus became the leading software for OCR of minority scripts and historical typeset. As stated in its documentation, “OCRopus is a collection of document analysis programs, not a turn-key OCR system”.⁴¹ While its modular approach to the OCR workflow is transparent and keeps each individual step simple, an average computer user might struggle with the lack of a Graphical User Interface (GUI); OCRopus needs to be accessed and controlled via the command line. Font-specific OCRopus models for Coptic are available from the *Coptic OCR* project.⁴²

3.5. OCR4all »» ♂ 🖥️ 📄

This suite of tools comes closest to the usability of a commercial OCR program for end-users. It is not an OCR program per se but provides a frontend (with a Graphical User Interface) to state-of-the-art OCR technology: Larex for preprocessing and Calamari for model training or recognition.⁴³ In OCR4all, the user can load their own Calamari models (e.g. those created by the *Coptic OCR* project as described in Sections 2.6 and 2.7).⁴⁴ The only obstacle one has to overcome is the installation process which requires a setup in Docker or in a Virtual (Linux) machine (with VirtualBox). Both create a virtual environment in which the OCR4all frontend and the underlying programs run smoothly without interfering with other installations on the system. The developers recommend a setup with VirtualBox for non-technical

³⁹ In my own tests, training Calamari was done in a quarter of the time that the OCRopus training process would take, roughly speaking 3.5–6 hours for Calamari vs. 18–24 hours for OCRopus with identical training data.

⁴⁰ Breuel 2008; <<https://github.com/tmbarchive/ocropy>> [accessed: 7/14/2020].

⁴¹ See the Readme.md file on <<https://github.com/tmbarchive/ocropy>> [accessed: 7/14/2020].

⁴² GitLab repository of *Coptic OCR*, DOI: 21.11101/0000-0007-C9D1-A; PID: 21.11101/0000-0007-C9D1-A.

⁴³ Reul et al. 2019; a short German presentation of the features of OCR4all can also be found in Wehner et al. 2020; <<https://www.uni-wuerzburg.de/en/zpd/ocr4all>>; <<https://github.com/OCR4all/OCR4all#ocr4all>> [both accessed: 7/10/2020].

⁴⁴ See footnote 6.

users. Once the setup is complete and OCR4all is started, it runs conveniently in a browser window. OCR4all is well documented in comprehensive setup guides and user manuals in English and German.

3.6. Other tools

The first fruitful trials for automated recognition of Coptic texts have been carried out by Moheb S. Mekhaïel in the 2000s.⁴⁵ Mekhaïel used a software called *Tesseract* (📄), which is a trainable OCR software developed by Hewlett-Packard since 1985. In 2005, HP released the source code for Tesseract, and the further development of the program has since been driven by Google.⁴⁶ At that time, Tesseract was the best trainable OCR software available. When other developers started producing OCR software based on Neural Networks, Coptic OCR projects switched to these new tools and didn't consider Tesseract any further.⁴⁷ In October 2018, Tesseract 4.0 was released, which integrates Neural Networks with LSTM, but it has not yet been tried with Coptic and no up-to-date training data and models are available for Tesseract 4.0 at present. One advantage of Tesseract is that there are several user/3rd party projects that develop and maintain Graphical User Interfaces that make interaction with Tesseract much easier, especially for non-technical users.⁴⁸

A fork (a kind of spin-off) from OCRopus (described above), called *Kraken* (👉, 📄), was developed by Benjamin Kiessling with a focus on historical and non-Latin scripts/fonts.⁴⁹ As, to the best of our knowledge, it has not yet been tested with Coptic material, it will not be discussed here any further.

The platform *Transkribus*⁵⁰ (👉, 📄, 📄), an outcome of *tranScriptorium*, funded by the EU's Seventh Framework Program and the Horizon2020 project *Recognition & Enrichment of Archival Documents* (READ), differs in several respects from the other tools presented in this section. Transkribus is designed for Handwritten Text Recognition (HTR) but can be used for OCR as well. Transkribus trains with ABBYY FineReader 11, the leading commercial text recognition software. Although it has not been trained on Coptic, recognition results are likely to be good. Unlike all other tools

⁴⁵ There is no published documentation of Moheb Mekhaïel's work other than the contents of his website; <<http://moheb.de/ocr.html>> [accessed: 7/14/2020].

⁴⁶ <<https://github.com/tesseract-ocr>> [accessed: 7/14/2020].



