End-to-end platform evaluation for Spanish Handwritten Text Recognition

(Evaluación de una plataforma completa para Reconocimiento de Textos Manuscritos en Español)

Eduardo Xamena,[1] Héctor Emanuel Barboza[2] & Carlos Ismael Orozco[3]

Campo temático: Machine Learning.

Abstract

The task of automated recognition of handwritten texts requires various phases and technologies both optical and language related. This article describes an approach for performing this task in a comprehensive manner, using machine learning throughout all phases of the process. In addition to the explanation of the employed methodology, it describes the process of building and evaluating a model of manuscript recognition for the Spanish language. The original contribution of this article is given by the training and evaluation of Offline HTR models for Spanish language manuscripts, as well as the evaluation of a platform to perform this task in a complete way. In addition, it details the work being carried out to achieve improvements in the models obtained, and to develop new models for different complex corpora that are more difficult for the HTR task.

Keywords: handwritten text recognition; segmentation; end-to-end htr; historical manuscripts processing.

[1] CONICET / Universidad Nacional de Salta. dreduardoxamena@gmail.com
[2] Universidad Nacional de Salta. emanuelbarboza5@gmail.com
[3] Universidad Nacional de Salta. ciorozco.unsa@gmail.com

Resumen

La tarea del reconocimiento automatizado de textos manuscritos requiere de diversas fases y tecnologías tanto ópticas como del lenguaje. En este artículo se describe un enfoque para la realización de esta tarea de forma completa, mediante el empleo de aprendizaje automatizado a lo largo de todas las fases del proceso. Además de explicar la metodología empleada, se describe el proceso de construcción y evaluación de un modelo de reconocimiento de manuscritos para el lenguaje español. La contribución original de este artículo está dada por el entrenamiento y evaluación de modelos de Offline HTR para manuscritos en español, así como la evaluación de una plataforma para la realización de esta tarea de forma completa. Además, se detallan los trabajos que se están llevando a cabo para lograr mejoras en los modelos obtenidos, y desarrollar nuevos modelos para distintos corpus de lectura compleja.

Palabras claves: reconocimiento de textos manuscritos; segmentación; htr punto a punto; procesamiento de manuscritos históricos.

1. Introduction

Among the documents that tell different parts of the history of many countries, we can find manuscripts with varied preservation qualities. The challenge of transcribing the texts of these manuscripts to a digital medium turning them legible by a computer is a very complex task, due to the presence of uncountable factors: the different authors and writing styles, the period of history and the place where the texts were written, the quality of conservation of the manuscripts, among others. Moreover, the information contained in these manuscripts is very valuable for the study of events and characters, and their interactions. Texts from judicial dependencies, civil registries, personal notes and correspondence between relevant people in history can be found, which were not explored by means of automated tools.

The digital transcription process for images manuscripts is called Offline HTR (Handwritten Text Recognition). There are different approaches to carry out this task, taking into account the previous detection of paragraphs, text lines, images, decorations, etc. The paragraph and text line detection subtask can be performed by means of Machine Learning (ML) tools trained to identify objects with certain features in images (Jeong et al., 2017). Particularly, Computer Vision architectures have been used with relative success for this purpose, given that certain previously trained deep neural networks provide feature maps that support the detection process (Oliveira et al., 2018). There are also segmentation-free approaches that do not require these previous phases, but consist of ML models trained for a straight transcription process, without previously identifying text lines.

Besides the process of obtaining text lines, there are different techniques to improve the quality of the generated text lines. For example, the accuracy of the results of the final HTR mechanism depend on the correction of rotation and inclination of these lines (Kar et al., 2019). With these techniques, it is possible to homogenize the writing style in many cases, with the consequent improvement in the complete performance of HTR tools.

