Recognition and Enrichment of Archival Documents

D6.11.
Line and Word Segmentation Tools P2

Georgios Louloudis, Nikolaos Stamatopoulos, Basilis Gatos, NCSR Demokritos

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
Contractual date of delivery	31/12/2017
Actual date of delivery	28/12/2017
Date of last update	21/12/2017
Deliverable number	D6.11
Deliverable title	Line and Word Segmentation Tools P2
Type	Demonstrator
Status & version	Public & version 1
Contributing WP(s)	WP6
Responsible beneficiary	NCSR
Other contributors	CVL, UPVLC, URO
Internal reviewers	UPVLC, EPFL
Author(s)	Georgios Louloudis, Nikolaos Stamatopoulos, Basilis Gatos NCSR
EC project officer	Martin Majek
Keywords	Text Line Segmentation, Word Segmentation

Table of Contents

Executive Summary

This deliverable reports on the achievements concerning the tasks of text line and word segmentation at the end of the second year of the READ project. Several research groups of the READ consortium contributed with their methods and present comparable experimental results using a variety of datasets. A subset of these datasets has been used for measuring the performance of state-of-the-art techniques in the framework of the ICDAR2017 international conference [Diem2017]. To this end, a comparison on the performance can be seen not only among the methods developed by the READ partners but also among recently developed methods from groups with strong background on the Document Analysis area of research spanning the entire globe. It is important to mention that the outcome of a text line segmentation method is represented using baselines rather than making use of polygons. This is in accordance with the work of [Romero2015] where it is mentioned that a very large amount of time is saved for the correction of the baselines produced by a text line segmentation method at the expense of a very small drop in HTR accuracy. Finally, it should be mentioned that the baseline evaluation protocol was developed by the URO READ partner and was used not only on the previously mentioned competition organized by several READ partners [Diem2017] but also on the "ICDAR2017 Competition on Layout Analysis for Challenging Medieval Manuscripts" [Simistira2017] which was also presented in the ICDAR 2017 conference and organized by a group which is not part of the READ consortium. To this end, it can be considered as a standard on the Document Image Analysis community.

1. Text Line Segmentation

One of the early tasks in a handwriting recognition system is the segmentation of a handwritten document image into text lines, which is defined as the process of defining the region of every text line on a document image. In most cases, the expected input to this module is a single column text region which is actually the output of the basic layout analysis module (task 6.2). For these cases, the effectiveness of the text line segmentation process is strongly related with the result of the layout analysis stage. However, there are also cases in the literature where the text line segmentation method is applied on the image without any prior information about the text regions. NCSR and UPVLC methods described below belong to the former case whereas URO and CVL methods belong to the latter case. At the same time, results of poor quality produced by the text line segmentation stage seriously affect the accuracy of the handwritten text recognition procedure. Several challenges exist on historical documents which should be addressed by a text line segmentation method. These challenges include: a) the difference in the skew angle between lines on the page or even along the same text line, b) overlapping and touching text lines, c) additions above the text line and d) deleted text. Figure 1.1 presents one example for each of these challenges.

Two main variations exist for representing the results of a text line segmentation method: i) using polygons that enclose the corresponding text lines and ii) using baselines i.e. a set of (poly)line segments which correspond to the imaginary lines on which the scribe writes the text. Figure 1.2 presents one example of each of the abovementioned representation variations.

As it was mentioned on the first year's report (D.6.10) since the baseline representation has the advantage of needing less time for correction and since according to [Romero2015] the baseline representation produces comparable results in terms of HTR accuracy with the polygon representation, for the next two years of the READ project baselines will be used to represent the results of a text line segmentation method.

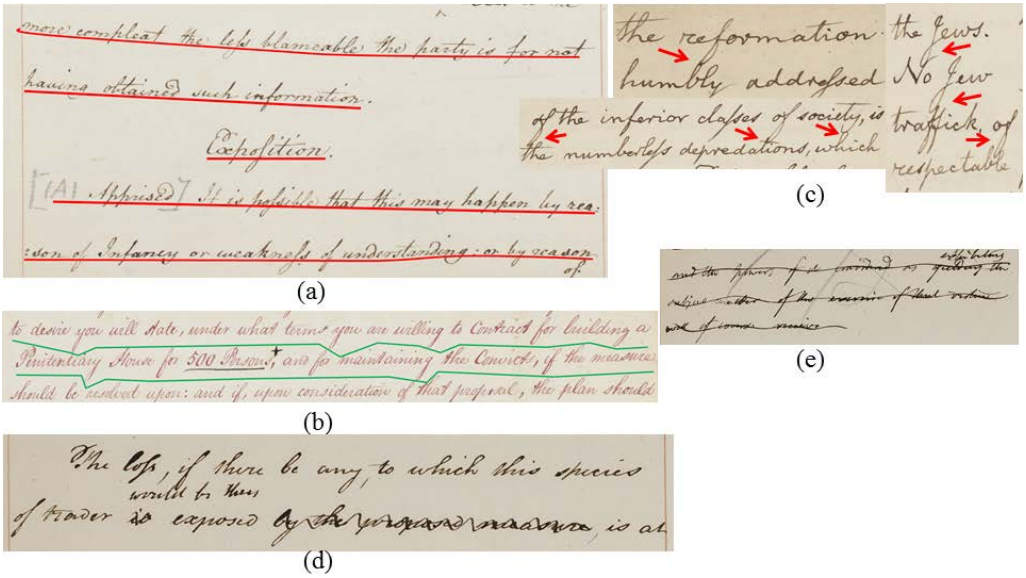


Figure 1.1: Challenges encountered on historical document images for text line segmentation: (a) Difference in the skew angle between lines on the page or even along the same text line, (b) overlapping text lines, (c) touching text lines, (d) additions above a text line, e) deleted text.

