The TRANSCRIPT Tool for Historical Ciphers by the DECRYPT Project

Ferenc Szigeti
Technical University of Munich
ferenc.szigeti@tum.de

Mihály Héder[*]
Department for Philosophy
and History of Science Budapest
University of Technology and Economics
heder.mihaly@gtk.bme.hu

Abstract

TRANSCRIPT is a web-based tool[1] for creating transcriptions for scanned images of historical manuscripts. The tool is interactive and leverages pre-trained image processing algorithms, therefore implementing a human-in-the-loop artificial intelligence. Different algorithms may be used in every step of the process: line and symbol segmentation, clustering and symbol recognition. However, at each step the user can manually intervene, clean up or enrich the results of algorithmic image processing. We present here the current work-in-progress version of the TRANSCRIPT tool.

1 Introduction

Historical documents are an important part of our cultural heritage, and their proper transcription and indexing is the key to their preservation. Particularly interesting type of historical documents are historical ciphers (see for example Figure 2, 3 or 4), which contain secret messages or instructions related to diplomatic, military or religious matters, among others.

It is usually not an easy task to reveal the secret information hidden inside such manuscripts, thus an interdisciplinary collaboration — known as the DECRYPT project (Megyesi et al., 2020) — formed to address this challenge. This project aims to create an integrated, web-based pipeline of components which covers all aspects of research historians undertake when investigating historical ciphers.

This involves the collection and digitization of documents — currently performed in the DECODE database (Héder and Megyesi, 2022), which is followed by the transcription of the documents. Although, it is possible to manually transcribe these documents, facilitating the task with image recognition has high research potential.

Our contribution — the TRANSCRIPT tool was developed as part of the DECRYPT project to accelerate the transcription process by providing users access to image processing algorithms.

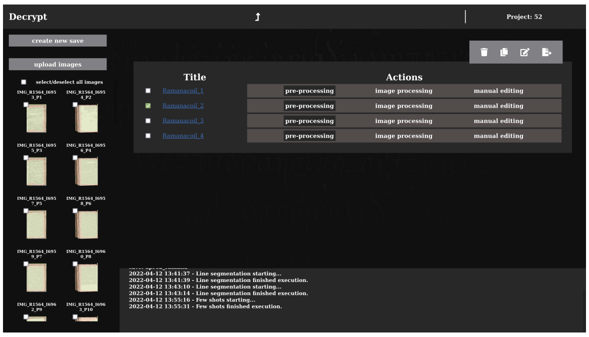
2 Related Work

Handwritten text recognition (HTR) methods can be applied to automate the segmentation and recognition of historical cipher symbols. This is a challenging task however, as these ciphers are written in different alphabets, thus training data is scarce for any specific a lphabet. Furthermore, touching symbols and the lack of punctuation introduce even more ambiguity. Recent advances nonetheless show that both unsupervised (Baró et al., 2019; Chen et al., 2021; Yin et al., 2019) and supervised (Renfei, 2020; Souibgui et al., 2021) methods can yield good enough results, so that integrated into an interactive tool, they could be a viable alternative to manual transcription.

Interactive transcription tools implement a semi-automated transcription process, which can mean an ad-hoc selection of generic tools (e.g. image, code or text editor), or a dedicated pipeline of methods. Such a pipeline is superior to any naive selection of tools, and it has already been demonstrated that ciphers of known alphabets can be efficiently transcribed even with the naive approach (Fornés et al., 2017; Magnifico, 2021). Unknown alphabets pose a challenge to supervised HTR methods, and interactive transcription tools are less usable as well in those cases as a consequence. Finally, promising results were shown by a dedicated interactive transcription tool (Chen et al., 2020) on digit ciphers (Johansson, 2019).

