READ

RECOGNITION & ENRICHMENT OF ARCHIVAL DOCUMENTS

D8.14

Large Scale Demonstrators - Venice Time Machine

Meta-learning model

Sofia Ares Oliveira, Frederic Kaplan

EPFL

Distribution: http://read.transkribus.eu/

READ
H2020 Project 674943

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 674943

Project ref no.	H2020 674943
Project acronym	READ
Project full title	Recognition and Enrichment of Archival Documents
Instrument	H2020-EINFRA-2015-1
Thematic priority	EINFRA-9-2015 - e-Infrastructures for virtual research environments (VRE)
Start date/duration	01 January 2016 / 42 Months

Distribution	Public
Contract. date of delivery	31.12.2017
Actual date of delivery	24.11.2017
Date of last update	December 18, 2017
Deliverable number	D8.14
Deliverable title	Large Scale Demonstrators - Venice Time Machine
Type	report
Status & version	reviewed by : Stefan Fiel, Maria Kallio
Contributing WP(s)	WP8
Responsible beneficiary	EPFL
Other contributors	
Internal reviewers	Günter Mühlberger, CVL, NAF
Author(s)	Sofia Ares Oliveira, Frederic Kaplan
EC project officer	Martin Majek
Keywords	Large Scale Demonstrators, Digitization, Transcription, Segmentation, Information Retrieval

Contents

1. Executive summary

The Venice Time Machine (VTM) project aims at building a multidimensional model of Venice and its evolution covering a period of more than 1000 years. The State Archives of Venice possesses an estimated 80 km of shelves that are filled with administrative documents, from birth registrations, death certificates and tax statements, all the way to maps and urban planning designs. These archives are currently being digitized, transcribed and indexed, setting the base of the largest database ever created on Venetian documents.
The Venice Time Machine project wants to give the archives a new, virtual existence on the Web through Cloud access and online tools. It aims to reanimate Venice's past life from them by re-creating social networks and family trees, and visualizing urban development and design.
As one of the four large scale demonstrators of the READ project, the objective is to provide an environment to test and use the technologies developed within the READ consortium on large real-world data and to develop new solutions to deal with large historical archive indexing and retrieval.

2. Main contributions

The work of the second year was structured around the problematic of the information extraction on cadaster maps. Archival documents do not only consist of handwritten text documents, for instance, cadaster map documents are an example of complex and rich document which structure is very different from 'traditional' archival documents. Cadaster plans provide information about the city urban shape as well as information about the urban population and city functions (census information, property, rent prices, etc.) The 1808 Venetian cadaster enables the study of the social and the economic structure of Venice but also its urban organization, thus being a key document for reconstructing dense representations of the history of the city. Being able to easily access and search these archival documents opens new perspectives to study and visualize the evolution of a city and its inhabitants.
The processing of cadaster maps can be separated into the transcription system for the recognition of parcels' numbers and the segmentation system for the extraction of the parcel's shapes. As we show in the next sections, both systems can be used for processing other type of documents and are thus not specific to cadaster plans processing. Thus we also worked on processing handwritten text documents, both on regions and lines segmentation and transcription.

The contributions of EPFL group to the READ project are divided in three sections :

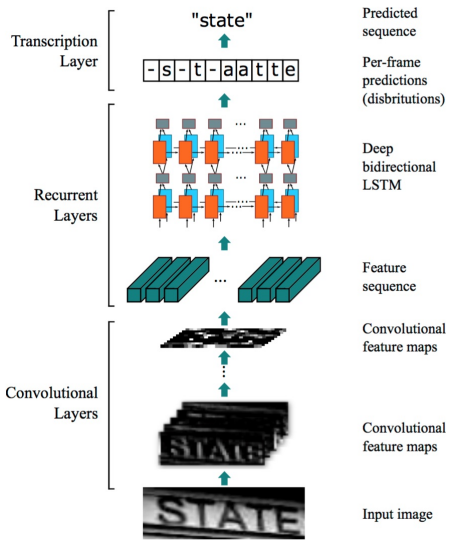
- Transcription system
- Document Segmentation tool (in collaboration with Benoit Seguin)
- Information extraction on cadaster maps
- Extension of NCSR Demokritos tools for python developers