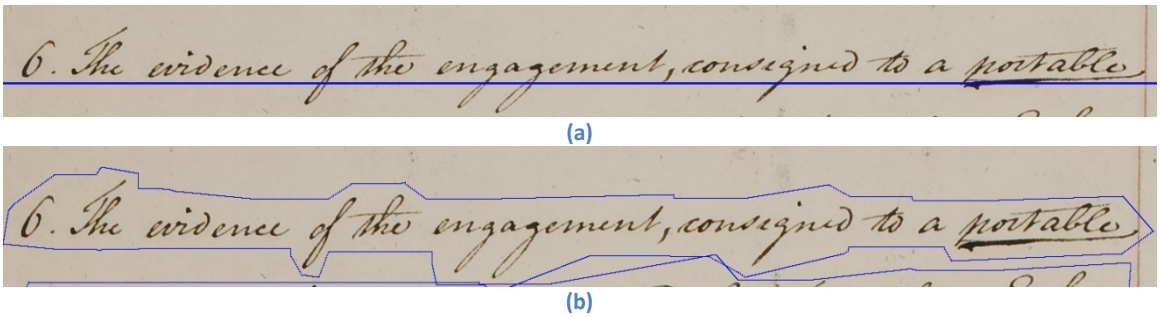


Figure 1.2: Representation of the text line segmentation result using (a) baseline and (b) polygon.

1.1. NCSR Text line Segmentation Method – 2nd Year

During year 2 of the project, the NCSR group worked on the basis of providing a more efficient method for text line segmentation in terms of accuracy and speed (NCSR (2nd year)). Since the focus of this task was given on the creation of a better baseline representation while neglecting the polygon representation which was the main focus of the previous year, NCSR group modified the algorithm in order to deal with these challenges. More specifically, the existing method was adapted to the nature and characteristics of historical handwritten documents. Several steps were reorganized in order to provide a text line segmentation result in the minimum time without affecting the accuracy. The main steps which were considered are summarized below.

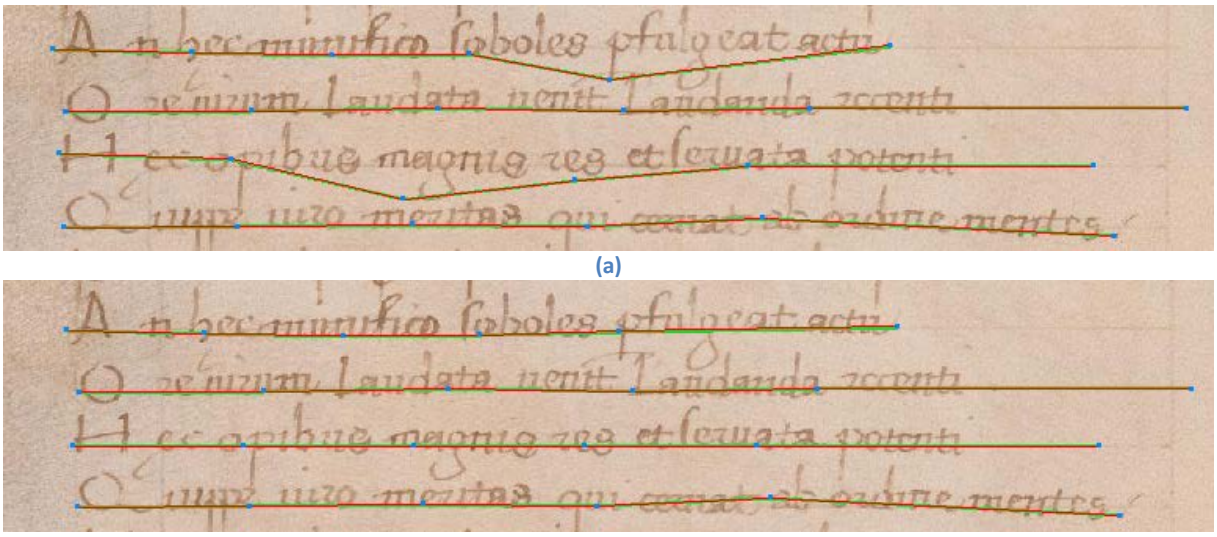
First of all, the algorithm was adapted to be able to work with tables. More specifically, since many documents contain tables, our algorithm was enriched with the ability to work on table cells and produce the corresponding text lines using the baseline representation. The

output of this procedure is an xml file using the updated PAGE format. Moreover, the polygon creation step was replaced by the production of baselines leading to a considerable reduction of the processing time.
After a careful error analysis on the results of the year 1 algorithm we noticed that the majority of the errors where produced due to the nature of the documents appearing on several collections of the READ datasets which contained mostly cursive handwriting with touching words and characters of adjacent text lines. Since the basic step of our algorithm (Hough transform step) was not considering points of interest coming from large components, we used the idea of [Zhang2014] and tried to keep the main body parts of the large components. Using this simple idea, more points of interest contribute on the Hough domain and thus the probability of a better detection of text lines is increased.
Finally, due to the presence of baselines with severe fluctuation on the results of the year 1 method, we developed a novel algorithm for smoothing the baselines which manages to solve the majority of the issues appearing on the results.
Some minor changes include the reorganization of the code for producing the PAGE xml output file. In our previous version, a new PAGE xml file was created containing only basic region information together with the text line representation produced by our algorithm. In the new version, the existing PAGE xml file is updated. In this way, very important information which existed in the PAGE xml (e.g. reading order, metadata information added by Transkribus users) is retained while at the same time the text line representation using baselines produced by our algorithm is appended to the corresponding text regions.
The effect of the abovementioned changes to the accuracy of the algorithm can be observed on the experimental results section. The increase in the accuracy compared with the year 1 method is clear. It should be noted that the method is totally unsupervised thus no training is involved.
The efficiency of the NCSR (2nd year) in terms of speed is demonstrated in table 1.1.1 in which we show the average processing time per document using several datasets. Moreover, we demonstrate the average time per document for the NCSR (1st year) method. The machine which was used to run the experiments was a desktop computer with 16GBytes of RAM and an Intel Core i7-4770k CPU @ 3.50GHz. Additional information on the datasets can be found in Section 1.6.

Table 1.1.1: Average processing time per document for NCSR methods.

	Average Processing Time (sec)	
Dataset	**NCSR (2nd year)**	**NCSR (1st year)**
Konzilsprotokolle (German)	0,7	5,2
NAF (Finnish)	0,9	8,4
BL (English)	1,1	8,9
cBAD - SIMPLE (train)	0,7	9,1
cBAD - SIMPLE (test)	0,8	5,9

Figure 1.1.1 demonstrates the application of the novel smoothing algorithm on fluctuating baselines.