[*] Műegyetem rkp. 3., H-1111 Budapest, Hungary https://orcid.org/0000-0002-9979-9101

[1] A publicly available demo version can be accessed here: https://de-crypt.org/transcript-tool-demo



	Title	Actions		
□	Ramanacoil_1	pre-processing	image processing	manual editing
☑	Ramanacoil_2	pre-processing	image processing	manual editing
□	Ramanacoil_3	pre-processing	image processing	manual editing
□	Ramanacoil_4	pre-processing	image processing	manual editing

```
2022-04-12 13:41:37 - Line segmentation starting...
2022-04-12 13:41:39 - Line segmentation finished execution.
2022-04-12 13:43:10 - Line segmentation starting...
2022-04-12 13:43:14 - Line segmentation finished execution.
2022-04-12 13:55:16 - Few shots starting...
2022-04-12 13:55:31 - Few shots finished execution.
```

Figure 1: Project view with Ramanacoil ciphers.

3 Overview of the TRANSCRIPT Tool

The purpose of the TRANSCRIPT tool is to ease and speed-up the transcription process by integrating image processing methods into a single human-in-the-loop framework.

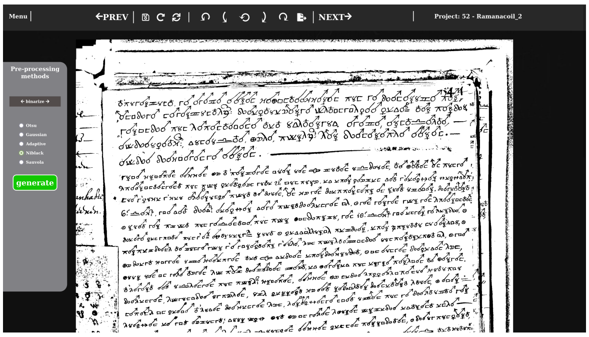
Users can choose from a selection of different image processing algorithms for the different tasks involved in the transcription process: cropping images, rotating images, binarization, line segmentation, symbol segmentation and clustering (Baró et al., 2019; Souibgui et al., 2021).

The TRANSCRIPT tool can be accessed from the DECODE database, but any work carried out in the TRANSCRIPT tool is sandboxed — i.e. the result will not be uploaded to the official record — enabling trial-and-error style experimentation. Once the user is satisfied w ith t he r esults, the plaintext transcription as well as other information (e.g. symbol positions on the images) can be exported out. All computation-heavy tasks take place on the server side, while the browser acts as a client.

After loading the TRANSCRIPT tool, the user lands in the ”project view” (Figure 1), where saved progresses can be resumed, deleted, copied or exported out. Furthermore, images can be uploaded into the project and new saves can be created by selecting any number of images already added to the project (either from the DECODE database or by directly uploading them).

4 Main Features

The main features of the TRANSCRIPT tool can be found in one of its three views: pre-processing (Figure 2), image processing (Figure 3) and post-processing (Figure 4).



○ Otsu
○ Gaussian
○ Adaptive
● Niblack
○ Sauvola

Figure 2: Pre-processing view on a Ramanacoil cipher binarized by the Niblack method.

4.1 Pre-Processing View

The pre-processing view (Figure 2) integrates the features necessary for preparing the images (of historical ciphers) for subsequent image processing steps. There are three features present in this view: cropping, rotation and binarization. The implementation of the tool, however, allows for an easy integration of new features, if the need would arise for them.

Cropping and rotation: these two features allow the user to rotate the image into the desired position and crop out the area of interest from it. This ensures that the images could be adjusted to make the text lines horizontal and to remove any unnecessary margin around the edges.

Binarization: this feature allows the user to binarize the image. This is a crucial pre-processing step, as most of the image processing algorithms work only well if their input image is binarized. The user can choose from five d ifferent binarization algorithms: Otsu, Gaussian, Adaptive, Niblack and Sauvola (Baró et al., 2019).

4.2 Image Processing View

The image processing view (Figure 3) integrates the features, which enable the user to utilize the different image processing algorithms. Note, that the user does not need to install, directly run or dedicate resources to these algorithms — these issues are abstracted away and taken care of by the TRANSCRIPT tool. The features part of this view are the following: box adjustment, line segmentation, symbol segmentation, Few-shot, clustering and label propagation.