2.1. Transcription using Convolutional Recurrent Neural Networks (CRNN)

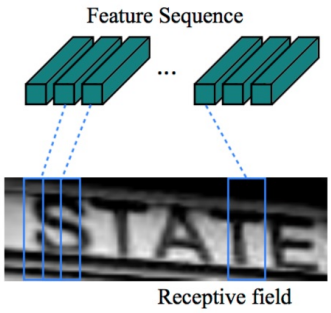
2.1.1. Architecture

We developed a transcription system based on the combination of convolutional and recurrent neural networks as described in [1] for handwritten text (Fig. 1a). The convolutional neural network (CNN) captures hierarchical spatial information, with the first layers capturing low level features and later ones capturing high level features. In other words, CNN are good at encoding spatial data. On the other hand, Recurrent Neural Networks (RNN) are used to capture temporal (or sequential) data, with the ability to capture contextual information within a sequence of arbitrary length. This systems combines the best of both worlds to handle multi-dimensional data as sequences.

From an input image, the convolutional layers extract a sequence of compact representations, the features vectors, which correspond to the columns of the feature map. They are processed from left to right of the image to form a sequence of local image descriptors (Fig. 1b).



(a) Network architecture. The architecture consists of three parts: 1) convolutional layers, which extract a feature sequence from the input image; 2) recurrent layers, which predict a label distribution for each frame; 3) transcription layer, which translates the per-frame predictions into the final label sequence. [1]



(b) Feature Sequence [1]

The sequence is then input to the recurrent layers which consist of stacked bidirectional Long Short-Term Memory (LSTM) cells. LSTM cells [2] have the ability to capture long-range dependencies but they are directional, and thus only use past contexts. Since in image-based sequences context from both directions are useful and complementary, one forward and one backward LSTM cells are combined to form bidirectional LSTM

which are then stacked to have several recurrent layers. The recurrent network outputs per-frame predictions (probabilities) that need to be converted into a label sequence.

In the transcription layer, the connectionist temporal classification [3] is used in order to obtain the "label sequence with the highest probability conditioned on the per-frame predictions". The sequence label (i.e the word or the sentence) is found by taking the most probable label at each time step and mapping the separated labels to the correct sequence label (see [3] to have the detailed explanation on how the repeated and 'blanks' labels are dealt with).

2.1.2. Training

The training was performed on a set of annotated data form various types of Venetian handwritten documents. The set consists of image segments of names, places and numbers that have been transcribed by experts in Venice. The content of the image segments ranges from one to several words. The set was randomly split into 90 % for training and 10% for testing. The training process took a few hours (between 2 and 6) with a Nvidia Titan X depending on the size of the image dataset.
Several experiments were performed using different set of characters (called 'Alphabet' here after) and resulted in one model per Alphabet.
Data-augmentation such as slight rotations and color changing (hue, contrast and saturation) was used in order to artificially increase the size of the dataset and its variety.

2.1.3. Results

To evaluate the performance of the system we use the Character Error Rate (CER) measure on the test set ($CER = (i + s + d)/n$ with i, s, d, n the number of character insertions, substitutions, deletions and total characters respectively). The numerical results are shown in Tab. 1. As expected, we observe that alphabet (3) has the lowest CER, since there are only 10 different characters and the number of image segments that can be used to train the system is quite high. The alphabet (2) has lower CER than alphabet (1) since most of the image segments composing the dataset are images of numbers, thus increasing the performance of the system. A few randomly selected examples can be seen in Appendix A.