(a)

(b)

Figure 1.1.1: Application of the novel smoothing algorithm on fluctuating baselines: (a) NCSR 1st year, (b) NCSR 2nd year.

Several additional tools have been developed during the second year of the READ project. All these tools are available at the github repository in the folder "NCSR_Tools" (it is a private repository on which only partners of the READ consortium have access). The first tool is under the folder "NCSR_AddBaselinesToPolygons". This tool enriches a PAGE xml file containing only text lines with polygon representation with their baseline representation. The baseline creation is achieved by linear regression on the lower pixels of connected components belonging to the text line. The second tool which appears at folder "NCSR_FromBaseLinesToPolylines" solves the inverse problem i.e. creates the polygon representation of text lines when only the baseline representation exists in the PAGE xml file. This tool is necessary for the upcoming word segmentation step which needs text lines using polygon representation in order to produce the word segmentation result. The polygon creation is achieved by grouping connected components to the closest baselines. Additionally, efficient separation of vertically connected characters is performed using a novel method based on skeletonization. The polygon creation procedure is based on an efficient algorithm which creates text line polygons with a small set of vertices [Retsinas2016].

As it was mentioned above, all NCSR developed tools during the second year of the READ project are developed in C++ and are available at the github repository of Transkribus following the guidelines of the Transkribus interface. Have in mind that these tools are stored under a private repository and are made available only to partners of the READ consortium. Finally, thanks to the EPFL partner, Cython bindings were created that allow calls to NCSR modules directly from python. The link to the NCSR private github folder is:

https://github.com/Transkribus/NCSR_Tools

1.2. URO Text line Segmentation Method – 2nd Year

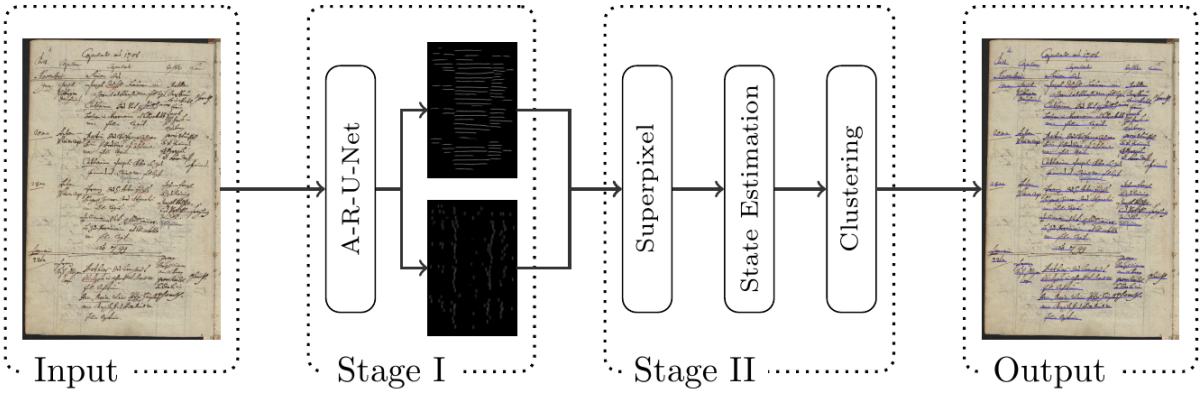


Figure 1.2.1: Shown is the two stage workflow of UROs text line segmentation method.

UROs text line segmentation method relies on a two stage process, see Fig. 1.2.1. The first stage performs a pixel labeling in means of a deep neural network. The proposed network (A-R-U-Net) is an extension of the well-known U-Net [Ronneberger2015]. Two additional concepts were integrated to improve the performance of the U-Net. First, residual blocks were introduced to increase the representative depth without strengthening the vanishing gradient problem. Second, an attention mechanism was designed and implemented to allow the architecture to "look" at different areas in an image at different resolutions. This network was trained in a purely supervised manner on the cBAD training set [Diem2017]. At inference time, the A-R-U-Net predicts two probability maps. One encodes the positions of baselines present in the image. The second predicts begin and end of text lines to enable the system to handle complex layouts, see Fig. 1.2.1.

The second stage is based on the work which was done in the first two years of READ and published in [Grüning2017a] and therefore it will not be described in detail here. Basically, the output of the A-R-U-Net is utilized to calculate reliable Super Pixels (SPs). The so-called states are estimated for the SPs. Given the states, the SPs are clustered to build baselines.

UROs text line segmentation method is already integrated and usable via Transkribus. The code is available on github: https://github.com/Transkribus/CITlabModule. The results are presented in Section 1.6. The average time per page on a dual core laptop (Intel Core i7-6600U) with 16GiB RAM ranges from 2s to 12s dependent on the image resolution.

1.3. CVL Text line Segmentation Method – 2nd Year

In contrast to recent advancements achieved by e.g. URO using deep learning for baseline extraction, the basic method developed at CVL is fully unsupervised. The strategy is to derive general rules and compute local statistics that indicate text orientation and text line spacing.

The method utilizes Super Pixels to localize potential text elements (see D.6.5). Having found potential text regions, a bottom-up clustering approach groups Super Pixels into text-line candidates. The parameters of the clustering (i.e. element distance and stop criteria) are tuned with local statistics such as the local text orientation and interline spacing (see D.6.4). Finally, we extract baselines with the help of Least Median Squares fitting of potential text line candidates.

The evaluation results show that the unsupervised method presented here cannot compete with state-of-the-art methods that were submitted to cBAD [Diem2017] or with supervised

methods such as the one developed by URO. Because of its unsupervised nature and high recall, we can use it in the context of different document domains and as basis of other processing stages such as text block segmentation. Since we designed this method rather as pre-processing stage for basic layout analysis then for direct HTR input, details and recent advancements are discussed in Deliverable D6.5.

1.4. UPVLC Text line Segmentation Method – 2nd Year

Two different text line segmentation methods were developed by UPVLC during the second year of the READ project, namely UPVLC_a (2nd year) and UPVLC_b (2nd year).

