ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

REPORT

Pushing the Limits of Optical Character Recognition on Complex Multilingual Documents - Recognition

EPFL SEMESTER PROJECT

Ziqi Zhao

10th June 2022

Abstract

Optical Character Recognition (OCR) has become a crucial and useful tool for digitizing past documents. Recent advance in OCR engines enable fast and accurate recognition for documents with multiple languages, but they still have non-negligible errors in some documents such as ancient Greek commentaries. In the context of the Ajax Multi-Commentary project, the goal is to analyze the styles of digitized ancient Greek commentaries. Consider that these digitized documents will be used for further NLP tasks (e.g. named entity recognition), errors will be accumulated through the process and a small OCR error will lead to significant performance degradation at the end of the task. Therefore, it is of great importance to further improve the performance of OCR engines. This report focus on the study of image pre-processing techniques and the OCR language models, and aims at improving the performance of OCR especially in ancient Greek languages. Through comprehensive experiments, we significantly reduce the Character Error Rate (CER) for two commentaries. A detailed analysis of experiments further provides insights for future research.

Chapter 1

Introduction

1.1 Context

Document digitization has gained increasing attention in the field of historical document analysis. By document digitization, researchers can automate the process of text extraction, utilize powerful Natural Language Processing (NLP) methods to conduct text analysis based on the extracted texts and highly improve the efficiency and diversity of historical document analysis. One commonly used techniques in document digitization is Optical Character Recognition (OCR), the process of recognizing texts from the scanned documents. A typical OCR process contains a series of steps that are shown in Figure 1.1, and is widely applied in modern OCR engines (Reul et al. 2019; Smith 2007; Wick, Reul and Puppe 2020). Given the scanned images, some image pre-processing methods will be applied, to binarize and deskew the image, as well as some denoising techniques to improve the visibility of texts. Then a segmentation process will detect the layout of the page, and segment it into several boxes. For each box, the OCR engine will recognize the containing texts based on the language models. Finally, the post-processing algorithm will check and correct the mistakes in the OCR output.

This project contributes to the Ajax Multi-Commentary project (Matteo, Sven and Bruce 2021) as the beginning step. Ajax Multi-Commentary project is a case study on the published commentaries (i.e. explanation or interpretation about the script) about *Ajax*, a Greek tragedy on the suicide of the Greek hero Ajax during the Trojan War. The goal of this project is to study the evolution of commentaries as a genre. As the first step, OCR will be conducted to the scanned images. Then the output texts can enable further useful applications, including the linking of the commentary section with corresponding texts, aligning different commentaries, and enriching the commentary sections by means of Natural Language Processing (NLP) algorithms.

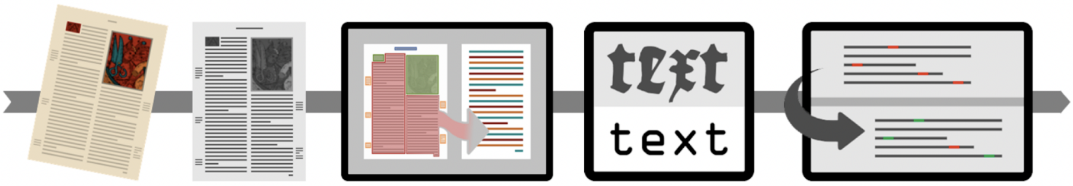


Figure 1.1
Typical workflow for OCR process shown by (Reul et al. 2019). Steps from left to right are: 1) get scanned page image; 2) pre-process the image; 3) conduct segmentation; 4) feed to the OCR engine; 5) post-process the OCR output.