Alphabet	# image segments	CER
(1) Capital-lowercase-symbols	24035	0.0887
(2) Capitals-lowercase-digits-symbols	96198	0.045
(3) Digits	72326	0.0133

Table 1: The Character Error Rate (CER) for each Alphabet in Tab. 2

Alphabet	Set of characters
(1) Capital-lowercase-symbols	A-Za-z'.,: -=
(2) Capitals-lowercase-digts-symbols	A-Za-z0-9'.,:; - _ ()[]/
(3) Digits	0-9

Table 2: The set of characters contained in each alphabet

2.2. Document segmentation using pixel-wise segmentation

2.2.1. Architecture

In order to extract regions of interest from documents we developed a general tool for segmenting documents. The system is based on neural networks that allow pixel-wise segmentation by assigning each pixel a label. The architecture is composed of an encoder ('contracting') network followed by a decoder ('expansive') network and is inspired by the U-Net and SegNet architectures [4, 5]. We use as encoder network a pretrained convolutional neural network from which the last fully connected layers are removed. The decoder network converts the low resolution features maps of the encoder to full input resolution features maps by combining the upsampled features maps with the corresponding resolution features maps from the encoder (see Fig. 2 to understand the general idea). Each encoding layer has a corresponding decoding layer, thus resulting in a quasi symmetrical u-shaped architecture. In the case of standard classification, the last layer is a soft-max layer which assigns a probability to each pixel independently and the segmentation output is simply the class with maximum probability for each pixel. If we are interested in multilabel classification we use as last layer a sigmoid layer.

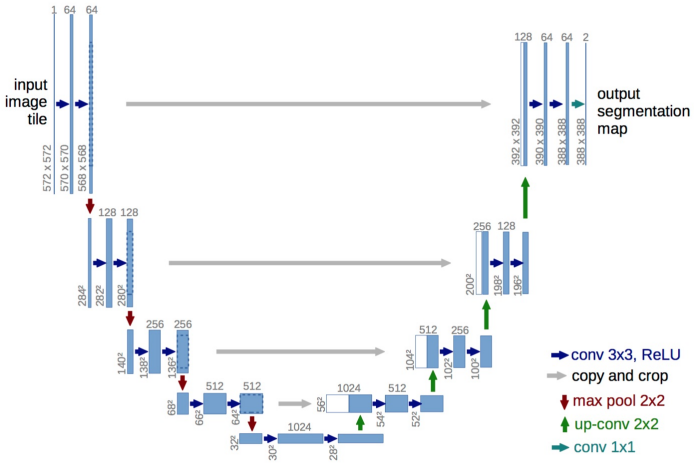


Figure 2: U-Net Architecture. The gray arrows show the combination of upsampled feature maps from the decoder with high resolution feature maps from the encoder to recover full input resolution feature maps. [4]

2.2.2. Training

We used two architectures for the experiments, one using VGG-16 [6] and the other using Resnet-50 [7] pre-trained model as encoders, which are two well-known and widely used architectures. The pre-trained weights of convolutional layers were used in order to take advantage of their already learned features and diminish the cost of the training process. We experimented the segmentation on four type of documents forming four datasets listed in Tab. 3. The training took on average one hour on a Nvidia Titan X. The training used data-augmentation such as slight rotations, flipping (left-right and up-down) and color changing (hue, contrast and saturation). The combination of pre-trained weights for encoding layers and aggressive data-augmentation diminishes the number of training elements needed.

Dataset name	Task
BNF-BCU	Extract ornaments and illustrations in text documents (1 class)
ICDAR	Extract line polygons and baselines in handwritten text documents (1 class)
CAD	Extract parcels and text in venetian cadaster maps (2 classes, multilabel)

Table 3: Datasets and tasks evaluated in the experiments

2.2.3. Results

Tab.4 lists the number of training and testing documents for each task and the classification accuracy (per-pixel), which is computed on the output of the softmax layer. The task for CAD dataset is not evaluated because of lack of testing data (work in progress, see Sec. 2.3). The examples of predictions shown in App. A are the probabilities output by the system (before the softmax layer).

Dataset	# doc. for training	# doc. for testing	Accuracy
BNF/BCU	50/342	0/85	0.989
ICDAR - line polygons	206	52	0.921
ICDAR - baselines	206	52	0.951

Table 4: Results for each task

This general system shows good results for several different tasks and has the advantage of not needing large datasets to be trained. Even if some specific post-processing should be done for each use case in order to obtain clean masks and precise extractions, we believe this approach is sufficiently generic to be applied in a variety of documents segmentation tasks with few annotated data.