⁴⁷ For a comparison of Moheb Mekhaïel's Tesseract models and OCRopus models trained on Coptic, cf. Miyagawa et al. 2019. Miyagawa et al. 2017 report that "Tesseract with Mekhaïel's models had difficulty with typeset Coptic texts that contain diacritics, punctuation, and editorial signs."



⁴⁸ See the Tesseract documentation for user projects under <<https://tesseract-ocr.github.io/tessdoc/User-Projects---3rdParty>> [accessed: 7/15/2020].

⁴⁹ <<http://kraken.re>>; <<https://github.com/mittagessen/kraken>> [both accessed: 7/14/2020].

⁵⁰ Kahle et al. 2017; <<https://transkribus.eu/Transkribus>>; <<https://github.com/transkribus>> [both accessed: 7/15/2020].

mentioned in this paper, with Transkribus the data is processed on high performance servers and not on the user's computer. Transkribus has an excellent interface (GUI), an exhaustive documentation (Wiki, manuals, video tutorials) and a newsletter, as well as a contactable support team that organizes workshops and user conferences. However, since models are trained on servers by the Transkribus team, the user does not have the same level of control over the training process or its outcome as with locally conducted model training. After funding for the READ project ended in June 2019, Transkribus is as of now financed by a European Cooperative Society and several digitization projects so that it remains free of charge for the time being.⁵¹ As it has a 5-figure number of users, chances are good that Transkribus will be further developed and maintained in the future.

The *Aletheia* Document Analysis System (»», , ) by the Pattern Recognition & Image Analysis Research Lab (PRImA) at the University of Salford is another example of a platform with great usability, interface and documentation.⁵² The Pro edition, which is not free of charge, includes the possibility to train OCR models for Tesseract 3 and 4.

PoCoTo, the Postcorrection Tool (»», , ) is a program for the correction of OCR output. It was developed at the CIS in Munich (for which see footnote 29).⁵³ OCR output files (e.g., in hOCR format) serve as an input and a line-by-line view of original image and OCR output helps to detect and correct OCR errors. Furthermore, a digital dictionary, if imported, can flag candidates for wrong OCR predictions. Unfortunately, PoCoTo does not seem to be maintained anymore.

4. DIGITIZATION OF COPTIC TEXTS

4.1. Digitized Coptic texts

As of now, only a minority of Coptic texts is available in electronic form. Recently, So Miyagawa et al. compiled a list of resources for Coptic texts that are digitally available, and Caroline T. Schroeder has published a comprehensive description,

⁵¹ Slides of a talk by Günther Mühlberger: „Transkribus. Eine Forschungsplattform für die automatisierte Digitalisierung, Erkennung und Suche in historischen Dokumenten“, May 2019, <<https://de.slideshare.net/ETH-Bibliothek/transkribus-eine-forschungsplattform-fr-die-automatisierte-digitalisierung-erkennung-und-suche-in-historischen-dokumenten>>, slides no. 57 and 61 [accessed: 7/15/2020].

⁵² Clausner et al. 2020; <<http://www.primaresearch.org/tools/Aletheia>> [accessed: 7/15/2020]. While *Aletheia* is not on GitHub, source code of related tools, especially for the PAGE XML standard, developed by PRImA are available in their repository: <<https://github.com/PRImA-Research-Lab>> [accessed: 7/15/2020], however, mostly without documentation.

⁵³ Vobl et al. 2014; <<http://ocr.cis.uni-muenchen.de/>>; <<https://github.com/cisocrgroup/PoCoTo>> [both accessed: 7/15/2020].

discussion and history of Coptic digital textual resources.⁵⁴ With respect to non-canonical Coptic texts, one could add that the so-called Nag Hammadi library is not only available on CD (CD-ROM #7 “Greek Documentary”, published by the Packard Humanities Institute) but that the editions have been digitized and put online in Unicode format as part of Brill’s reference works.⁵⁵ Readers who are looking for already available digital Coptic texts can refer to the two above mentioned contributions, which are up to date and give convenient overviews of websites and other resources that provide access to digitized Coptic texts.