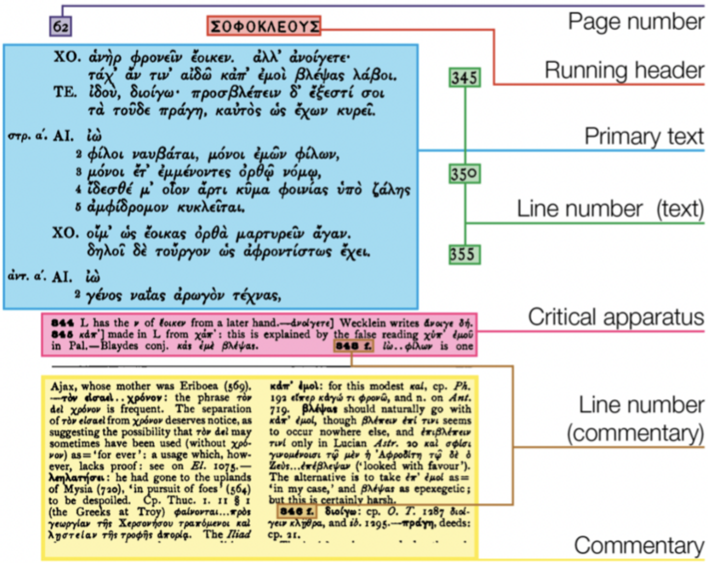


FIGURE 1.2
Typical layout for a ancient Greek Commentary in (Matteo, Sven and Bruce 2021).

1.2 MOTIVATION

Although OCR engines are featured as a fast and accurate tool for text recognition, their actual performance is still limited by the complicated scenarios in the real-world applications. In the context of Ajax Multi-Commentary project, the page layout are composed of multiple sections. Figure 1.2 gives an example of one page from the documents. The layout includes the page and line number, the header, primary text, critical apparatus and a double-column commentary. Besides the complex layout, the font in each section varies. In addition, texts in the commentary section are usually written in Greek and another language. They add difficulties to correctly segmenting the page and recognizing the correct language for texts by the OCR engines. Specifically, current OCR engines have unsatisfactory performance on Greek texts within the commentary. Furthermore, since the recognized texts are then fed to the other text analysis methods, the effect of OCR error will be enlarged to the state that significantly reduces the performance of analysis. To avoid such a problem, current OCR performance needs improvements to meet the requirement for further analyses.

1.3 GOALS

The goal of this project is to improve the performance of OCR recognition for ancient Greek commentaries. Specifically, this project will focus on: 1) finding image pre-processing techniques that are beneficial to the OCR engines; and 2) obtaining a Greek language model that can improve the accuracy of ancient Greek letter recognition.

CHAPTER 2

METHODS

2.1 OPEN-SOURCE OCR PLATFORMS

In this project, we only introduce open-source OCR engines, for the reason that they are highly customizable and allows for local training. The most popular open-source OCR platforms are Tesseract (Smith 2007), Kraken (Kiessling 2019), and Calamari (Wick, Reul and Puppe 2020).

Tesseract provides a wide variety of trained language models with remarkable inference speed and accuracy. It also includes basic image pre-processing and several segmentation methods. Starting from Tesseract 4.0, LSTM-based network architectures are supported. Currently, Tesseract models can be trained on CPU, and the support for GPU training is still an experimental feature.

Both Kraken and Calamari are based on OCRopy. Kraken makes some optimizations for historical and non-Latin script material, and both its recognition and layout analysis models are trainable. Calamari focus only on the recognition model, and currently does not support pre-processing, segmentation and post-processing. Both of them support training models on GPU.

These platforms are easily accessible on GitHub [1], and we will use their code in this project.

(Matteo, Sven and Bruce 2021) have compared the OCR performance of Calamari, Kraken (with Ciaconna) and Tesseract in their paper. They report that overall Tesseract outperforms the other two OCR engines except the recognition of Greek texts. Based on this observation, we mainly focus on improving Tesseract's performance on Greek texts in this project.

2.2 DATASET AND PRE-PROCESSING

In Tesseract, the language models are trained with synthetic text images, so the distribution of images might be different from real-world images. To overcome this issue, we use PoGreTra from (Matteo, Sven and Bruce 2021), a large-scale dataset with real-world scanned Greek commentary images, to fine-tune Tesseract's Greek model or train a new one from scratch.