The development of this tools has been done in collaboration with Benoit Seguin from EPFL group.

2.3. Cadaster maps processing

Based on the systems exposed in the previous sections, we improved the processing of the cadaster maps proposed in annual report Y1. As a reminder, we developed a fully automated process capable of segmenting and interpreting Napoleonic cadaster maps of the Veneto region dating from the beginning of the 19th century. The system extracts the geometry of the parcels, georeferences it if a coordinate reference system is provided and reads the handwritten labels. This process has the objective of making the cadaster maps searchable, allowing an easier, more efficient and more interactive study of these documents.
First, the transcription system of the old version was replaced by the CRNN architecture (see Fig. 3). The model was trained on numbers extracted from the venitian archives ($\sim$ 30K elements) and on synthetic data generated from MNIST digit dataset (100K elements). On the test set the model has an accuracy of 0.956 and a CER of 0.013. On the parcel's numbers recognition task, the CRNN architecture greatly improves the transcription, see the results in Tab 5.

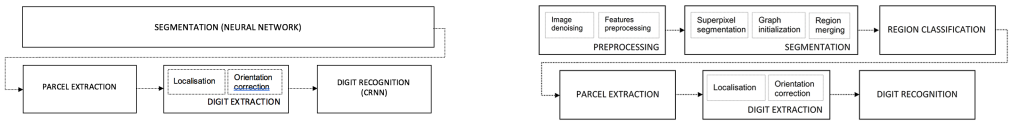


Figure 3: Comparison between the previous (right) and the new (left) cadaster processing system

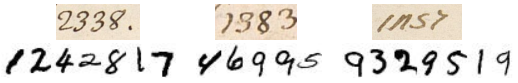


Figure 4: Examples of numbers from the Venetian archives and artificially generated from MNIST dataset

Number transcription	CRNN	2 conv-layers
Recall	0.83 (608)	0.09 (66)
Character Error Rate (CER)	0.14	0.77
Ground truth (# parcels' number)	736	736

Table 5: Results of the transcription using the old cadaster processing system with the new transcription CRNN architecture and the old architecture (2 convolutional layers)

Second, the segmentation steps of the previous system were replaced by the architecture presented in Sec 2.2. The network was trained on cadaster samples with its corresponding labels mask (Fig. 5). As it can be seen in App. A, there are two probability maps output corresponding to the two classes. The possibility for a pixel to belong

to both classes leads to better results when extracting the parcels' shape and the text regions.

Figure 5: Sample of cadaster image with its labels mask (text in green and parcels contours in red). Note that a pixel may belong to both classes (yellow).

The evaluation of parcel's extraction is compared to the previous system in Tab. 6. We consider a parcel to be correctly extracted, if it has an IoU (Intersection over Union) measure of at least 0.8 with the groundtruth (see example of groundtruth mask in Fig. 6). The recall is slightly better but the precision decreases. This is mainly due to the fact that with the new system, we extract all the vectors/lines of the cadaster and some post-processing would be necessary to discard non-parcel elements. These are preliminary results and we are currently working on this task.

Parcel extraction	neural network	old architecture
Recall	0.79 (941)	0.72 (583)
Precision	0.44	0.51
Ground truth (# parcels)	1185	810

Table 6: Results of the transcription using the old cadaster processing system with the new transcription CRNN architecture and the old architecture (2 convolutional layers)

With the new segmentation algorithm it is possible to directly export the contours as a vector layer to be used in GIS systems (Fig. 7). When using georeferenced images, we are able to superimpose the vectorized parcels from 1808 to an actual map as shown in Fig. 8.