4.2. Objectives of digitization

An increasing number of research questions and research designs (collection and processing of source materials) require scholars to digitize texts. At present, digital Coptic texts are used, for instance, as base texts for the edition of the *Coptic Old Testament*,⁵⁶ as input for the Natural Language Processing pipeline of *Coptic Scriptorium*,⁵⁷ in the corpus of Coptic Scriptorium providing the attestations for lemmata in the *Coptic Dictionary Online* (CDO)⁵⁸ and as references for the *Database and Dictionary of Greek Loanwords in Coptic* (DDGLC).⁵⁹ Digital texts are also the basis for the application of data visualization techniques (e.g. word clouds, a vast variety of plots, network visualization).⁶⁰ Furthermore, they enable full-text search, re-use and dissemination, and, if shared publicly, easy access for communities interested in Coptic cultural heritage (e.g. researchers from different disciplines such as Coptology, Egyptology, Theology, Religious Studies, Linguistics etc., and, of course, the Coptic community). Furthermore, the possibility of digitizing texts themselves through OCR enables individual scholars to obtain a digital version of the texts that constitute an interesting corpus for their research, and, in this sense, OCR makes them independent of already digitally available texts as well as of the progress of digitization projects and the status of their data publication.

⁵⁴ Miyagawa et al. 2019; Schroeder 2019.

⁵⁵ Robinson 2000.

⁵⁶ For an overview of the project *Digital Edition of the Coptic Old Testament*, see Behlmer – Feder 2017 and the project’s website <<http://coptot.manuscriptroom.com>> [accessed: 7/14/2020].

⁵⁷ Schroeder – Zeldes 2016; Schroeder – Zeldes 2020; <<https://corpling.uis.georgetown.edu/coptic-nlp/>> [accessed: 7/14/2020].

⁵⁸ *Coptic Dictionary Online*, ed. by the Koptische/Coptic Electronic Language and Literature International Alliance (KELLIA), <<https://coptic-dictionary.org/>>. The corpus is also accessible directly through the ANNIS database: <<https://corpling.uis.georgetown.edu/annis/scriptorium>> [both accessed: 7/14/2020].

⁵⁹ <<https://www.geschkult.fu-berlin.de/en/e/ddglc/index.html>> [accessed: 7/14/2020].

⁶⁰ For a recent application in Coptic studies, see the set of entity visualizations that Amir Zeldes, Caroline T. Schroeder and Lance Martin have published for the Entity Recognition results of a subset on the Coptic Scriptorium corpus: <<https://copticcriptorium.org/entities/breakdown.html>> [accessed: 7/15/2020]; Zeldes et al. 2020.

4.3. Cost-benefit ratio of digitization

In manual text digitization, there are several possible workflows, depending on the number of texts to be digitized and the resources available. All of them involve text acquisition as a first step, be it in printed or digital form. The central step consists of the typing the text. In a process called double-keying, the latter is done twice, i.e., by two people independently. In a third step, the two versions are matched, and the errors are filtered out. This method has been recognized by funding agencies and, according to digitization guidelines of the DFG,⁶¹ transcriptions produced by means of this method are 99.997% accurate.⁶² Another possibility is to transcribe the text only once and then correct errors in the transcript (e.g., letter confusion, typos, skipped words, phrase or lines). This should be done by a second person or with a certain delay in order to minimize the effect of overlooking one's own errors. However, the accuracy rate is not as high as with the double-keying method.

Keying text, be it in a double-keying or other workflow, is the main and most time-consuming task. Miyagawa et al. have measured the time needed for manual transcriptions of Coptic texts (33 lines of text, 10 min) and for their proofreading (10 min).⁶³ My own count for keying one page of Coptic text containing 34 lines is 11 minutes. A computer can do that significantly faster; OCR programs extract the same amount of text in well under a minute. These data are somewhat impressionistic, as they were collected with few test persons and small amounts of data. A larger-scale study has not yet been done for Coptic, but it helps to look at studies from other digitization projects (of historical printings) to see the potential. Reul et al. have measured the time that each of four users takes to preprocess and OCR 50 pages of an incunabulum. The users needed 105–142 minutes for the task that included a mark-up of the complex layout of the book (layout analysis).⁶⁴ The average time spent on one page is of 2.1–2.85 minutes. In the *Coptic OCR* project, the preprocessing of the image files in ScanTailor (see Section 3.1), another way to prepare images for OCR, took 2 minutes per page, and this included a rather careful process of noise removal that may not always be necessary. It remains to be tested if and to what extent OCR results for Coptic texts downgrade if less time is invested in preprocessing.