This paper explains the general methodology of Offline HTR with previous segmentation of text lines, and an end-to-end software implementation of all the associated phases. Section 2 shows the state of the art on the subject. Then, section 3 details the theoretical framework for this process. Section 4 discusses the characteristics of the platform used for this implementation. Section 5 shows the results obtained for a known dataset from HTR literature, and finally section 6 details the conclusions and current and future work of the associated research group on HTR for Spanish historical manuscripts.

2. Related work

Historical documents tell stories about our ancestors and unsuspected facts can be discovered through them. However, the automatic processing of text from images of printed or handwritten scanned documents is a difficult task due to different factors, such as: different typographies, poor image conditions due to the partial decomposition of paper, digitization mechanism, among others. It is possible to extract useful information from documentary sources present in, for example, national archives, in an automated way. The Transcriptorium project (Sánchez et al., 2013) comprises a case study of digital text extraction from a large data set, using extensively proven methods for offline HTR.

HTR can be performed both online and offline. The former corresponds to a workflow with a reader that needs to understand a handwritten message in real time (Ahlawat and Rishi, 2017). For example: recognizing numbers written on bank checks or options on manually filled in forms. The latter (offline HTR) is related to the case of processing large volumes of handwritten text (Sánchez et al., 2013), for example in information extraction tasks. The present work is oriented to an offline HTR task, given the nature of the collected documents.

There are different approaches to the task of recognizing text from handwritten or printed documents. On the one hand, documents are segmented into lines, lines into words and words into characters (Sarathy and Manikandan, 2018). On the other hand, each recognized text line is processed through a set of finite state models or LSTM networks (Castro et al., 2018). New approaches are based on CNN network models followed by RNN layers called BiLSTM such as in Bluche (2015) or Granell et al. (2020), and suitable CNN+RNN configurations were shown to acquire similar performances to more complex networks such as the multidimensional network proposed in Graves and Schmidhuber (2009). The work of Sanchez et al. (2019) comprises a complete guide to the state-of-the-art tools for HTR until 2019.

In a very recent work, higher performances than the previously mentioned are obtained by including the Gated-CNN technology (De Sousa et al., 2020). In addition, the advantage of using Transformers or Encoder-Decoder architectures with Attention mechanisms to capture the dependencies between elements of a sequence is widely known. The training of this type of models that do not use recurrent structures can be performed more efficiently than, for example, LSTM structures, due to the availability and usefulness of parallel computing. In this sense, the works of Michael et al. (2019) and Kang et al. (2020) show complex implementations but with optimal performances and results according to or even better than the state of the art. Another complementary task, spelling correction, is added in the form of additional ML procedures, in the work of Neto et al (2020), enabling a post-processing stage that enhances the complete recognition process.

Different research projects in humanities and social sciences require the transcription of large document datasets, in many cases handwritten, constituting useful resources for information retrieval and visualization platforms. The availability of software tools for the transcription task generates a technological gap that can be filled with HTR platforms such as OCR4All. In this work such strategy is explained, particularly for Spanish handwritten datasets.

3. HTR: Entire process methodology

The typical HTR process consists of the following phases: Binarization of the text image, segmentation of paragraphs and non-text objects, segmentation of lines and words, and finally recognition of the characters in the lines of text. These phases are explained in the following sections.

3.1 Binarization

The binary phase is the process of turning 1 or 0 (black or white) each pixel of the original image, depending on whether it belongs to the foreground or background. There are local methods for performing binarization, which look for intensity thresholds on a node (pixel) and its neighbors, repeating this process throughout all nodes. On the other hand, global methods establish general measures, according to average values of intensity, color, contrast or other criteria. An example of a local method is Niblack's technique (Niblack, 1986), and an example of global binarization is Otsu's method (Otsu, 1979). Nowadays, Otsu's method is still widely used. In Figure 1, the result of the binarization process can be seen on a historical manuscript image.