To fine-tune Tesseract's Greek model, we need to make sure the dataset only contains Greek letters. So we used regular expression to clean the PoGreTra dataset and only keep the images with only Greek letters. The dataset after cleaning contains 23504 line images.

In this project, we will first give a preliminary demonstration for the effect of image pre-processing, then the image pre-processing pipeline for Campbell will be explored.

[1] Tesseract:tesseract-ocr/tesseract, Kraken:mittagessen/kraken, Calamari:Calamari-OCR/calamari

2.3 FINE-TUNING AND RE-TRAINING

There are two commonly used techniques to get a new model from our dataset, namely fine-tuning and re-training. In fine-tuning, the goal is to start from an existing model and let it learn the new patterns from our dataset. Usually the learning rate for fine-tuning is relatively small. Different from fine-tuning, re-training tries to train a new model from scratch using our dataset.

2.4 IMAGE PRE-PROCESSING

The quality of image plays an important part in the OCR accuracy, and there have been efforts in image pre-processing that aims to improve the performance of OCR. Traditional methods include the denoising algorithms (e.g. Gaussian low-pass filtering), edge removal algorithms (e.g. Canny edge detector), and morphological operations (e.g. dilation, erosion, opening, closing). There are also deep learning algorithms (Sporici, Cușnir and Boiangiu 2020) that adaptively pre-process the images.

2.5 EXPERIMENT SETUP

We followed the instructions on the official GitHub page to setup the training environment and configs. Unless specified, we use the learning rate 0.0001 for fine-tuning and 0.002 for retraining. We use the same configuration file [2] as the original Greek model.

2.6 EVALUATION METRICS

Since the study of image pre-processing is still in an early stage, we will give some image examples and evaluate them. For evaluation of fine-tuned or retrained models, we use 12 page images from 2 commentaries: Campbell and Schneidewin. They are not included in PoGreTra dataset so that the evaluation results show the transferability of the models. The first commentary contains English and Greek texts, while the second one contains German and Greek texts. In both commentaries, Greek characters contribute to approximately 26% of the texts. A detailed view of these two commentaries is shown in Table 2.1. Both of them include a large portion of commentary texts, and the number of Greek characters in the commentary section is low, i.e. around 20%. Most of the texts in primary section are Greek texts.

	Campbell				Schneidewin			
	Global	Commen.	Primary	Intro.	Global	Commen.	Primary	Intro.
#chars	16415	8067	1762	4734	8436	3497	1332	1842
#Greek chars	4309	1600	1590	58	2191	714	1230	222
ratio of Greek chars (%)	26.25	19.83	90.24	1.23	25.97	20.42	92.34	12.05

TABLE 2.1
Number of characters and Greek characters in each section of two commentaries. The ratio of Greek characters are measured in %.

The performance of new Greek models will be measured in Character Error Rate (CER), i.e. the accuracy in character level. The CER will be calculated for the whole document, and also for each section of the document, to give both overall and detailed views of the performance. We will also use an available model called GT4HistOCR [3] during evaluation. This model is trained on GT4HistOCR dataset by (Springmann et al. 2018), containing historical documents in German Fraktur and Early Modern Latin.

[2] The required files are accessible via `https://github.com/tesseract-ocr/langdata`

[3] Download link here

Chapter 3

Results

3.1 Performance for Different Fine-tuning Iterations

Following the suggestion from a post on Tesseract Google Group, we limit the number of iterations to 100-2000 and evaluate how fine-tuning iterations will affect the performance. Since we are only using a few images for fine-tuning, we set the portion of validation dataset to 90% of the whole, i.e. around 21k.

Figure 3.1 and Figure 3.2 shows the results. In both figures, CER generally decreases as the finetuning goes on, and the improvement is minor when fine-tuning iteration is more than 1000. This implies that a large dataset is not necessary to improve the performance of OCR, only a small number of images will suffice. With the combination of original Greek model, fine-tuned Greek model, GT4HistOCR and corresponding language model for the commentaries (i.e. English for Campbell and German for Schneidewin), we can significantly reduce both Global CER and Greek CER. If we make a fair comparison and remove the original Greek model from this combination, the performance is still better than the original model, but the CER in Greek texts is less stable and worse than the full combination.