There are several possibilities for adapting digitization workflows by integrating OCR. One suggestion is to proceed as with double-keying but to use two OCR output files instead of two manual transcriptions. Another option is to use only one OCR output and match it against an already existing manual transcription. A third

⁶¹ DFG = *Deutsche Forschungsgemeinschaft* (German Research Foundation), the most important organization for research funding in German academia.

⁶² DFG 2009: 11. For transcriptions of German historical books, this number is confirmed by an evaluation study, cf. Haaf et al. 2013.

⁶³ Miyagawa et al. 2019: i138 and Fig. 4.

⁶⁴ Reul et al. 2017b.

possibility, which is implemented in the OCR workflow and tools set that were presented in Sections 2 and 3, is to produce an OCR output and correct it manually in an editor facilitating this process.

Thus, OCR can help to save resources or compensate a lack of resources to a certain extent and enable the digitization of more text. But what applies to all automation and computer-aided work in the humanities also applies here. It will help researchers to make what they want to do more efficient. It could even enable us to address research questions that require data volumes that cannot be obtained manually. But it will not lead to a redundancy of manpower in research or a reduced workload for the people involved. Tito Orlandi, a pioneer in the digitization of Coptic texts and in the Digital Humanities (or Humanities Computing as his field of interest was called at the time), spoke of a “labour-saving myth”:

*The labour-saving myth: We know this myth to be silly; we know that only the dull, unimaginative scholar would not be inclined to do a better job with the time liberated from mechanical. We also know that the computer does not so much save labour as change the nature as well as scope of what we labour at.*⁶⁵

Coptic OCR greatly benefits from larger-scale projects which develop OCR tools that comply with the requirements of Coptic, especially projects that work on OCR for historical books. This has the further advantage that such tools might also be useful for non-OCR purposes like the encoding of layout information and its xml export in compliance with standards of (meta)data encoding. In the future, the now fast-developing HTR tools (with a focus on the digitization of handwritten manuscripts and on non-literary documents) will certainly give new inputs for the digitization of Coptic textual material.

5. COMMENTED BIBLIOGRAPHY

Chaudhuri et al. 2017. In Chapter 2 of their book on Optical Character Recognition, the authors describe the typical OCR workflow. Later chapters deal with OCR for particular scripts (English, French, German, Latin, Hindi, Gujrati).

Lincke et al. 2019 (open access); Miyagawa et al. 2017; Miyagawa et al. 2019 (open access). These three papers deal specifically with the training of OCR models for Coptic (typeset, fonts). At present, these are the only contributions published on the topic.

Neudecker 2019. Those who read German find an accessible and richly illustrated description and explanation of OCR with state-of-the-art tools, including Calamari, in this post on a blog of the Berlin State Library (Prussian Cultural Heritage).

⁶⁵ Orlandi 2002.

- Piotrowski 2012 (open access). Chapter 4 of Piotrowski's introduction to NLP for heritage texts discusses text acquisition including OCR, but also manual entry. As far as ancient languages are concerned, Greek serves as example. From the tools relevant for Coptic OCR only OCRopus was available at the time of writing and is referred to in the context of Ancient Greek OCR.
- Rehbein 2017. This chapter from a recent introduction to the Digital Humanities (written in German), deals with digitization. Conveniently, the author not only gives an overview of text digitization but also of image digitization which, of course, is a prerequisite of OCR. The descriptions are concise, without going into technical details, a good first step.
- Robertson 2019 (open access). Bruce Robertson, a leading expert in OCR for (polytonic) Ancient Greek and Latin, has recently written an introductory chapter on OCR for Classical Philology. Although the tools used for Ancient Greek OCR are others than the ones used for Coptic, most of what he says about the possibilities and challenges as well as about the workflow and general principles can be transferred to Coptic. And the paper also discusses the recent developments in OCR including the relevant tools used for Coptic OCR (OCRopus, Calamari).
- Rohrer 2017. If one wishes to understand how Recurrent Neural Networks and Long Short-Term Memory work, the best introduction is not in a book but on Youtube. In the video, Brandon Rohrer demonstrates and visualizes the basics of RNN and LSTM by means of two simple questions: "What's for dinner?" and "What is the next word if we were to write a children's book?"