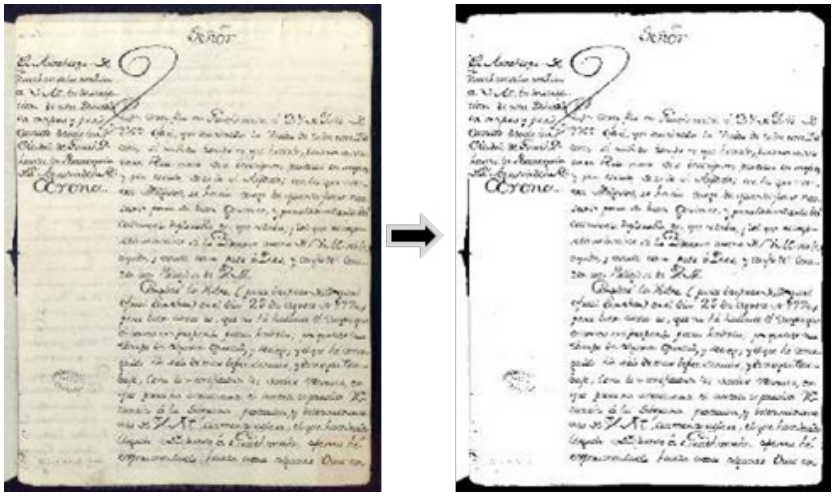

Figure 1: Result of applying a binarization process to an original manuscript image. At left, original image. At right: Binarized image.

Once the binarization process is applied to a manuscript image, the subsequent phases of the HTR process acquire highly better performance. In addition, the storage space and processing time required for a binary image are considerably lower than for the original image, disregarding the process applied.

3.2 Segmentation

The main goal of document segmentation algorithms is the identification of physical limits of paragraphs and non-text objects within the document. In order to carry out an analysis of historical documents, segmentation plays a fundamental role in the pre-processing stage, since its performance has a direct impact on the semantic analysis of the documents. The algorithms associated with this task must have robust mechanisms to handle the numerous variations in the design structure of letters, decoration in writing styles, ink degradation, paper deterioration, as many other factors. Even the optical acquisition mechanisms add noise to the original documents. Figure 2 shows the result of object and line segmentation processes applied to the image of an original manuscript.

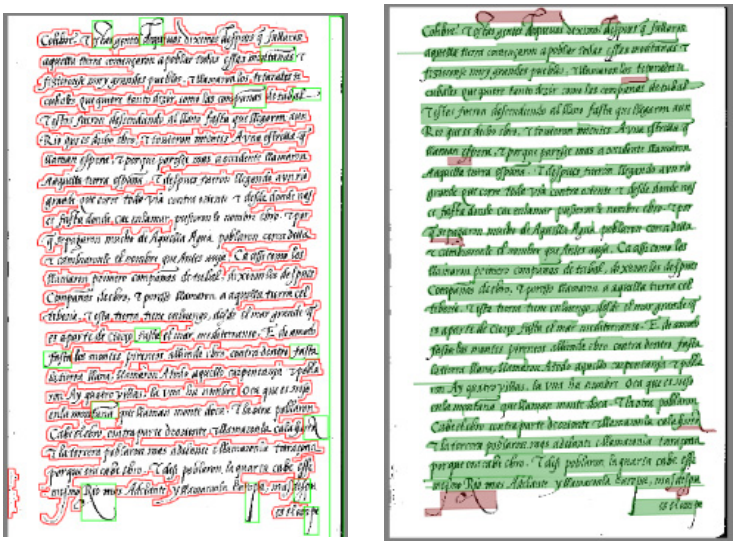

Figure 2: Result of objects and lines segmentation processes applied to an original manuscript image. At left: Objects segmentation. At right: Lines segmentation

Different approaches have been proposed to solve this problem. (Grüning et al., 2019) present a two-stage text line detection method for historical documents, being able to handle complex designs such as curved and arbitrarily oriented text lines. (Oliveira et al., 2018) propose an open source framework called dhSegment

for page extraction, baseline extraction, layout analysis, and photo extraction. They use a convolutional neural network architecture for this purpose. (Liebl and Burghardt, 2020) employ a transfer learning approach that improves the generalization performance of the previous approach.