Regarding the full system with the updated segmentation and transcription modules, it has already been implemented end-to-end. Current work focuses on improving the

Figure 6: Annotated groundtruth for parcel extraction

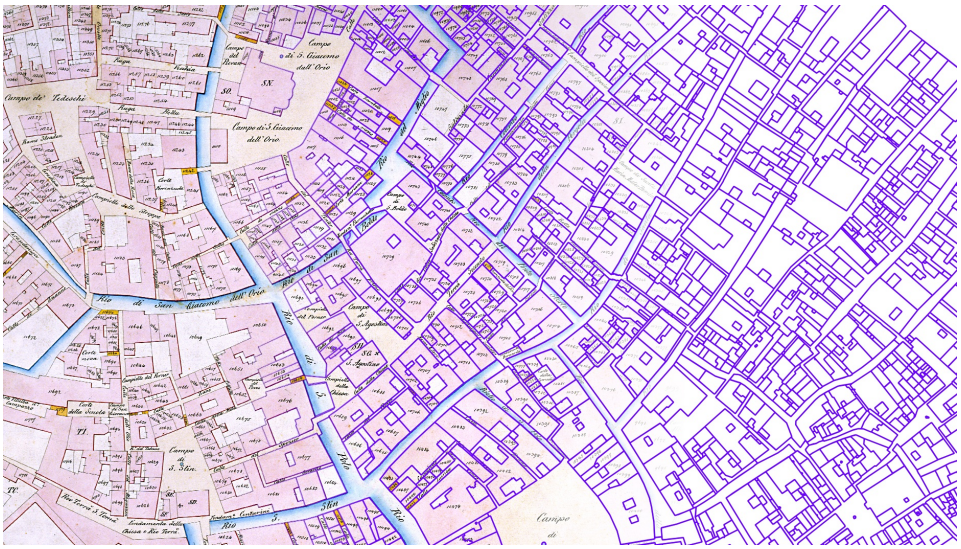


Figure 7: Parcel extraction using pixel-wise segmentation to obtain polygonal vector shapes

results of the transcription, by correcting the errors using contextual information from neighbouring parcels but also by improving the localization of the number.

2.4. Misc. : Extension of NCSR Demokritos' tools

We extended the tools developed by NCSR Demokritos partners (Binarization, TextLineSegmentation, WordSegmentation and FromBaseLinesToPolylines tools) to allow their testing within our workflow but also to make them available to python developers. This work consisted in building a cython wrapper for the C++ code and resulted in an easy and straightforward usage in python programming language. The python bindings can be found on the original repository `https://github.com/Transkribus/NCSR_Tools` (see the Readme_cython).

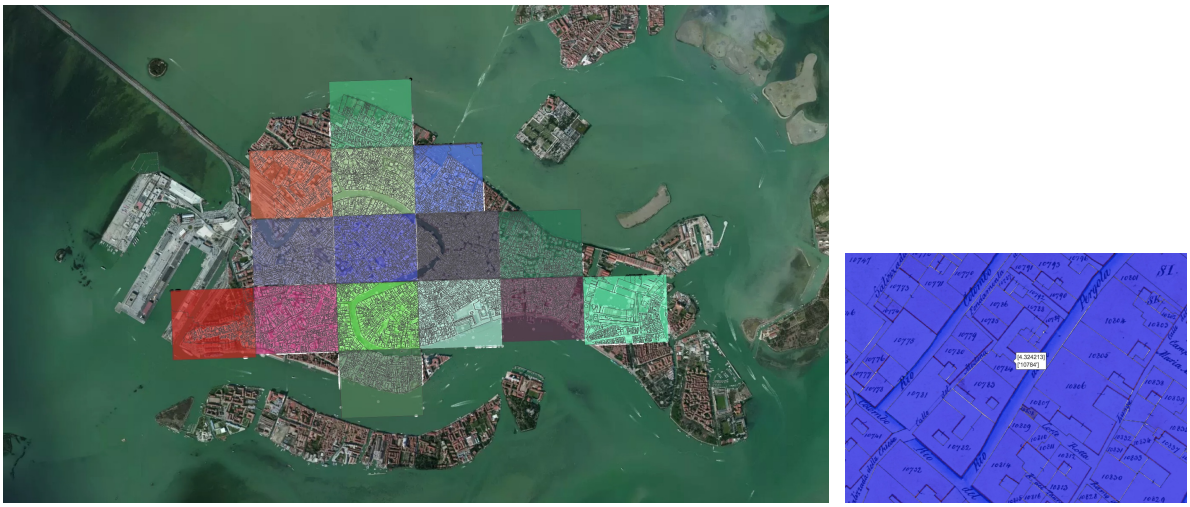


Figure 8: View of the georeferenced parcels from 1808 Napoleonic cadaster on satellite image of Venice. This image is a screenshot of an interactive visualization that allows to display the transcription when a parcel is hovered or clicked