6. REFERENCES

- Balestri, G. – Hyvernat, H. (1924). *Acta Martyrum, Vol. II. Corpus Scriptorum Christianorum Orientalium* 86, Scriptorum Coptici 6. Paris.
- Behlmer, H. – Feder, F. (2017). "The Complete Digital Edition and Translation of the Coptic Sahidic Old Testament". *Early Christianity* 8: 97–107 (doi: 10.1628/186870317X14876711440169).
- Breuel, Th.M. (2008). "The OCRopus open source OCR system". In B.A. Yanikoglu – K. Berkner (eds.), *Document Recognition and Retrieval XV, 29–31 January 2008, San Jose, California, USA*, Proceeding of SPIE 6815: 120–134 (doi: 10.1117/12.783598).
- Breuel, Th.M. – Ul-Hasan, A. – Al Azawi M.A. – Shafait, F. (2013). "High-Performance OCR for Printed English and Fraktur Using LSTM Networks". In *ICDAR 2013. 12th International Conference on Document Analysis and Recognition, 25–28 August 2013, Washington, DC*, Washington: 683–687 (doi 10.1109/ICDAR.2013.140).
- Chaudhuri, A. – Mandaviya, K. – Badelia, P. – Ghosh, S.K. (2017). *Optical character recognition systems for different languages with soft computing*. Studies in

- fuzziness and soft computing 352, Cham (doi: 10.1007/978-3-319-50252-6).
- Clausner, Ch. – Antonacopoulos, A. – Pletschacher, S. (2020). “Efficient and effective OCR engine training”. *International Journal on Document Analysis and Recognition* 23: 73–88 (doi: 10.1007/s10032-019-00347-8).
- Depuydt, L. (1991). *Homiletica from the Pierpont Morgan Library. Seven Coptic Homilies Attributed to Basil the Great, John Chrysostom, and Euodius of Rome*. Corpus Scriptorum Christianorum Orientalium 524, Scriptorum Coptici 43. Leuven.
- Deutsche Forschungsgemeinschaft (ed.) (2009). *Scientific Library Services and Information Systems (LIS): DFG Practical Guidelines on Digitisation*. Bonn. <https://www.dfg.de/download/pdf/foerderung/programme/lis/praxisregeln_digitalisierung_en.pdf> [accessed: 7/14/2020].
- Graves, A. – Liwicki, M. – Fernández, S. – Bertolami, R. – Bunke, H. – Schmidhuber, J. (2009). “A Novel Connectionist System for Unconstrained Handwriting Recognition”. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 31: 855–868 (doi: 10.1109/TPAMI.2008.137).
- Haaf, S. – Wiegand, F. – Geyken, A. (2013). “Measuring the Correctness of Double-Keying: Error Classification and Quality Control in a Large Corpus of TEI-Annotated Historical Text”. *Journal of the Text Encoding Initiative* 4 (doi: 10.4000/jtei.739).
- Hochreiter, S. – Schmidhuber, J. (1997). “Long Short-Term Memory”. *Neural Computation* 9: 1735–1780 (doi: 10.1162/neco.1997.9.8.1735).
- Johnson, D.W. (1980). *A Panegyric on Macarius Bishop of Tkôw Attributed to Dioscorus of Alexandria*. Corpus Scriptorum Christianorum Orientalium 415, Scriptorum Coptici 41. Leuven.
- Kahle, Ph. – Colutto, S. – Hackl, G. – Mühlberger, G. (2017). “Transkribus – A Service Platform for Transcription, Recognition and Retrieval of Historical Documents”. In *ICDAR 2017. 14th IAPR International Conference on Document Analysis and Recognition, 9–15 November 2017, Kyoto, Japan*: vol. 4, 19–24 (doi: 10.1109/ICDAR.2017.307).
- Kasser, R. (1991). “Alphabets, Coptic”. In A.S. Atiya (ed.), *The Coptic Encyclopedia*, New York: vol. 8, 32–41. <<http://cdl.libraries.claremont.edu/cdm/ref/collection/cce/id/2008>> [accessed: 7/15/2020].
- Kasser, R. (1958). *Papyrus Bodmer III. Evangile de Jean et Genèse I-IV,2 en bohairique*. Corpus Scriptorum Christianorum Orientalium 177, Scriptorum Coptici 25. Leuven.
- Kuhn, K.H. (1960). *Pseudo-Shenoute on Christian Behaviour*. Corpus Scriptorum Christianorum Orientalium. 206, Scriptorum Coptici 29. Leuven.
- Lefort, L.Th. (1933). *S. Pachomii Vitae Sahidice Scriptae*. Corpus Scriptorum Christianorum Orientalium 99, Scriptorum Coptici 9. Paris.
- Lincke, E.-S. (2016). “Optical Character Recognition (OCR) for Coptic. Testing Automated Digitization of Texts with OCRopy”. Presentation at the 11th