Concerning UPVLC_a (2nd year), the method was developed in order to deal with the *Oficio de Hipotecas de Girona* corpus (GIRONA dataset) and is composed by the following steps (Fig. 1.4.1):

A. Image-to-Image translation via Conditional Generative Adversarial Networks in order to:
   - Automatically eliminate all noise and perform adequate image transformations.
   - Detect and classify text pixel zones into text lines and/or text regions.
B. Topological structural component analysis in order to group and obtain region contours of classified pixels.
C. Basic line regression inside text line regions to obtain the main body line of the text line contours.

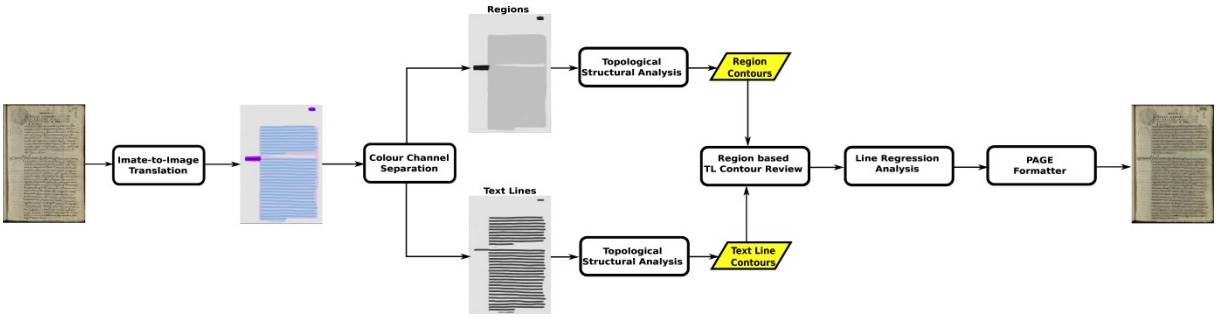


Figure 1.4.1: Process workflow of the *Oficio de Hipotecas de Girona system*.

On Fig. 1.4.2 we can observe a segment of a page image through the different stages of the process.

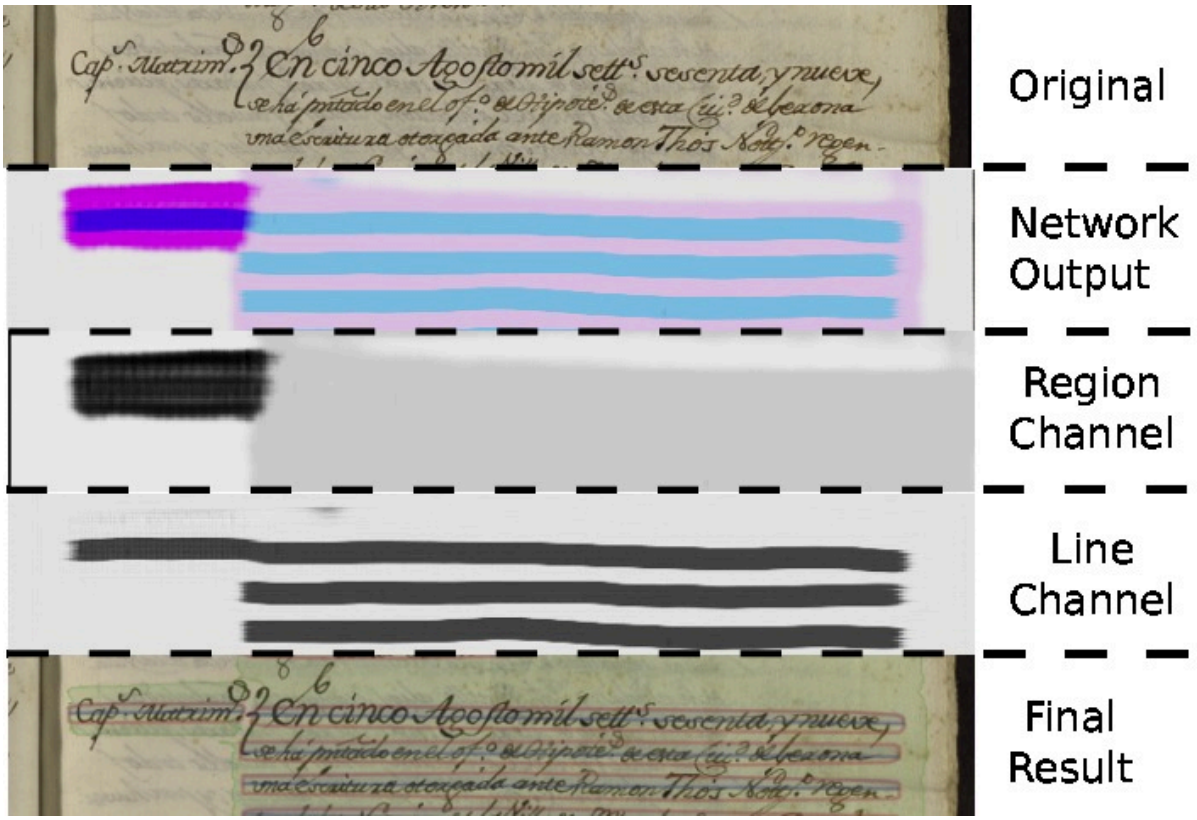


Figure 1.4.2: Visual representation of the impact of the different process steps on a page image segment.

Concerning UPVLC_b (2nd year), the method is composed by the UPVLC_a (2nd year) for the detection of the text regions. As soon as regions are detected, a new method described in [Fawzi2017] was developed for the detection of baselines.

1.5. Evaluation Protocol

The evaluation protocol used to measure the text line segmentation quality was developed in the first two years of READ. It is documented [Grüning2017b] and is planned to be submitted for publication. The tool is available via github:

https://github.com/Transkribus/TranskribusBaseLineEvaluationScheme.

It was recently used in two competitions organized under the ICDAR2017 international conference in Kyoto.