3. Work in progress

EPFL group is currently working on assembling the document segmentation system with the transcription system in order to have a full workflow allowing a complete processing from the image document to its transcription. We will also explore the possibility of a single system trained end-to-end to perform both tasks.
Regarding the cadaster processing, future work will focus on adapting the digit extraction steps to the new methods in order to achieve better number transcription. We plan to use this tool to facilitate the study of spatial relations in the cadastral documents and registers and possibly open new lines of research. We will also try to apply the system to other types of cadasters in order to confirm the generality of the method.

References

[1] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 11, pp. 2298–2304, 2017.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–1780, Nov. 1997.

[3] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, ACM, 2006.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomed-

ical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer, 2015.

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, 2015.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

A. Examples

Figure 9: Transcription examples. GT is the groundtruth annotated by the Venetian expert, P is the prediction of the system.

Concernant la Taille. 327

EDICT

De ne verifier aucunes Lettres d'exẽptions, sans en bailler advis à S. A. R. & en recevoir les Ordres de vive voix.

CHARLES EMANVEL. Par la grace de Dieu, Duc de Savoye, &c. A nos tres-Chers bien Amés & Feaux Conseillers les gens tenans nos Senat & Chambre des Comptes en Savoye, Salut. Comme aprés avoir fait rechercher tous les moyens plus propres & convenables pour faciliter l'exaction de nostre droict de Commutation de Gabelle du Sel & Vin également sur nos sujets, le fort portant le foible, sans foule & à la décharge du tiers-état, finalement par nostre Edict du 27. Mars dernier verifié & publié. Entre autre, avons fait distinction des qualités, exempts, & declarations des Contribuables par cy-aprés; Lequel Voulons & Entendons être inviolablement entretenu & observé & qu'il sorte effet, sans qu'aucuns des Contribuables, ou leurs Successeurs, prendront le temps prefix par iceluy, de puissent par cy aprés eximer, ou décharger de leur Cotte des Tailles, au détriment & surcharge du public & tiers-état, pour quelque pretexte de Noblesse, Declarations, ou Exemptions, que par priores, & grande importunité, ou surprise, ils pourroient avoir obtenu, ou obtenir par cy-aprés de Nous, & lesquelles provisions ne voulons sortir effet. A cette cause, vous mandons & donnons tres-expressement, & deffendons dores en avant procéder à la verification d'aucunes Lettres d'exemption qui pourroient estre obtenuës de Nous sous quelque pretexte ou couleur que ce soit contraires à nostre Edict, sy à icelles avoit aucun égard, sans préalablement & expressément nous faire entendre de vive voix & remonstrer du préjudice & consequence, & sur ce attendre & avoir verbal commandement de Nous: Car ainsi Nous voulons & Nous plait être fait; quelles clauses derogatoires, comminatoires, que autres provisions pourroient être apposées, auquelles & aux derogatoires, comminatoires par ces Lettres, avons expressément dérogé & dérogeons. Donné à Chambery le 8. Septembre 1584. CHARLES EMANVEL. Visa MILLIET, BRVYSET.

Le present Edict a été verifié, publié, lu, & enregistré le 7. Novembre 1584.

Si a EDICT

74 VARON,

ACTE II.

SCENE PREMIERE.

VARON, EURIBAN.

VARON.

Ecroirai-je mes yeux? Suis-je dans cet azile,
Dans ce même Palais, en revers si fertile,
Où surpris sans secours dans les bras du sommeil,
J'eus peine à fuir la mort, offerte à mon reveil?
Souffre que mes regards en parcourent l'enceinte;
Souffre qu'un lieu funeste, où ma honte est empreinte,
Ranime un désespoir qui ne s'est occupé
Que du sort du Barbare, à mes coups échapé.
Quoi! d'un sang odieux la source coule encore!
Ah! déja dans mon cœur ma haine le dévore.
Parle. Que dit ma Fille? Est-ce ici que tes soins,
Que ton zéle à mes yeux doit l'offrir sans témoins?