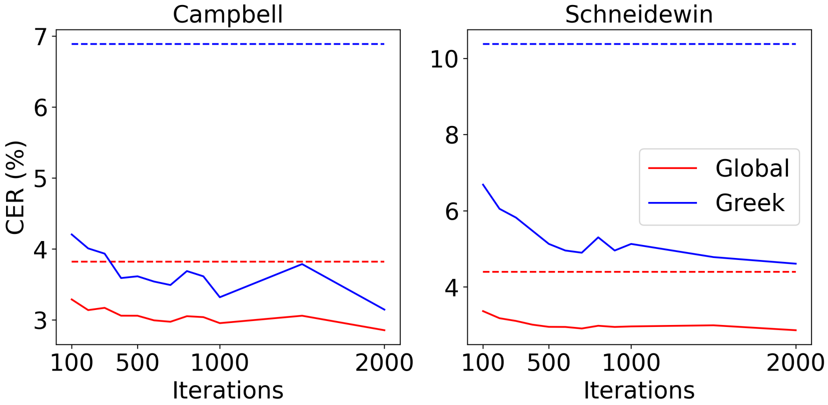


Figure 3.1
CER of models with different finetuning iterations. The finetuned models are evaluated using the best model combination, i.e. finetuned-GT4HistOCR(with grc). Dash lines represents the best performance of original Greek model (grc-GT4HistOCR).

We also fine-tuned the model for different epochs instead of iterations. The number of data for evaluation set is set to 5% to ensure that the training dataset is large enough. After we fine-tuned for 4 epochs, we get our best fine-tuned model. Using the combination of that model, the original Greek model, GT4HistOCR

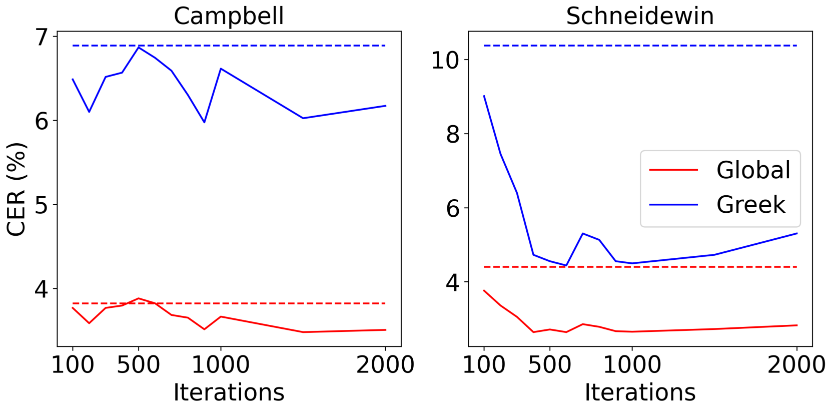


FIGURE 3.2
CER of models with different finetuning iterations. The finetuned models are evaluated using the fair model combination, i.e. finetuned-GT4HistOCR. Dash lines represents the best performance of original Greek model (grc-GT4HistOCR)

and corresponding language model for the commentaries, we achieve global CER of 2.72% and Greek CER of 3.05% for Campbell, global CER of 2.52% and Greek CER of 3.57% for Schneidewin.

3.2 PERFORMANCE FOR DIFFERENT RETRAINING EPOCHS

We tested how OCR performance will change over training epochs. Figure 3.3 and Figure 3.4 shows the performance of retrained models with training epochs from 1 to 6. By using the full combination of models, the retrained models can significantly reduce the CER for both commentaries, and the CER is stable as the training epoch increases. However, after moving the original Greek model from the combination, there is a notable increase of CER for the first commentary, and the retrained model cannot outperform the original one. This indicates that more efforts are needed (e.g. more training epochs, data augmentation, etc.) to obtain a competitive retrained model. It also reveals that fine-tuning is much more efficient and competitive than retraining in this context.