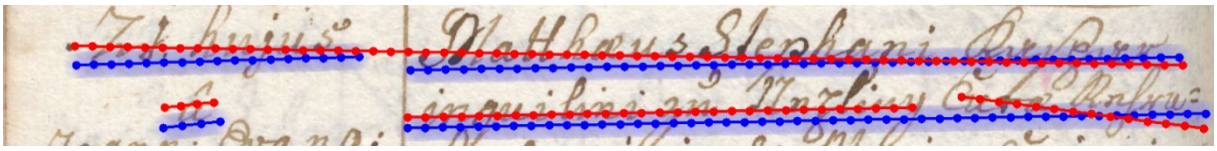


Figure 1.5.1: Groundtruth baselines (blue) along with their tolerance areas as well as hypothesis baselines (red). The evaluation protocol for this image snippet results in: R=0.91, P=0.61, F=0.73.

The basic idea is to calculate R-, P- and F-values, which are very similar to the well-known precision and recall values. The R-value should encode how many of the ground truth baselines were detected. The P-value indicates the quality (over-/under segmentation) of the results. The F-value is just the harmonic mean of the above mentioned values. Since there is not one correct baseline – a baseline is still correct if it is slightly different with respect to the ground truth baseline, because it still allows for entirely correct text

recognition – tolerance areas (Fig. 1.5.1 blue areas) were calculated for each ground truth baseline. With respect to these tolerance areas, the fractions of correctly detected ground truth baselines as well as the fraction of correct hypothesis baselines are estimated. These values lead to the above mentioned R-, P- and F-values. For more details, we refer to [Grüning2017b].

We would like to mention that there is a difference in the evaluation scheme which was used in the deliverable of the first year and the one used in this deliverable. In the first version of the evaluation scheme, the size of the tolerance area was determined by a user-defined parameter. This approach didn't take into account the resolution of the image nor the interline spacing. Therefore, the evaluation scheme was extended in order to calculate dynamically the size of the tolerance area for each image. This leads to different P-, R- and F-values for the same results. These differences could be quite big for high resolution images. Nevertheless, the new version of the tool will be used for further evaluation. The results reported on the deliverable of the first year are re-evaluated with the new version of the evaluation scheme.

1.6. Experimental Results

The performance of the all text line segmentation methods developed during the second year of the READ project by the corresponding partners together with the NCSR text line segmentation method (1st year) which was part of the year 1 deliverable (D.6.10) have been tested using five challenging datasets of historical handwritten documents: (i) Konzilsprotokolle (German), (ii) NAF (Finnish), (iii) BL (English), (iv) cBAD competition simple scenario (training + test subset) and (v) cBAD competition complex scenario (training + test subset). The UPVLC group also ran text line segmentation trials on the GIRONA corpus. Table 1.6.1 summarizes the number of documents together with the number of text lines and words for each dataset.

Table 1.6.1: Summary of dataset information used to evaluate the text line and word segmentation methods.

Dataset	#documents	#text lines	#words
Konzilsprotokolle (German)	100	2555	15567
NAF (Finnish)	56 (double pages)	3186	16201
BL (English)	115	2971	15739
cBAD - SIMPLE (train)	216	6379	-
cBAD - SIMPLE (test)	539	14735	-
cBAD - COMPLEX (test)	1010	88962	-
GIRONA	350	13647	-

Tables 1.6.2 – 1.6.7 present comparative experimental results for each dataset using the baseline evaluation protocol in terms of (R)ecall, (P)recision, and (F)-measure. All methods are sorted with respect to the F-value. In addition, the number of Ground-Truth as well as the number of Result lines are presented for each dataset and method referred as # GT lines and #RS lines, respectively.

Table 1.6.2: Comparative experimental results using Konzilsprotokolle dataset

Method	# GT lines	# RS lines	P	R	F
URO (2nd year)	2555	2574	**99,14**	**99,76**	**99,45**
NCSR (1st year)		2532	96,96	98,08	97,52
NCSR (2nd year)		2459	97,30	96,16	96,72
UPVLC_b (2nd year)		2488	97,90	94,30	96,10
CVL (2nd year)		2642	91,33	91,84	91,58

Table 1.6.3: Comparative experimental results using NAF dataset

Method	# GT lines	# RS lines	P	R	F
URO (2nd year)	3186	3238	97,00	**98,64**	**97,82**
NCSR (2nd year)		3083	97,00	96,59	96,80
NCSR (1st year)		3053	96,45	95,31	95,88
UPVLC_b (2nd year)		2953	**97,80**	91,80	94,80
CVL (2nd year)		3308	89,66	91,99	90,81

Table 1.6.4: Comparative experimental results using BL dataset

Method	# GT lines	# RS lines	P	R	F
URO (2nd year)	2961	3019	**94,86**	**96,99**	**95,91**
NCSR (2nd year)		2708	**94,85**	92,95	93,89
NCSR (1st year)		2889	91,63	94,83	93,20
UPVLC_b (2nd year)		2845	91,67	91,10	91,40
CVL (2nd year)		2879	87,98	83,45	85,65

Table 1.6.5: Comparative experimental results using cBAD training dataset (Simple Scenario)

Method	# GT lines	# RS lines	P	R	F
NCSR (2nd year)	6379	6151	**88,66**	**90,97**	**89,80**
NCSR (1st year)		5067	86,99	77,75	82,11
CVL (2nd year)		-	-	-	-
URO (2nd year)		-	-	-	-
UPVLC_b (2nd year)		-	-	-	-

Table 1.6.6: Comparative experimental results using cBAD test dataset (Simple Scenario)

Method	# GT lines	# RS lines	P	R	F
URO (2nd year)	14735	14673	**97,50**	**98,04**	**97,77**
NCSR (2nd year)		14496	89,54	91,57	90,54
UPVLC_b (2nd year)		14370	93,70	85,50	89,40
NCSR (1st year)		12496	87,79	81,72	84,64
CVL (2nd year)		22578	61,00	88,00	72,00

Table 1.6.7: Comparative experimental results using cBAD test dataset (Complex Scenario)

Method	# GT lines	# RS lines	P	R	F
URO (2nd year)	88962	87363	**92,59**	**92,02**	**92,30**
UPVLC_b (2nd year)		40011	83,30	60,60	70,20
CVL (2nd year)		61917	52,00	78,00	62,00

Finally, in order to present a comparison of the text line segmentation methods developed by the READ partners during the second year of the project, we add tables 1.6.8 and 1.6.9 which contain the results of the methods participated to the cBAD competition (Simple and Complex Scenario, respectively). It can be observed that URO's text line segmentation method is ranked first among all methods which used the cBAD test set of the simple scenario while NCSR method is ranked third. Have in mind that NCSR method is totally unsupervised whereas the two first techniques make use of a deep learning (supervised) approach.