EURIBAN.

Oui, Seigneur, vous voyez la paisible retraite
Où son cœur s'abandonne à la crainte inquiéte.
Softrate a défendu qu'on y vienne épier
Les soupirs que son trouble ose m'y confier.
J'ai sçu, par des détours que l'on connoit à peine,
Vous guider jusqu'aux murs de l'enceinte prochaine.
Ne craignez rien. D'ailleurs, sous ce déguisement,
Vos traits sont fi voilés qu'on s'y trompe aisément.
Votre Fille, elle même, auroit pû s'y méprendre,

Figure 10: Mask of probabilities output by the segmentation model for ornaments and illustrations extraction on BNF/BCU dataset. The first row are images from the BNF colection and the second row are images from the BCU collection.

Figure 11: Mask of probabilities output by the segmentation model for line extraction on ICDAR dataset. The first row are images from ICDAR testing set and the second row are images from venetian archives.

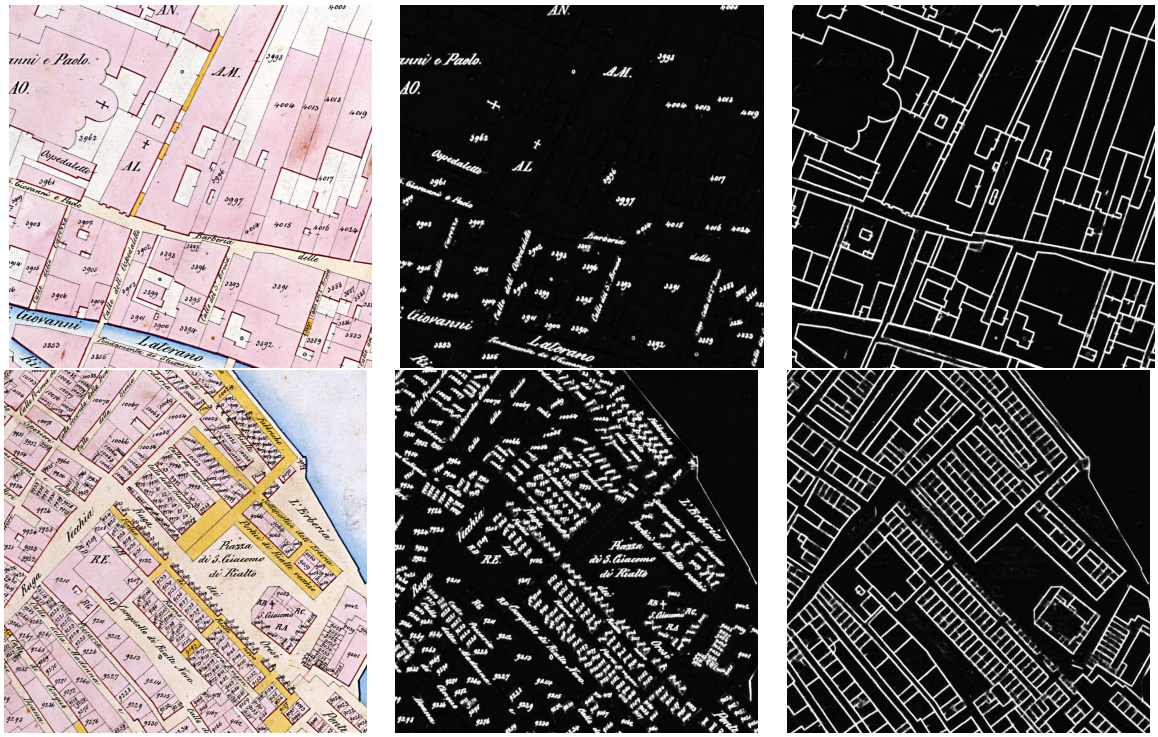


Figure 12: Mask of probabilities output by the segmentation model for text and parcel extraction on CAD dataset.