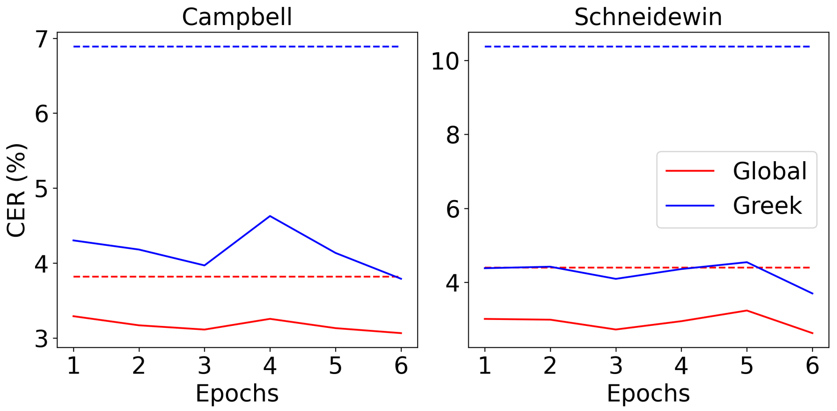


FIGURE 3.3
CER of models with different retraining epochs. The retrained models are evaluated using the best model combination, i.e. retrained-GT4HistOCR(with grc). Dash lines represents the best performance of original Greek model (grc-GT4HistOCR).

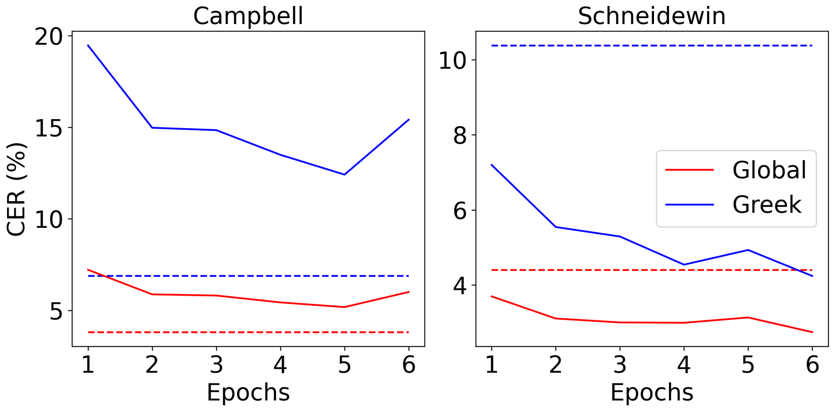


FIGURE 3.4
CER of models with different retraining epochs. The retrained models are evaluated using the fair model combination, i.e. retrained-GT4HistOCR. Dash lines represents the best performance of original Greek model (grc-GT4HistOCR)

3.3 COMPARISON BETWEEN OBTAINED MODELS

We compared our best fine-tuned Greek model and our best retrained one with the original Greek model. Since the choice of language models will affect the final performance, we evaluated them in different combinations of language models. For the original model, we evaluated both with GT4HistOCR and without, and add commentary-specific language models as discussed in Section 2.2. For both the fine-tuned model and retrained model, we use four combinations: only the model, the model with GT4HistOCR, the model with original Greek model, and using all of the three models.

Model	Campbell		Schneidewin	
	Global CER	Greek CER	Global CER	Greek CER
grc	3.87	6.86	4.58	10.76
grc-GT4HistOCR	3.82	6.89	4.40	10.38
finetuned	3.02	5.18	2.45	3.46
finetuned-GT4HistOCR	2.89	4.99	2.34	3.46
finetuned(with grc)	2.79	3.19	2.61	3.57
finetuned-GT4HistOCR(with grc)	2.72	3.05	2.52	3.57
retrained	5.33	12.49	3.13	4.47
retrained-GT4HistOCR	5.19	12.41	3.13	4.93
retrained(with grc)	3.19	4.25	3.21	4.08
retrained-GT4HistOCR(with grc)	3.13	4.14	3.24	4.54

TABLE 3.1
Character Error Rate of the best fine-tuned model, best retrained model and original model (grc) on the commentaries. We use an additional English model during OCR of Campbell and an additional German model during OCR of Schneidewin. The CER is measured in %.

