Bleed-through Removal by Learning a Discriminative Color Channel

Mauricio Villegas and Alejandro H. Toselli

mauvilsa@upv.es ahector@prhlt.upv.es

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

tranScriptorium

Slides available at:
http://mvillegas.info/pub/Villegas14_ICFHR_Bleed-through_presentation.pdf

Outline

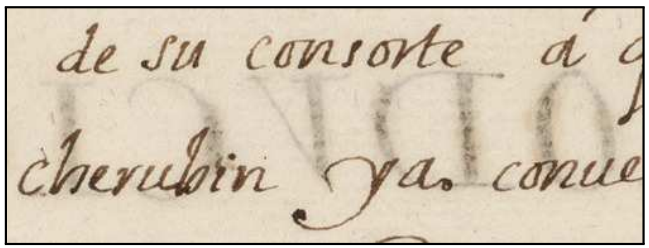
Introduction

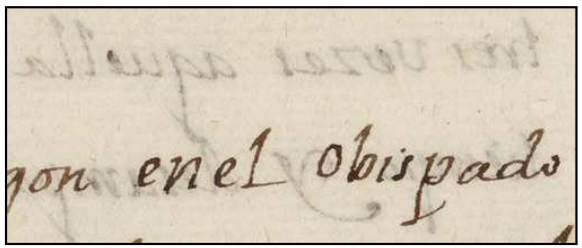
Removal of noise is an important step for improving the performance of handwritten recognition systems.

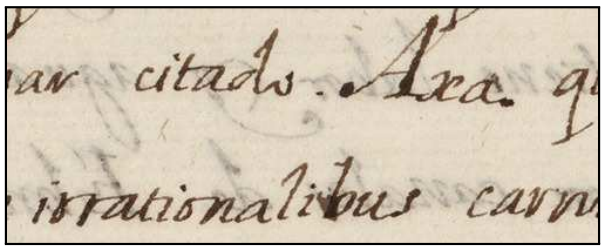
One type of noise specific of scanned document images is the appearance of content from the reverse side of the pages, commonly known as *bleed-through*:

- Due to the seeping of ink from the reverse side.
- Due to the transparency of the paper.

Example images:







Introduction

- Many bleed-through removal methods have been proposed in the literature.
- In general it is difficult for a model/method to work well for any document.
  - High variability: color of the paper and ink, degradation due to age, characteristics of the scanner, etc.
- What to do when a method fails? Easy, more training data and adjust parameters!

  But parameters are generally very cryptic and not easy to modify by end users!

Motivation and Proposed Approach

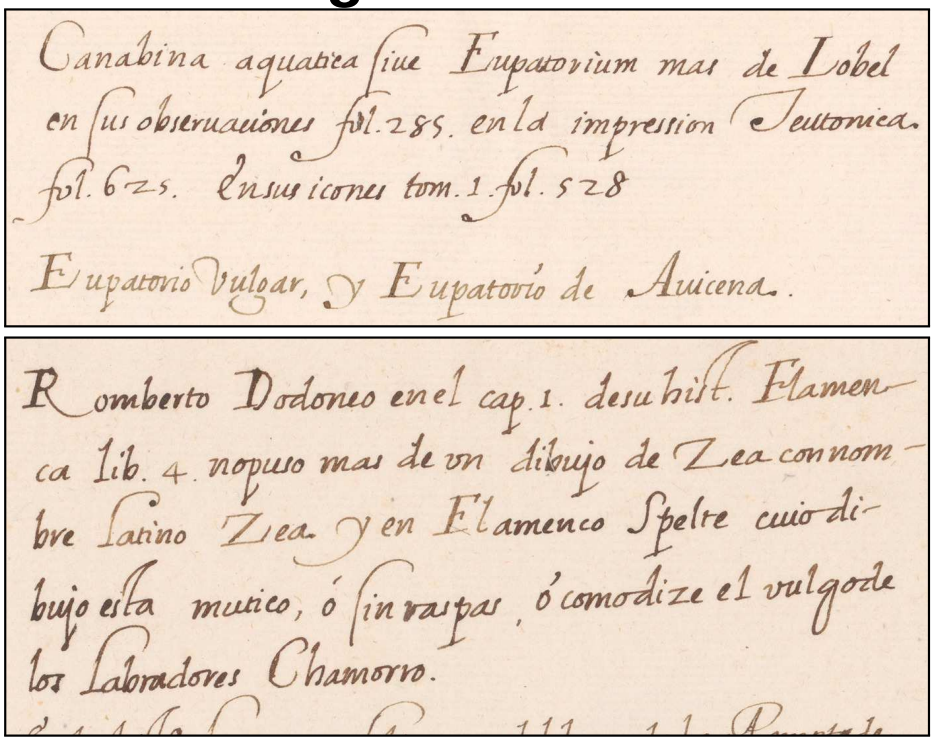
- When high quality is required, transcriptions are done by people, and nowadays with the help of emerging interactive (assisted) text recognition systems.
- In interactive systems the objective is to reduce the effort required by the users to transcribe.
- **Bleed-through removal approach:** Define a simple task for the user that allows the system to learn to discriminate noise. Has the advantage that can be done for every collection to transcribe.
- **User task:** Select regions of clean text and bleed-through.