3.3 Text Recognition

The last phase of the entire Offline HTR process is the text recognition phase. At this stage, with the available text lines detected in the previous segmentation phase -for the case of approaches that use it-, a text string must be obtained for each line identified. Previous approaches used certain handcrafted features of the images, such as the proportion of stroke pixels in each image column, and Hidden Markov Models (HMM) for determining the sequence of output states, which would then form the text string through a process of decoding (Romero et al., 2013). Today, most of the research works that implement machine learning architectures for Offline HTR use convolutional networks (CNN) for the automated extraction of features from line images, followed by recurrent networks or long-short term memories (LSTM) and a subsequent decoding phase for the determination of text outputs. An outline of the two approaches can be seen in Figure 3.

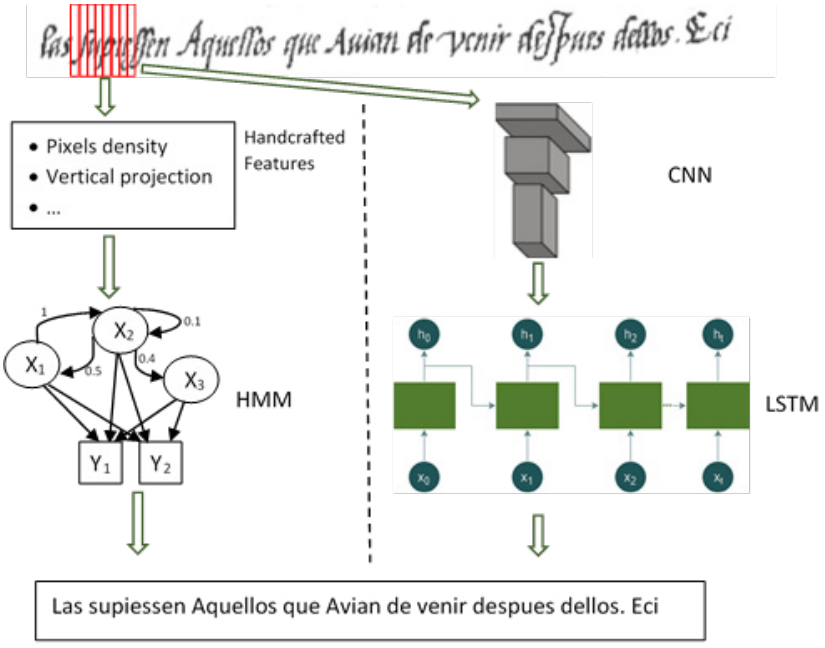

Figure 3: General workflows in the Offline HTR task. At left: Traditional Handcrafted-features and HMM scheme. At right: CNN-LSTM scheme

The first step to obtain a representation of a text-line image is to divide it into columns of a preset size. Each column will be represented by specific features in the traditional approach, such as black pixel density, and by synthetic features extracted by convolutional neural networks according to the most recent approaches. The process continues with the interpretation of the sequence of each line representation. This phase of the process, formerly worked by Hidden Markov Models, is now carried out using bidirectional or multidimensional LSTM neural networks to generate the output signals.

Once the output signals are obtained from the network, the next step is the translation of the output sequence into a valid sequence of characters within the target alphabet. This step is necessary because of the arbitrary length of the columns representing the input steps in the original image representation. The width of these columns will not necessarily correspond to the width of a character or a blank space in the target text, and therefore a decoding step on the obtained signal sequence is required. Figure 4 illustrates this situation.