Table 3.1 shows their performance on the commentaries in terms of global CER and Greek text CER. Our fine-tuned model yields better performance in all language model combinations. By replacing the original Greek model with the fine-tuned one, the global CER reduces around 23% for the first commentary and 46% for the second commentary. The Greek text CER reduces around 28% for the first commentary and 70% for the second one. The retrained model performs worse than the original model in the first document, but after using both of them for OCR, the CER will be reduced by 18%. It works well on the second commentary, with 27% lower global CER and over 50% lower Greek CER. Overall the retrained model is less accurate than the fine-tuned model but more accurate than the original one.

We observe that using both the original Greek model and the fine-tuned/re-trained model will result in lower CER, this infers that these models are complementary with each other. Our results also show that our obtained models are transferable, because the datasets for training and evaluation are different.

We also remove other models, and only test the Greek models on Greek characters in the commentaries. The results are shown in Table 3.2. Our fine-tuned model achieves 29.87% and 70.46% less CER than the original one on two commentaries. Our retrained model performs worse in Campbell, with an increase of 186.6% CER. However, it reduces 54.63% of CER than the original one in Schneidewin.

Model	Campbell	Schneidewin
grc	2.31	8.97
finetuned	1.62	2.65
Δ	-29.87%	-70.46%
retrained	6.62	4.07
Δ	+186.6%	-54.63%

TABLE 3.2
CER of original Greek model (grc) and the best fine-tuned model with no additional language models. The CER is measured in %.

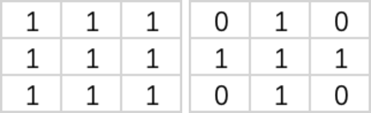
The CER for Greek texts in each section of the commentaries are shown in Table 3.3. It can be seen that the CER in commentary sections are the highest among all sections. This is because compared with primary text (mostly Greek) and Introduction (mostly non-Greek), the commentary part is a mixture of 20% of Greek texts and 80% of non-Greek texts, and this mixture limits the performance of the OCR models. For both the fine-tuned model and the retrained one, their performance on the primary texts and introduction is better. For the first commentary, the performance on its commentary section for both fine-tuned and retrained models does not improve until adding the original Greek model, meaning that the improvement of the these models mainly comes from primary texts and introduction. For the second commentary, the CER on its commentary section for both fine-tuned and retrained models drops to 50% less, accounting for the decrease of global Greek CER. Besides, we witness similar trends as in Table 3.1, i.e. generally, the fine-tuned model works better than the retrained model in all sections.

Model	Campbell				Schneidewin			
	Overall	Commen.	Primary	Intro.	Overall	Commen.	Primary	Intro.
grc	6.86	10.46	3.25	1.72	10.76	18.85	8.34	4.12
grc-GT4HistOCR	6.89	10.31	3.25	1.72	10.38	17.42	8.34	4.12
finetuned	5.18	10.49	1.31	0.00	3.46	7.91	2.22	1.03
finetuned-GT4HistOCR	4.99	10.00	1.31	0.00	3.46	6.63	1.42	1.03
finetuned(with grc)	3.19	6.30	0.65	0.00	3.57	8.93	1.96	1.55
finetuned-GT4HistOCR(with grc)	3.13	5.92	0.65	0.00	3.57	8.93	1.96	1.55
retrained	12.49	22.82	3.20	20.69	4.47	10.33	2.30	1.55
retrained-GT4HistOCR	12.41	22.46	3.20	20.69	4.93	10.33	3.01	1.55
retrained(with grc)	4.25	7.66	1.17	1.73	4.08	8.93	2.75	1.03
retrained-GT4HistOCR(with grc)	4.14	7.22	1.17	1.72	4.54	8.93	3.46	1.03

TABLE 3.3
CER of Greek characters in each section of the commentaries, using original Greek model (grc), the best fine-tuned model and the best retrained model. The CER is measured in %.