Selection of training regions

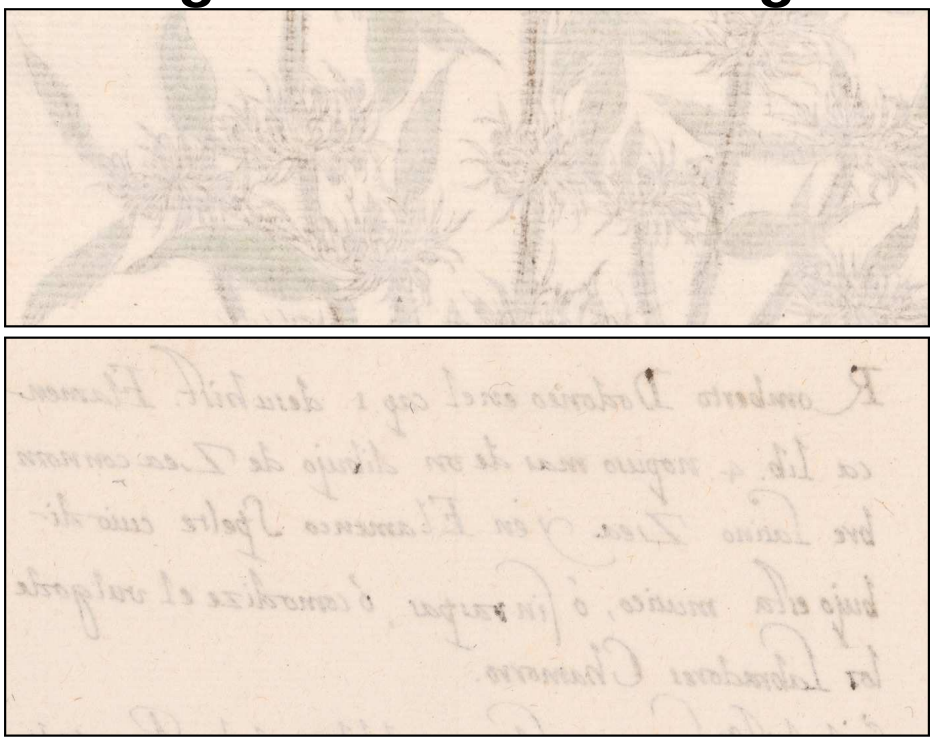
From a few example pages the user selects regions of clean text and regions where bleed-through is visible.

From each region many overlapping training patches are extracted.