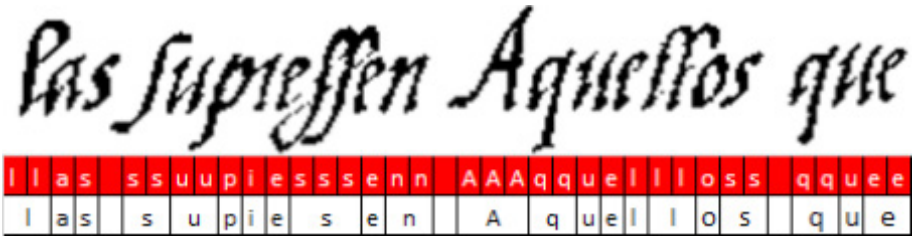

Figure 4: Example of representations of a text line. Above: original text line. In the middle (red background): output sequence from the CNN-LSTM network. Below: decoded output sequence

3.4 Training with sequence decoding

In the decoding process of the CNN-LSTM or Encoder-Decoder network outputs, more than one appropriate segmentation on the output signals is suitable. This is a particular feature of the HTR workflow. In other words, we can find different correct sequences in the output signal regarding the target text string in each case, if we consider the output segmentation not only for the most likely state in each time step. The training of this kind of segmented output to represent the characters can be carried out using specific techniques, such as Connectionist Temporal Classification (CTC) (Graves et al., 2016).

CTC is a method for computing the score value for each output sequence in the optical network. Each of these output sequences is a matrix related to the sequence of features in the text line image. The columns of this matrix represent the probability distributions of the characters (states) associated with each portion

of the image. The rows of the matrix provide different final output sequences that can be segmented as decoded strings. CTC assigns a score to the whole output of the network, according to how many rows produce the expected final sequence after being segmented. In addition, CTC provides a loss function (loss or deviation from the ground truth classification) based on the scores obtained through this approach, for the training of the neural networks that process the data in the prior layers.

4. OCR4All Platform

A complete HTR platform that covers all the phases of this task was required. Not only the text recognition task, but also the segmentation workflows needed to be covered, preferably in one single software package. OCR4All (Reul et al., 2019) was the chosen software package, including the training tasks of HTR models for the Spanish language. This software contains several modules for the image pre-processing phases (binarization, noise removal and paragraph and line segmentation), and also has a complete infrastructure for training, evaluation and inference of text recognition models based on CNN-LSTM architectures. All the above mentioned phases can be carried out through graphical user interfaces over the same OCR4All platform. The LAREX module, for example, allows the operations of region and text line segmentation, with very advanced graphical interfaces. In Figure 5 one of the LAREX screens can be seen.

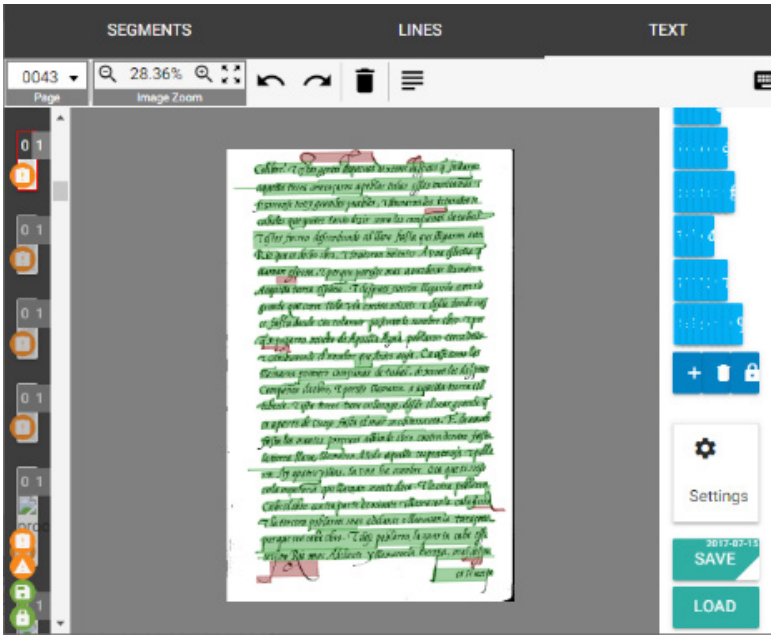

Figure 5: Graphical user interface of the LAREX module, within OCR4All

OCR4All allows different architecture setups for training text line recognition phase models. Figure 6 depicts the platform workflow, showing the possibilities for changing model architectures in terms of CNN, LSTM and Dropout layers as well, in the text recognition phase.