3.4 PRELIMINARY RESULTS ON IMAGE PRE-PROCESSING

Due to limited time for this project, the experiment of image pre-processing is still at an early stage. We first show the efficacy of image pre-processing, and then give a quantitative analysis on our selected image pre-processing pipeline.



1	1	1
1	1	1
1	1	1

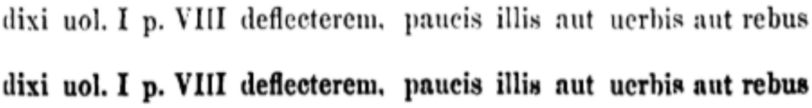
0	1	0
1	1	1
0	1	0

FIGURE 3.5
The kernels used for preliminary study on image pre-processing. The first one is a square kernel, and the second one is a cross kernel.

We first use the following operations for image pre-processing:

- Dilation with a 3-by-3 square kernel
- Closing with a 3-by-3 cross kernel

We show these kernels in Figure 3.5.



dixi uol. I p. VIII deflecterem, paucis illis aut uerbis aut rebus

dixi uol. I p. VIII deflecterem, paucis illis aut uerbis aut rebus

FIGURE 3.6
The original line image (above) and the pre-processed image (below) using dilation and closing operations stated in Section 2.4.

We demonstrate the results for image pre-processing for an example image shown in Figure 3.6. The OCR result for the original image is (red characters indicate the OCR errors):

dixi uol. I p. VIII defleeterem, paueis 1Π15 aut uerbis aut rebus

and the OCR result for the pre-processed image is (green characters indicate the corrected OCR results):

dixi uol. I p. VIII deflecterem, paucis illis aut uerbis aut rebus

It is clear that the OCR result of the examplge above have several errors, and image pre-processing can enhance the quality of the image and get rid of these errors. However, this image pre-processing pipeline does not extends well to the whole dataset. Therefore, we chose to fine-tune and re-train the model without image pre-processing, and then developed and evaluated the pipelines after we obtain the best models. We leave the fine-tuning and re-training with pre-processed images as future work.

Through try and trial, we developed the following image pre-processing pipeline for the commentary Campbell:

- Enlarge the image resolution to 3×.
- Erosion using square kernel of size 3.
- Closing using cross kernel of size 5.
- Rescale the image to the original resolution.

We show the results in Table 3.4. Compared with Table 3.1, the CER slightly decreases. This confirms that image pre-processing is capable of improving the performance. Besides, the image pre-processing pipeline works for both the original Greek model and our new models, showing that the image pre-processing pipelines for one Greek model can extend to all the other Greek models. However, this pipeline does not work in Schneidewin, inferring that the image pre-processing operations are dependent to different image qualities. Since the images in the second commentary is of high quality, we have not found a helpful image pre-processing pipeline for them. This will be left as future work.

Model	Raw		Processed	
	Global CER	Greek CER	Global CER	Greek CER
grc-GT4HistOCR	3.82	6.89	3.37	6.03
finetuned-GT4HistOCR(with grc)	2.72	3.05	2.43	2.60
retrained-GT4HistOCR(with grc)	3.13	4.14	3.08	3.31

TABLE 3.4

Character Error Rate of the best fine-tuned model, best retrained model and the original Greek model on Campbell. Images are pre-processed before using Tesseract's OCR. We use an additional English model during OCR of this commentary. The CER is measured in %.

Chapter 4

Discussion

Due to limited time span of this project, some of the interesting directions have not been explored in detail. In addition, current experiment results also reveal some potentially useful research directions. We will briefly discuss them in this section.

4.1 Image Pre-processing

In Section 3.4, we just give a simple demonstration of how image pre-processing will help with the OCR performance, and develop a helpful image pre-processing pipeline only for the first commentary. This pipeline will not help for the second commentary, indicating that this pipeline is not extensible to other commentaries. Therefore, a more in-depth study of image pre-processing is recommended after a good Tesseract language model is obtained. Deep learning based methods (Sporici, Cușnir and Boiangiu 2020) can be one solution to develop an adaptive image pre-processing pipeline for all commentaries.