Box adjustment: boxes can be moved around and resized with the cursor on the images, and they can also be added to or removed from any image. This allows the user to adjust the generated

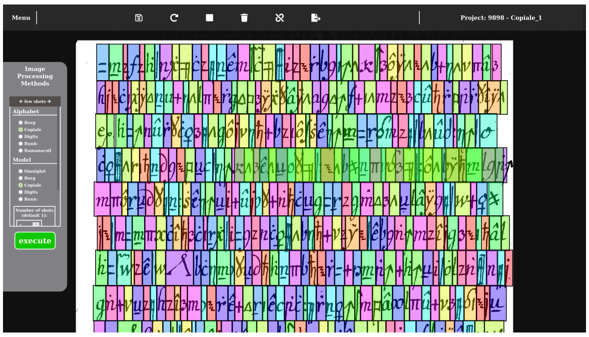


Figure 3: Image processing view on a Copiale cipher processed by the Few-shot method.

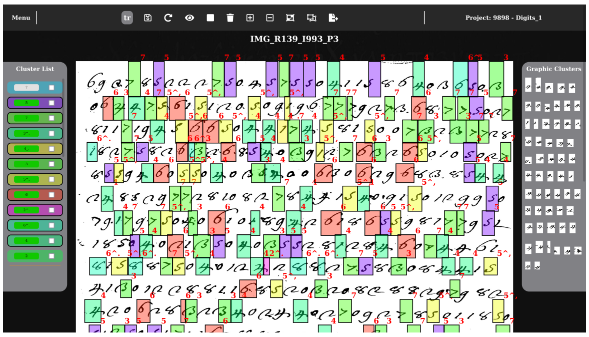


Figure 4: Post-processing view on a Digit cipher with some graphic clusters.

boxes (let them represent lines or symbols) whenever necessary.

Line segmentation: takes an empty image as input and produces boxes corresponding to lines on the image (Baró et al., 2019).

Symbol segmentation: takes an empty image as input and produces boxes corresponding to symbols on the image. (Baró et al., 2019).

Few-shot: takes line boxes as input and produces boxes and transcription corresponding to symbols inside the lines. (Souibgui et al., 2021).

Clustering: takes symbol boxes as input and produces clusters for those symbol boxes. The different clusters are indicated by different colors of the boxes (Baró et al., 2019).

Label propagation: takes clustered symbol boxes and improves on the clustering (Baró et al., 2019).

4.3 Post-Processing View

The post-processing view (Figure 4) integrates the features, which enable the user any kind of manual adjustment necessary to adjust the output of the image processing algorithms. This includes the following features: box adjustment, cluster adjustment, transcription assignment, graphic cluster view and exporting transcription.

Box adjustment: boxes — the same way as in the image processing view — can be moved around and resized according to the needs of the user.

Cluster cleaning: clusters — i.e. boxes of the same color — can be adjusted via assigning boxes to them or removing boxes from them. Furthermore, entire clusters can be added or removed, if necessary.

Transcription assignment: each cluster has a corresponding transcription (by default this is an empty string), but the user can enter any unicode string — adhering to the transcription guidelines (Megyesi and Tudor, 2021). This transcription is assigned then to each individual box inside the cluster.

Graphic cluster view: once the user selects a cluster on the cluster list (left sidebar), then the individual boxes inside the cluster are displayed on the graphic cluster view (right sidebar). In this view, the symbols in a given cluster can be inspected, and outliers can be easily identified.

Exporting transcription: if the user has cleaned all clusters and assigned to them the desired transcription, then this information — along with the positions and clusters of boxes — can be exported out.

5 Integration

In order for the TRANSCRIPT tool to be a success, it needs to work in concert with the other elements of the toolchain of the DECRYPT project. There are several key areas we need to concentrate on to achieve this.

First, the authentication and authorization process needs to be uniform across all tools, like linguistic analyzer, the key structure analyzer and the DECODE database itself. This part of the integration is already achieved.

In terms of underlying technology, the system needs to make use of GPU-based image processing in a way that users won't interfere with each other's experiments by locking up these non-shareable resources. In this area we expect a lot of changes until the end of DECRYPT project and beyond, as the technological stacks on top of GPU processing constantly improve and may even provide time-sharing in the future.