Table 1.6.8: Comparative experimental results of methods participated to the cBAD competition (Simple Scenario - Test Set)

Method	P	R	F
DMRZ	**97,30**	**97,00**	**97,10**
UPVLC_b (2nd year)	93,70	85,50	89,40
BYU	87,80	90,70	89,20
IRISA	88,30	87,70	88,00
LITIS	78,00	83,60	80,70

Table 1.6.9: Comparative experimental results of methods participated to the cBAD competition (Complex Scenario - Test Set)

Method	P	R	F
DMRZ	**85,40**	**86,30**	**85,90**
BYU	77,30	82,00	79,60
IRISA	69,20	77,20	73,00
UPVLC_b (2nd year)	83,30	60,60	70,20

Experiments ran by UPVLC using UPVLC_a (2nd year) method on the GIRONA dataset are demonstrated in table 1.6.11. The experiments have been executed on batches of 50 pages. At a next step, each batch's results are reviewed and corrected manually and used to train models for the next batch. A summary of the batches of the GIRONA dataset used for evaluation is presented in table 1.6.10.

Table 1.6.10: Summary of the batches of the GIRONA dataset

Batch	#Lines
b004	1960
b005	1985
b006	1978
b0007	1762
b008	1963
b009	1976
b010	2023

Table 1.6.11: Experimental results using the GIRONA dataset

Train Data	Test Data	P	R	F
b004	b005	97,43	96,39	96,91
b004-b005	b006	97,82	96,39	97,10
b004-b006	b0007	97,07	96,86	96,97
b004-b007	b008	97,72	95,91	96,81
b004-b008	b009	97,78	95,93	96,85
b004-b009	b010	97,42	95,08	96,24

2. Word Segmentation

Word segmentation refers to the process of defining the word regions of a text line. Since nowadays most handwriting recognition methods assume text lines as input, the word segmentation process is usually necessary only for segmentation-based query by example (QbE) keyword spotting (KWS) methods. Segmentation of historical handwritten document images still presents significant challenges and it is an open problem. These challenges include the appearance of skew along a single text line, the existence of slant, the non-uniform spacing of words as well as the existence of punctuation marks (Figure 2.1).

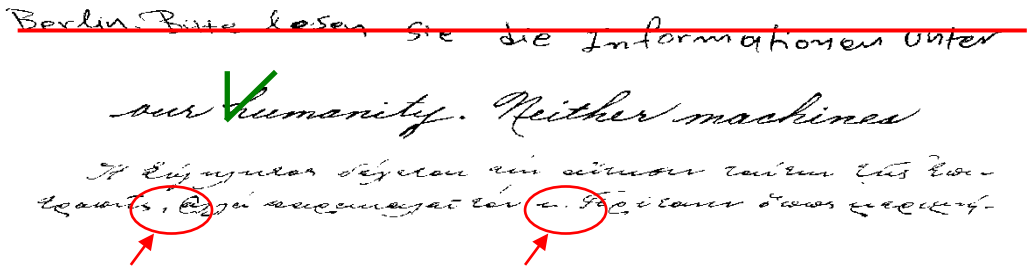


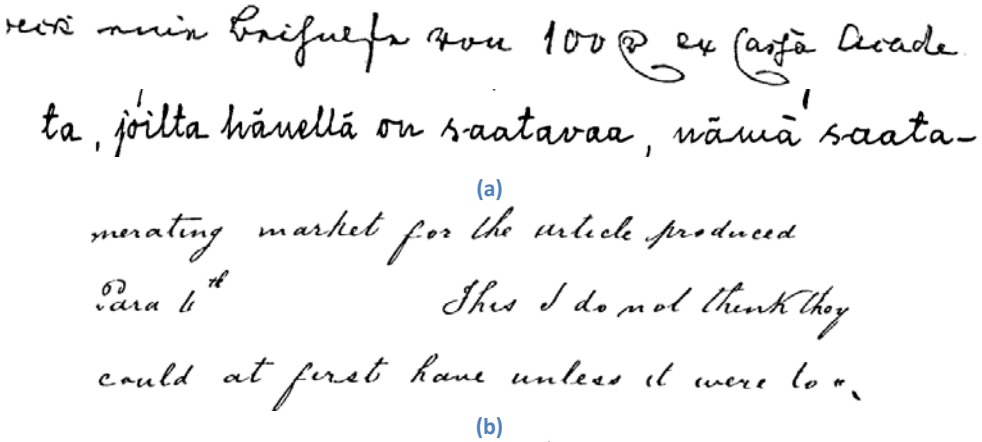
Figure 2.1: Challenges encountered on historical document images for word segmentation.

2.1. NCSR Word Segmentation Method – 2nd Year

In the frame of "READ" project a word segmentation method (NCSR 1st Year) was delivered in the first year. This method was an extension of the method presented in [Louloudis2009], adapted to historical handwritten documents and it contains two steps. The first step deals with the computation of the Euclidean distances of adjacent components in the text line image and the second step concerns the classification of the previously computed distances as either inter-word gaps or intra-words distances using the Gaussian Mixture Modeling clustering technique.

After a careful error analysis on the results of the developed method for the first year (D6.10), we noticed that the majority of the errors encountered were due to:

i. Incorrect calculation of the distance of adjacent words due to the existence of long ascenders/descenders as well as punctuation marks (Figure 2.1.1(a)).
ii. Inaccurate classification of the distances due to presence of extreme values/outliers (e.g. large distances of adjacent words) (Figure 2.1.1(b)).