- International Congress of Coptic Studies, August 25–30, 2016, Claremont, CA.*
- Lincke, E.-S. – Bulert, K. – Büchler, M. (2019). “Optical Character Recognition for Coptic fonts. A multi-source approach for scholarly editions”. In *Proceedings of the 3rd International Conference on Digital Access to Textual Cultural Heritage – DATeCH 2019*, ACM: 87–91 (doi: 10.1145/3322905.3322931).
- Miyagawa, S. – Bulert, K. – Büchler, M. (2017). “Utilization of Common OCR Tools for Typeset Coptic Texts”. Extended Abstract of the presentation at the *2nd International Conference on Digital Access to Textual Cultural Heritage – DATeCH 2017*. <https://www.academia.edu/33464206/Utilization_of_Common_OCR_Tools_for_Typeset_Coptic_Texts> [accessed: 7/15/2020].
- Miyagawa, S. – Bulert, K. – Büchler, M. – Behlmer, H. (2019). “Optical character recognition of typeset Coptic text with neural networks”. *Digital Scholarship in the Humanities* 34, Supplement 1 (doi: 10.1093/llc/fqz023, i135–i141).
- Neudecker, C. (2019). “Kuratieren mit KI: Erste Ergebnisse aus dem QURATOR-Projekt”. <<https://blog.sbb.berlin/zwischenenergebnisse-qrator-jahr-1/>> [accessed: 7/1/2020].
- Orlandi, T. (2002). “Is Humanities Computing a discipline?”. *Jahrbuch für Computerphilologie* 4: 51–58. <<http://computerphilologie.digital-humanities.de/jg02/orlandi.html>> [accessed: 7/13/2020].
- Piotrowski, M. (2012). *Natural Language Processing for Historical Texts*. Synthesis Lectures on Human Language Technologies 17. San Rafael (doi: 10.2200/S00436ED1V01Y201207HLT01).
- Pletschacher, S. – Antonacopoulos, A. (2010). “The PAGE (Page Analysis and Ground-Truth Elements) Format Framework”. In *Proceedings of the 20th International Conference on Pattern Recognition (ICPR2010), Istanbul, Turkey, August 23–26, 2010*, IEEE: 257–260 (doi: 10.1109/ICPR.2010.72).
- Rehbein, M. (2017). “Digitalisierung”. In F. Jannidis – H. Kohl – M. Rehbein (eds.), *Digital Humanities. Eine Einführung*, Stuttgart: 179–198.
- Reul, Ch. – Christ, D. – Hartelt, A. – Balbach, N. – Wehner, M. – Springmann, U. – Wick, Ch. – Grundig, Ch. – Büttner, A. – Puppe, F. (2019). “OCR4all—An Open-Source Tool Providing a (Semi-)Automatic OCR Workflow for Historical Printings”. *Applied Sciences* 9: No. 4853 (doi: 10.3390/app9224853).
- Reul, Ch. – Springmann, U. – Wick, Ch. – Puppe, F. (2018). “Improving OCR Accuracy on Early Printed Books by combining Pretraining, Voting, and Active Learning”. *Journal for Language Technology and Computational Linguistics* 33: 3–24, <https://jlcl.org/content/2-allissues/1-heft1-2018/jlcl_2018-1_1.pdf> [accessed: 7/14/2020].
- Reul, Ch. – Springmann, U. – Puppe, F. (2017a). “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 – DATeCH 2017*, ACM: 137–142 (doi: 10.1145/3078081.3078097).