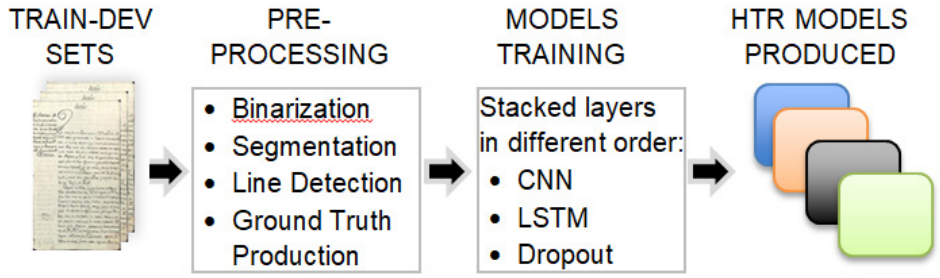

Figure 6: OCR4All models training workflow. After the pre-processing phase, the user can define different setups for deep learning HTR models. CNN, LSTM and Dropout layers can be stacked in different order for each model

The OCR4All software is an open source project, and was implemented in this particular case on the Natural Language Processing and Text Mining platform of the National University of Salta.[4] Currently, we are working on the construction of different HTR models for particular manuscript collections. For example, manuscripts from the collection "Fondo Cabildo, Justicia y Regimiento de Buenos Aires",[5] from the Archivo General de la Nación (AGN, Argentina), were uploaded to the platform.

5. Results

For the evaluation of the usefulness of OCR4All platform in our project, the first part of this work was the transcription of a subset of a manuscript corpus to digital format, upon the base of this software. For this purpose, the dataset of Archbishop Don Rodrigo, about the history of Spain until the 16th century,[6] was used. This corpus consists of 853 pages in old Castilian (Spanish), written by a single author. The writing style of this corpus denotes a very easy to read and nearly calligraphed text, with well separated lines. Most pages contain a single text block, in one column, and 24 lines on average.

Approximately 150 pages of the Rodrigo corpus were segmented and transcribed, using the LAREX module for object and line segmentation, and also for production of Ground Truth data. Although there are already versions of ground truths generated for this dataset, the re-transcription was carried out to

[4] http://nlp.unsa.edu.ar:1476/
[5] Code AR-AGN-CJR01-3023, Archivo General de la Nación, Argentina.
[6] https://www.prhlt.upv.es/wp/resource/the-rodrigo-corpus

correct possible errors present in the original transcriptions and to make the data structures compatible with the OCR4All software.

Particularly for the Rodrigo dataset, all the pages only contain text, and often a single block. There are no drawings or tables nor figures, making easier the segmentation task. We are currently working on the transcription of other corpora with different writing styles that are less easy to read by humans, and the difficulty is notably higher on every task than for Rodrigo dataset. However, the segmentation routines from OCR4All did not produce very good recognition of paragraphs and lines in Rodrigo documents in some cases, requiring manual identification and correction during this stage.

After the transcription, some models were generated and employed to make the first evaluations of the software, and therefore of the ML architectures used internally by OCR4All. The general architecture used by OCR4All to generate the recognition models can be seen in Table 1. However, the software allows the configuration of this architecture using several parameters, allowing deep architectures with CNN and LSTM layers interspersed in different orders. Also the dimensions of each CNN layer, the number of units in each bidirectional LSTM layer and the inclusion of dropout capabilities can be configured, among other features.