Regions of text

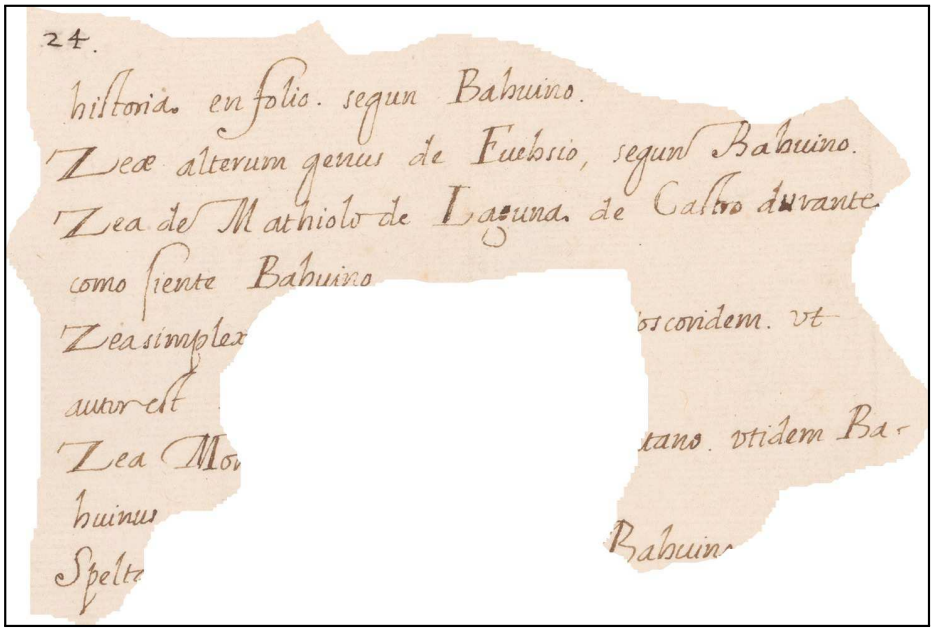


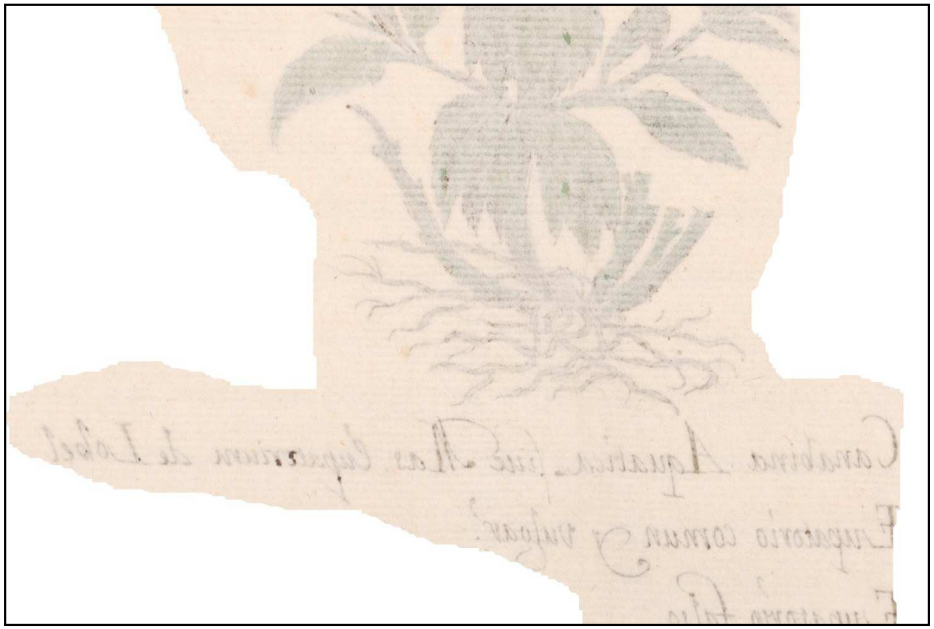
Regions of bleed-through

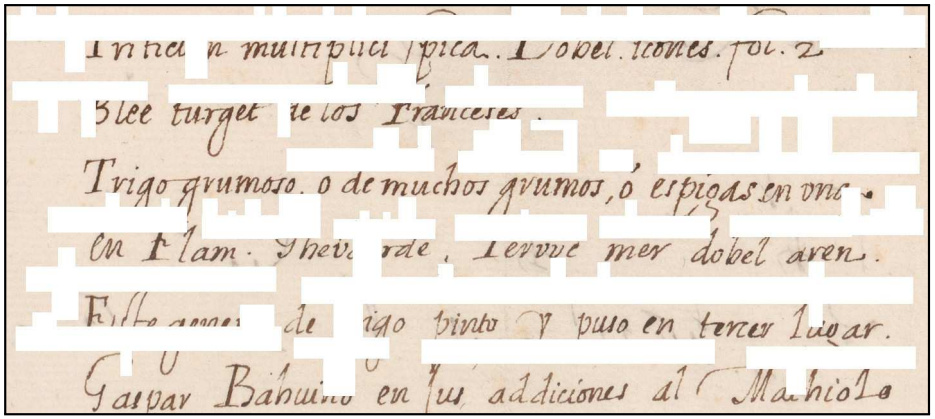


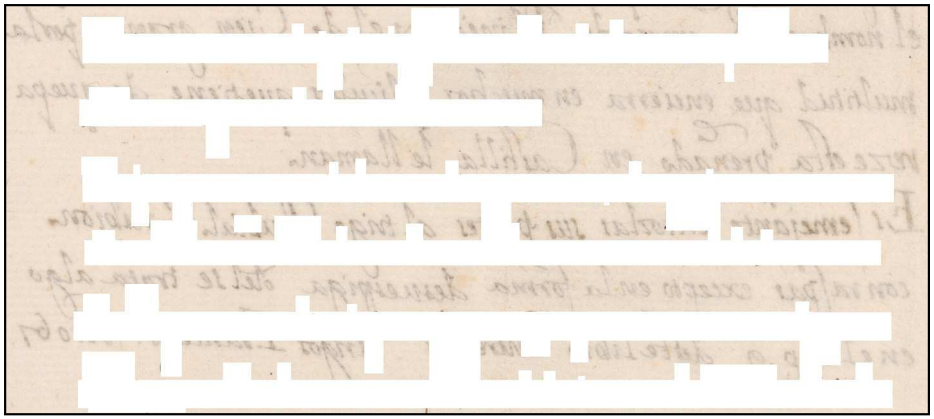
Selection of training regions (cont.)

More complex regions can be for example defined by curves or computer assisted based on text line detectors.









Color Channel Learning

Objective: Parametrized pixel transformation function

$$f_{\boldsymbol{\theta}} : \mathbb{R}^C \rightarrow \mathbb{R}$$

optimized so that it maximizes the ratio of expected patch variances (clean text patches / bleed-through patches)

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \frac{\mathsf{E}\left[\mathrm{var}(\boldsymbol{f}_{\boldsymbol{\theta},\mathsf{x}})\right]}{\mathsf{E}\left[\mathrm{var}(\boldsymbol{f}_{\boldsymbol{\theta},\mathsf{y}})\right]}$$

Considered the following family of transformation functions

$$f_{\boldsymbol{b}}(\boldsymbol{p}) = [\boldsymbol{g}(\boldsymbol{p})]^{\mathsf{T}} \boldsymbol{b}$$