- Reul, Ch. – Dittrich, M. – Gruner, M. (2017b). “Case Study of a Highly Automated Layout Analysis and OCR of an Incunabulum: ‘Der Heiligen Leben’ (1488)”. In *Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage – DATeCH 2017*, ACM: 155–160 (doi: 10.1145/3078081.3078098).
- Robertson, B. (2019). “Optical Character Recognition for Classical Philology”. In M. Berti (ed.), *Digital Classical Philology. Age of Access?* Grundfragen der Informationsgesellschaft 10, Berlin – Boston: 117–136 (doi: 10.1515/9783110599572-008).
- Robertson, B. – Boschetti, F. (2017). “Large-Scale Optical Character Recognition of Ancient Greek”. *Mouseion* 14: 341–359 (doi: 10.3138/mous.14.3-3).
- Robinson, J.M. (ed.). (2000). *The Coptic Gnostic Library (online) — A Complete Edition of the Nag Hammadi Codices* (digital version of the 5-vol. ed. Brill 2000). <<https://referenceworks.brillonline.com/browse/coptic-gnostic-library>> [accessed: 7/1/2020].
- Rohrer, B. (2017). “Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM)”. *Youtube*, uploaded 26 June 2017. <<https://youtu.be/WCUNPb-5EYI>> [accessed: 7/1/2020].
- Schroeder, C.T. (2020). “Cultural Heritage Preservation and Canon Formation: What Syriac and Coptic Can Teach Us about the Historiography of the Digital Humanities”. In G. Frank – A. Jacobs – S. Holman (eds.), *The Garb of Being. Embodiment and the Pursuit of Holiness in Late Ancient Christianity*, New York: 318–346 (doi: 10.1515/9780823287048-017).
- Schroeder, C.T. – Schmid, U. – Miyagawa, S. – Platte, E. – Zeldes, A. (2019). “KELLIA White Paper on Transcription and Encoding Standards for Digital Coptic”, 27 March 2019 <<https://kellia.uni-goettingen.de/downloads/KELLIA-transcription-white-paper.pdf>> [accessed: 7/1/2020].
- Schroeder, C.T. – Zeldes, A. (2016). “Raiders of the Lost Corpus”. *Digital Humanities Quarterly* 10 <<http://www.digitalhumanities.org/dhq/vol/10/2/000247/000247.html>> [accessed: 7/1/2020].
- Schroeder, C.T. – Zeldes, A. (2020). “A Collaborative Ecosystem for Digital Coptic Studies”. Special issue of the *Journal of Data Mining and Digital Humanities* on Collecting, Preserving, and Disseminating Endangered Cultural Heritage <<https://jdmhd.episciences.org/6797>> [accessed: 4/6/2020].
- Schuster, M. – Paliwal, K. (1997). “Bidirectional recurrent neural networks”. *IEEE Transactions on Signal Processing* 45: 2673–2681 (doi: 10.1109/78.650093).
- Sobhy, P.G. (1919). *Le martyre de Saint Hélias et l’encomium de l’évêque Stéphane de Hnès sur Saint Hélias*. Bibliothèque d’Etudes Coptes 1. Cairo.
- Springmann, U. (2015). *Ocrocis. A high accuracy OCR method to convert early printings into digital text. A Tutorial*. Munich. <<http://cistern.cis.lmu.de/ocrocis/tutorial.pdf>> [accessed: 7/14/2020].

- Thompson, H. (1924). *The Gospel of St. John According to the Earliest Coptic Manuscript*. Publications of the British School of Archaeology in Egypt 36. London.
- Vobl, Th. – Gotscharek, A. – Reffle, U. – Ringlstetter, Ch. – Schulz, K.U. (2014). “PoCoTo – an Open Source System for Efficient Interactive Postcorrection of OCRed Historical Texts”. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage – DATeCH 2014*, ACM: 57–61 (doi: 10.1145/2595188.2595197).
- Wick, Ch. – Reul, Ch. – Puppe, F. (2020). “Calamari – A High-Performance Tensorflow-based Deep Learning Package for Optical Character Recognition”. *Digital Humanities Quarterly* 14 <<http://www.digitalhumanities.org/dhq/vol/14/2/000451/000451.html>> [1/7/202].
- Wehner, M. – Dahnke, M. – Landes, F. – Nasarek, R. – Reul, Ch. (2020). “OCR4all – Eine semi-automatische Open-Source-Software für die OCR historischer Drucke”. In Ch. Schöch (ed.), *Dhd2020. Spielräume. Digital Humanities zwischen Modellierung und Interpretation. 7. Jahrestagung des Verbands Digital Humanities im deutschsprachigen Raum, Universität Paderborn 02. bis 06. März 2020*: 43–45 (doi: 10.5281/zenodo.3666689).
- Zeldes, A. – Schroeder, C. – Martin, L. (2020). “Exposing Coptic entities: automation, search and visualization”. In *Digital Coptic 3 (online conference, 12–13 July 2020)* <http://kellia.uni-goettingen.de/digitalcoptic3/slides/DC3_entities_2020.pdf> [accessed: 7/4/2020].