Table 1: Basic scheme of the CNN-LSTM network used by OCR4All

# Layer	Type
1	Input
2	CNN-2D
3	Pool-2D
4	CNN-2D
5	Pool-2D
6	Bidirectional LSTM
7	Dropout
8	Softmax

The generated models can be downloaded from the github channel of one of the authors.[7] Two of the models were trained with only 50 of the pages transcribed, while the other 5 with 100 pages. The evaluation of the trained models was carried out on three groups of a subset of full pages of the Rodrigo dataset, which were not included in any of the training phases, in order to have a clear idea of the generalization capability of the models. Initially, an evaluation was carried out on 4 of the pages set aside for testing, totaling approximately 96 lines of text. Then, a group of 10 pages was taken (about 240 lines) and finally the complete group of 37

[7] https://github.com/edus1984/HTR-models

pages (about 890 lines) of the test set. As shown in Table 2, the value of Average sentence confidence becomes higher as the number of test instances increases, but also follows this trend the value of Normalized label error rate, indicating that the error rate is growing for some of the characters of the alphabet. Due to the high variability of handwriting in general, these errors are considerably lower in proportion when post-processing phases with target language models are included.

Table 2: Results of the evaluations performed on test pages of the Rodrigo dataset

# Pages for training	# Pages for test	Average sentence confidence	Normalized label error rate
50	4	72,22%	9,38%
100	4	72,74%	5,46%
50	10	74,97%	10,15%
100	10	74,75%	6,62%
50	37	74,86%	14,66%
100	37	76,78%	11,23%

6. Conclusions and Future work

In this work the different phases of the handwritten text recognition process (HTR) were explained, and the implementation of a software platform for this task was also detailed. In particular, different HTR models were developed for the Spanish language, on a known dataset of the literature. The results of the application for this case were encouraging, but they show the need for other additional processes on the same output produced by the HTR models generated. The average error rates for each of the character labels are quite high and the confidence rates in the text lines generated do not even reach 80%, so the text recognition obtained must be improved with auxiliary tools, or even machine learning architectures different from the employed in this work.

In the near future some of the architectures mentioned in the state of the art will be employed to evaluate and implement models over the Rodrigo dataset, to determine the levels of improvement that can be obtained in general. In addition, the transcription of new datasets is being performed on the OCR4All platform, as another research objective in the field of Humanities and Social Sciences. The results obtained for these new case studies will also be analyzed with different HTR architectures. The origin of the new datasets explored is the AGN. This work is a task of a bigger project of information retrieval and visualization tools development, in the scope of Argentinian history.

Acknowledgement

This work was carried out with funds from Universidad Nacional de Salta (Proyectos CIUNSa C 2659 y CIUNSa A 2364), CONICET (Proyecto UE 22920160100056CO), and Universidad Nacional del Sur (PGI-UNS 24/N051). Thanks to Archivo General de la Nación (AGN, Argentina) for allowing the publication of manuscripts from its documentary collections, and NVIDIA for the donation of a TITAN Xp GPU for Departamento de Informática, Facultad de Ciencias Exactas, Universidad Nacional de Salta, Argentina.

References

Ahlawat, S. & Rishi, R. (2017), Off-line handwritten numeral recognition using hybrid feature set–a comparative analysis, Procedia computer science 122, 1092--1099.

Bluche, T. (2015), Deep neural networks for large vocabulary handwritten text recognition, PhD thesis, Paris 11.

Castro, D.; Bezerra, B. L. D. & Valença, M. (2018), Boosting the deep multidimensional long-short-term memory network for handwritten recognition systems, in 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 127--132.

De Sousa Neto, A. F.; Bezerra, B. L. D.; Toselli, A. H. & Lima, E. B. (2020), HTR-Flor++: A Handwritten Text Recognition System Based on a Pipeline of Optical and Language Models, in Proceedings of the ACM Symposium on Document Engineering 2020, Association for Computing Machinery, New York, NY, USA.