Finally, the data formats need to be harmonized across the system, in which we still have work to do.

6 Conclusion

This paper introduced the TRANSCRIPT tool, which is a web-based and human-in-the-loop image processing tool enabling the semi-automated transcription of scanned images of historical manuscripts based on server-side image processing algorithms. This tool is the part of the DECRYPT pipeline.

The tool could easily be extended and allows for the integration of different image processing algorithms. Future work includes primarily the integration of such algorithms and enhancing the workflow. Addressing this requirement is what distinguishes TRANSCRIPT tool from manual transcription tools: our solution needs to visualize whatever output the image processing algorithms yield and let the user fine-tune the results. This is highly different from the meticulous manual construction of transcriptions and requires process engineering and novel UX solutions, like incorporating user provided adjustments into subsequent runs of the algorithm to improve on the original results.

Another area of future work is the GPU scheduling and dependency minimization (some algorithms may only need GPU for training but not for running, etc.).

The TRANSCRIPT tool is currently in beta testing. For the address of the publicly available demo version see *footnote 1*.

Acknowledgments

This work has been supported by the Swedish Research Council, grant 2018-06074, DECRYPT — Decryption of Historical Manuscripts. We would like to thank our colleagues in the DECRYPT team, in particular Beáta Megyesi, Nikolina Milioni, Crina Tudor, George Lasry, Jialuo Chen, Mohamed Ali Souibgui and Alicia Fornés for their feedback on the design, both of the user experience and the integration with the machine learning algorithms.

References

Arnau Baró, Jialuo Chen, Alicia Fornés, and Beáta Megyesi. 2019. Towards a generic unsupervised method for transcription of encoded manuscripts. In *Proceedings of the 3rd International Conference on Digital Access to Textual Cultural Heritage*, pages 73–78. 1, 2, 3

Jialuo Chen, Mohamed Ali Souibgui, Alicia Fornés, and Beáta Megyesi. 2020. A web-based interactive transcription tool for encrypted manuscripts. In *the 3rd International Conference on Historical Cryptology*. 1

Jialuo Chen, Mohamed Ali Souibgui, Alicia Fornés, and Beáta Megyesi. 2021. Unsupervised alphabet matching in historical encrypted manuscript images. In *International Conference on Historical Cryptology*, pages 34–37. 1

Alicia Fornés, Beáta Megyesi, and Joan Mas. 2017. Transcription of encoded manuscripts with image processing techniques. In *DH*. 1

Mihály Héder and Beáta Megyesi. 2022. The decode database of historical ciphers and keys: Version 2. In *International Conference on Historical Cryptology*. 1

Kajsa Johansson. 2019. Transcription of historical encrypted manuscripts: Evaluation of an automatic interactive transcription tool. 1

Giacomo Magnifico. 2021. Lost in transcription: Evaluating clustering and few-shot learningfor transcription of historical ciphers. 1

Beáta Megyesi and Crina Tudor. 2021. Transcription of historical ciphers and keys: Guidelines, version 2.0. 3

Beáta Megyesi, Bernhard Esslinger, Alicia Fornés, Nils Kopal, Benedek Láng, George Lasry, Karl de Leeuw, Eva Pettersson, Arno Wacker, and Michelle Waldispühl. 2020. Decryption of historical manuscripts: the decrypt project. *Cryptologia*, 0(0):1–15. 1

Han Renfei. 2020. Using attention-based sequence-to-sequence neural networks for transcription of historical cipher documents. Master's thesis, Uppsala University. 1

Mohamed Ali Souibgui, Alicia Fornés, Yousri Kessentini, and Crina Tudor. 2021. A few-shot learning approach for historical ciphered manuscript recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5413–5420. IEEE. 1, 2, 3

Xusen Yin, Nada Aldarrab, Beáta Megyesi, and Kevin Knight. 2019. Decipherment of historical manuscript images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 78–85. IEEE. 1