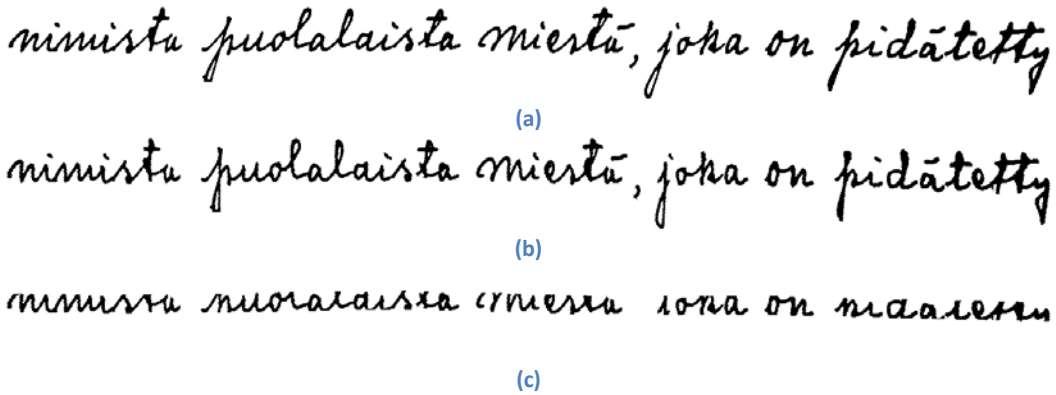


(a)

(b)

Figure 2.1.1: Majority of the errors encountered from the NCSR 1st Year method: (a) Incorrect calculation of the distance of adjacent words; (b) inaccurate classification of the distances due to presence of outliers.

Taking into account the above mentioned observations, we provided a more reliable word segmentation method (NCSR 2nd Year) in order to cope with these challenges. Concerning the distance computation stage, a main zone detection procedure is added in order to exclude the ascenders/descenders as well as the punctuation marks. Baseline information provided by the text line segmentation procedure was used in order to define the main zone (see Figure 2.1.2). Moreover, concerning the distance classification step, we replaced the Gaussian distribution with the Student's-t distribution. The main advantage of the Student's-t distribution concerns its robustness to the existence of outliers.



(a) (b) (c)

Figure 2.1.2: (a) Original text line image; (b) after slant correction; (c) after main zone detection.

It should be stressed that the NCSR word segmentation method is developed in C++ following the guidelines of the Transkribus interface and it is available at github:

https://github.com/Transkribus/NCSR_Tools

2.2. Evaluation

The performance of the NCSR method (1st and 2nd year) as well as the sequential clustering method [Kim2001] has been tested on three challenging datasets of historical handwritten documents: (i) Konzilsprotokolle (German), (ii) NAF (Finnish) and (iii) BL (English). Table 1.6.1 summarizes the number of documents as well as the number of words for each dataset.

For the evaluation of the word segmentation methods we follow the same protocol which was used in the ICDAR 2013 Handwriting Segmentation Competition [Stamatopoulos2013]. An analytic description of the protocol is provided in the corresponding deliverable of the first year "D6.10 Line and Word Segmentation Tools P1". Tables 2.2.1 - 2.2.3 present comparative experimental results for each dataset in terms of Precision (PP), Recall (PR), and F-Measure (PFM).

Table 2.2.1: Comparative experimental results using Konzilsprotokolle dataset

Method	# GT words	# RS words	#o2o	PP	PR	PFM
NCSR (2nd year)	15567	15310	13532	88.39	86.93	87.65
NCSR (1st year)		16252	12587	77.45	80.86	79.12
Sequential Clustering		11418	8325	72.91	53.48	61.70

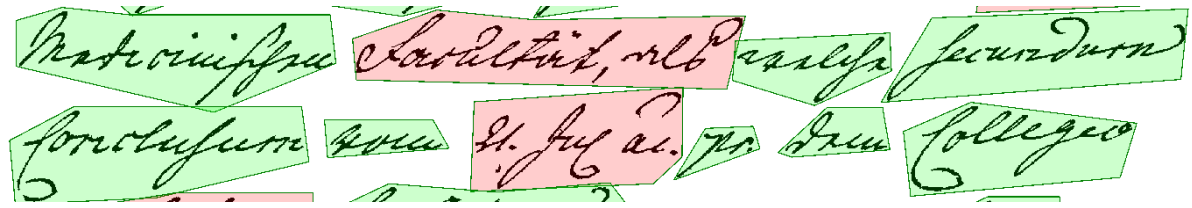
Table 2.2.2: Comparative experimental results using NAF dataset

Method	# GT words	# RS words	#o2o	PP	PR	PFM
NCSR (2nd year)	16201	19035	13245	68.58	81.75	75.18
NCSR (1st year)		20045	13404	66.87	82.74	73.96
Sequential Clustering		13200	10168	77.03	62.76	69.17

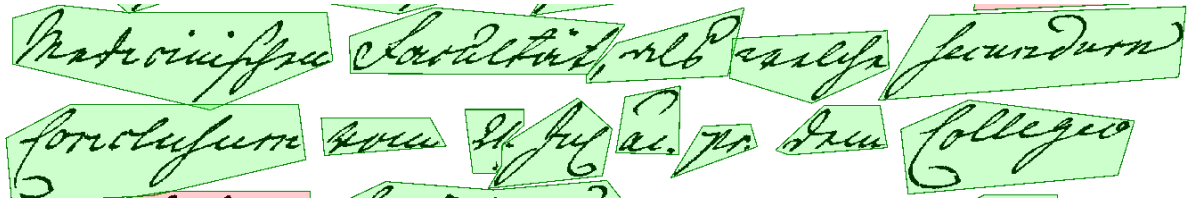
Table 2.2.3: Comparative experimental results using BL dataset

Method	# GT words	# RS words	#o2o	PP	PR	PFM
NCSR (2nd year)	15739	15243	11011	72.24	69.96	71.08
NCSR (1st year)		16908	11128	65.81	70.70	68.17
Sequential Clustering		11858	8049	67.88	51.14	58.33

As the experimental results indicate, the new NCSR 2nd year method outperforms both NCSR 1st year and the sequential clustering methods on all datasets and it achieves the highest F-Measure on Konzilsprotokolle dataset (87.65%) in which the increase in performance is about 9%. Concerning the NAF and BL datasets, the increase in performance is not extremely high since a lot of errors have been produced due to the presence of ditto marks or broken characters as a result of the binarization procedure. A representative result using a document of the Konzilsprotokolle dataset is presented in Figure 2.2.1. The NCSR 1st year method produced two segmentation errors which have been corrected by the new segmentation method.



(a)



(b)

Figure 2.2.1: Indicative results of the NCSR (a) 1st year and (b) 2nd year methods on Konzilsprotokolle dataset in which errors are depicted with red polygons.