Granell, E.; Romero, V. & Martinez-Hinarejos, C.-D. (2020), Study of the influence of lexicon and language restrictions on computer assisted transcription of historical manuscripts, Neurocomputing.

Graves, A.; Fernández, S.; Gomez, F. & Schmidhuber, J. (2006), Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, in Proceedings of the 23rd international conference on Machine learning, pp. 369--376.

Graves, A. & Schmidhuber, J. (2009), Offline handwriting recognition with multidimensional recurrent neural networks, in Advances in neural information processing systems, pp. 545--552.

Grüning, T., Leifert, G., Strauß, T., Michael, J., & Labahn, R. (2019). A two-stage method for text line detection in historical documents. International Journal on Document Analysis and Recognition (IJDAR), 22(3), 285-302.

Jeong, J.; Park, H. & Kwak, N. (2017), Enhancement of SSD by concatenating feature maps for object detection, CoRR abs/1705.09587.

Kang, L.; Riba, P.; Rusicol, M.; Fornés, A. & Villegas, M. (2020), Pay Attention to What You Read: Non-recurrent Handwritten Text-Line Recognition, arXiv preprint arXiv:2005.13044.

Kar, R.; Saha, S.; Bera, S. K.; Kavallieratou, E.; Bhateja, V. & Sarkar, R. (2019), Novel approaches towards slope and slant correction for tri-script handwritten word images, The Imaging Science Journal 67(3), 159--170.

Liebl, B., & Burghardt, M. (2020). An Evaluation of DNN Architectures for Page Segmentation of Historical Newspapers. arXiv preprint arXiv:2004.07317.

Michael, J.; Labahn, R.; Grüning, T. & Zöllner, J. (2019), Evaluating sequence-to-sequence models for handwritten text recognition, in 2019 International Conference on Document Analysis and Recognition (ICDAR), pp. 1286--1293.

More, P. K. & Dighe, D. D. (2016), A review on document image binarization technique for degraded document images, Int. Res. J. Eng. Technol, 1132--1138.

Neto, A. F. S.; Bezerra, B. L. D. & Toselli, A. A. H. (2020), Towards the Natural Language Processing as Spelling Correction for Offline Handwritten Text Recognition Systems, Applied Sciences 10(21), 7711.

Niblack, W. (1986), An Introduction to Digital Image Processing (Englewood CliVs, NJ, Prentice-Hall.

Oliveira, S. A.; Seguin, B. & Kaplan, F. (2018), dhSegment: A generic deep-learning approach for document segmentation, in 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 7--12.

Otsu, N. (1979), A threshold selection method from gray-level histograms, IEEE transactions on systems, man, and cybernetics 9(1), 62--66.

Reul, C.; Christ, D.; Hartelt, A.; Balbach, N.; Wehner, M.; Springmann, U.; Wick, C.; Grundig, C.; Büttner, A. & Puppe, F. (2019), OCR4all—An open-source tool providing a (semi-) automatic OCR workflow for historical printings, Applied Sciences 9(22), 4853.

Romero, V.; Fornés, A.; Serrano, N.; Sánchez, J. A.; Toselli, A. H.; Frinken, V.; Vidal, E. & Lladys, J. (2013), The ESPOSALLES database: An ancient marriage

license corpus for off-line handwriting recognition, Pattern Recognition 46(6), 1658--1669.

Sánchez, J. A.; Mühlberger, G.; Gatos, B.; Schofield, P.; Depuydt, K.; Davis, R. M.; Vidal, E. & de Does, J. (2013), tranScriptorium: a european project on handwritten text recognition, in Proceedings of the 2013 ACM symposium on Document engineering, pp. 227--228.

Sánchez, J. A.; Romero, V.; Toselli, A. H.; Villegas, M. & Vidal, E. (2019), A set of benchmarks for handwritten text recognition on historical documents, Pattern Recognition 94, 122--134.

Sarathy, S. & Manikandan, J. (2018), Design and evaluation of a real-time character recognition system, in 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 519--525.