4.2 Retrain the Model with Data Augmentation

Besides image pre-processing, data augmentation can also improve the robustness of the OCR, since the augmented images might have slightly different patterns and features from the original ones, and the model trained from data augmentation might be able to capture them. Existing data augmentation libraries (Ma 2019; Luo et al. 2020) can be used for data augmentation. The image pre-processing pipeline can also be used as a data augmentation method.

4.3 Better Language Detection

From our experiments, it can be shown that the choice of language models for Tesseract will have a notable effect on the accuracy for each language. For example, the OCR performance on Greek texts will be much higher if we only use the Greek model instead of using the combination of Greek model and other language models. This suggests that Tesseract's language detection of the texts might not be accurate enough. Further research can address this issue and explore a better way of OCR such as making each segment only containing one language, and also having a better language detection model that serves as a plugin to Tesseract.

CHAPTER 5

CONCLUSION

In this project, we studied how to improve the performance of OCR on Greek commentaries, and especially focused on image pre-processing, finetuning Tesseract's Greek model and retraining a new Greek model using a large-scale Greek commentary dataset. Our fine-tuned Greek model outperforms the original one with only a few fine-tuning iterations. Our retrained model can also achieve better performance with proper selection of language models. Preliminary experiments on image pre-processing has demonstrated its capability to improve the OCR performance. Furthermore, we observe that the fine-tuned model and retrained model can be complementary to the original Greek model, and using them together in the OCR will reduce both overall CER and Greek CER. Further research can be conducted in the field of image pre-processing, data augmentation, better segmentation algorithms and better language detection algorithms.

Bibliography

Reul, Christian et al. (2019). 'OCR4all—An open-source tool providing a (semi-) automatic OCR workflow for historical printings'. In: *Applied Sciences* 9.22, p. 4853.

Smith, R. (2007). 'An Overview of the Tesseract OCR Engine'. In: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. Vol. 2, pp. 629–633. DOI: 10.1109/ICDAR.2007.4376991.

Wick, Christoph, Christian Reul and Frank Puppe (2020). 'Calamari - A High-Performance Tensorflow-based Deep Learning Package for Optical Character Recognition'. In: *ArXiv* abs/1807.02004.

Matteo, Romanello, Najem-Meyer Sven and Robertson Bruce (2021). 'Optical Character Recognition of 19th Century Classical Commentaries: The Current State of Affairs'. In: *The 6th International Workshop on Historical Document Imaging and Processing*. HIP '21. Lausanne, Switzerland: Association for Computing Machinery, pp. 1–6. ISBN: 9781450386906. DOI: 10.1145/3476887.3476911. URL: https://doi.org/10.1145/3476887.3476911.

Kiessling, Benjamin (2019). *Kraken - a Universal Text Recognizer for the Humanities*. Version V2. DOI: 10.34894/Z9G2EX. URL: https://doi.org/10.34894/Z9G2EX.

Sporici, Dan, Elena Cușnir and Costin-Anton Boiangiu (2020). 'Improving the Accuracy of Tesseract 4.0 OCR Engine Using Convolution-Based Preprocessing'. In: *Symmetry* 12.5. ISSN: 2073-8994. DOI: 10.3390/sym12050715. URL: https://www.mdpi.com/2073-8994/12/5/715.

Springmann, Uwe et al. (2018). 'Ground Truth for training OCR engines on historical documents in German Fraktur and Early Modern Latin'. In: *J. Lang. Technol. Comput. Linguistics* 33.1, pp. 97–114. URL: https://jlcl.org/content/2-allissues/1-heft1-2018/jlcl_2018-1_5.pdf.

Ma, Edward (2019). *NLP Augmentation*. https://github.com/makcedward/nlpaug.

Luo, Canjie et al. (2020). 'Learn to Augment: Joint Data Augmentation and Network Optimization for Text Recognition'. In: *CVPR*.