Since the word segmentation process is usually necessary only for segmentation-based query by example keyword spotting methods, we are currently working on the scenario of providing multiple hypothesis segmentation results (Figure 2.2.2) in order to increase the number of correctly segmented words.

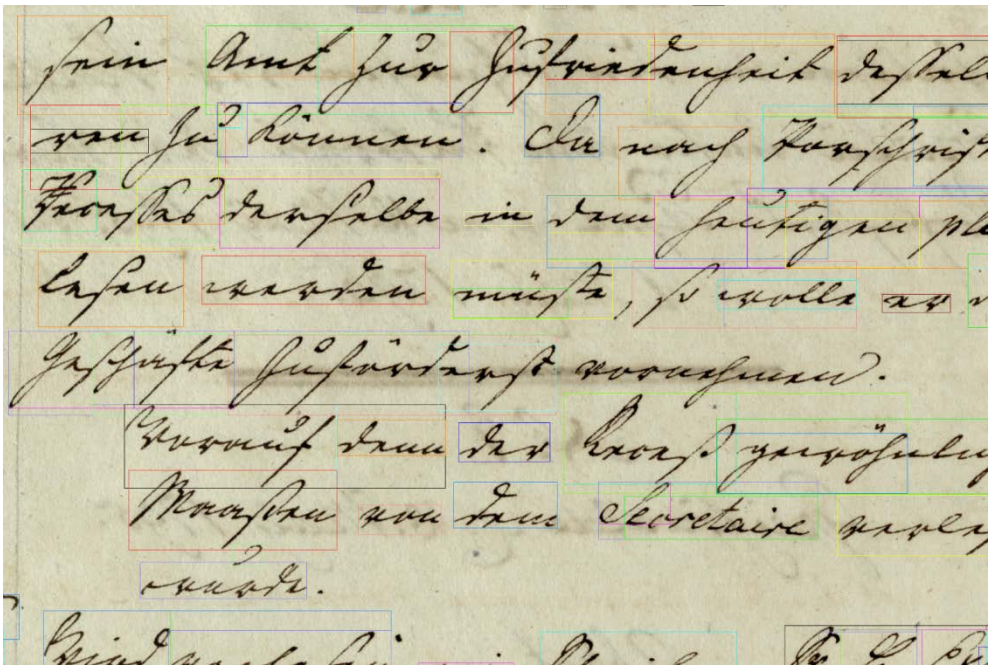


Figure 2.2.2: An example of a multiple hypothesis word segmentation result.

3. References

[Gatos2014] B. Gatos, G. Louloudis and N. Stamatopoulos “Segmentation of Historical Handwritten Documents into Text Zones and Text Lines”, 14th International Conference on Frontiers in Handwriting Recognition (ICFHR'14), pp. 464-469, 2014.

[Gatos2015] http://transcriptorium.eu/pdfs/deliverables/tranScriptorium-D3.2.2-31August2015.pdf

[Gruning2016] https://github.com/Transkribus/TranskribusBaseLineMetricTool

[Kim2001] S.H. Kim, S. Jeong, G.S. Lee and C.Y. Suen, "Word segmentation in handwritten Korean text lines based on gap clustering techniques", 6th International Conference on Document Analysis and Recognition (ICDAR’01), pp. 189-193, 2001.

[Louloudis2009] G. Louloudis, B. Gatos, I. Pratikakis, C. Halatsis, "Text line and word segmentation of handwritten documents", Pattern Recognition, vol. 42, no 12, pp. 3169-3183, 2009.

[Retsinas2016] G. Retsinas, G. Louloudis, N. Stamatopoulos and B. Gatos, “Efficient Document Image Segmentation Representation by Approximating Minimum-Link Polygons”, 12th Workshop on Document Analysis Systems (DAS'16), pp. 293-298, 2016.

[Romero2015] V. Romero, J.A. Sanchez, V. Bosch, K. Depuydt, and J. de Does, “Influence of text line segmentation in handwritten text recognition”, 13th International Conference on Document Analysis and Recognition, pp. 536-540, 2015.

[Stamatopoulos2013] N. Stamatopoulos, G. Louloudis, B. Gatos, U. Pal and A. Alaei, “ICDAR2013 Handwriting Segmentation Contest”, International Conference on Document Analysis and Recognition (ICDAR’13), pp. 1402-1406, 2013.

[Ronneberger2015] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in MICCAI, pp. 234–241, Springer, 2015.

[Grüning2017a] Grüning, Tobias and Leifert, Gundram and Strauß, Tobias and Labahn, Roger: A Robust and Binarization-Free Approach for Text Line Detection in Historical Documents. In Proceedings of the 14th International Conference on Document Analysis and Recognition (2017), 236-241.

[Grüning2017b]Tobias Grüning, Roger Labahn, Markus Diem, Florian Kleber, Stefan Fiel: “READ-BAD: A New Dataset and Evaluation Scheme for Baseline Detection in Archival Documents”, technical report, https://arxiv.org/abs/1705.03311.

[Diem2017] Markus Diem, Florian Kleber, Stefan Fiel, Tobias Grüning, and Basilis Gatos: “cBAD: ICDAR2017 Competition on Baseline Detection”, in Proceedings of the 14th International Conference on Document Analysis and Recognition (2017), 1355-1360.

[Simistira2017] Fotini Simistira, Manuel Bouillon, Mathias Seuret, Marcel Würsch, Michele Alberti, Rolf Ingold, and Marcus Liwicki: “ICDAR2017 Competition on Layout Analysis for Challenging Medieval Manuscripts”, in Proceedings of the 14th International Conference on Document Analysis and Recognition (2017), 1361-1370.

[Zhang2014] Xi Zhang and Chew Lim Tan: “Text Line Segmentation for Handwritten Documents Using Constrained Seam Carving”, in 14th International Conference on Frontiers in Handwriting Recognition (ICFHR'14), pp. 98-103, 2014.

[Fawzi2017] Ahmed Fawzi, Moises Pastor, and Carlos D. Martinez-Hinarejos. Baseline detection on arabic handwritten documents. In Proceedings of the 2017 ACM Symposium on Document Engineering, DocEng '17, pages 193-196, New York, NY, USA, 2017. ACM.