and for the mapping function $\boldsymbol{g} : \mathbb{R}^C \rightarrow \mathbb{R}^D$, in this work it was tried linear, and general 2nd and 3rd order.

Color Channel Learning

Objective: Parametrized pixel transformation function

$$f_{\boldsymbol{\theta}} : \mathbb{R}^C \rightarrow \mathbb{R}$$

optimized so that it maximizes the ratio of expected patch variances (clean text patches / bleed-through patches)

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \frac{\mathsf{E}\left[\mathrm{var}(\boldsymbol{f}_{\boldsymbol{\theta},\mathrm{x}})\right]}{\mathsf{E}\left[\mathrm{var}(\boldsymbol{f}_{\boldsymbol{\theta},\mathrm{y}})\right]}$$

Considered the following family of transformation functions

$$f_{\boldsymbol{b}}(\boldsymbol{p}) = [\boldsymbol{g}(\boldsymbol{p})]^{\mathsf{T}} \boldsymbol{b}$$

and for the mapping function $\boldsymbol{g} : \mathbb{R}^C \rightarrow \mathbb{R}^D$, in this work it was tried linear, and general 2nd and 3rd order.

Color Channel Learning (cont.)

Solution: Largest generalized eigenvalue λ of

$$\boldsymbol{H}_{\mathsf{x}}\boldsymbol{b} = \boldsymbol{H}_{\mathsf{y}}\boldsymbol{b}\lambda$$

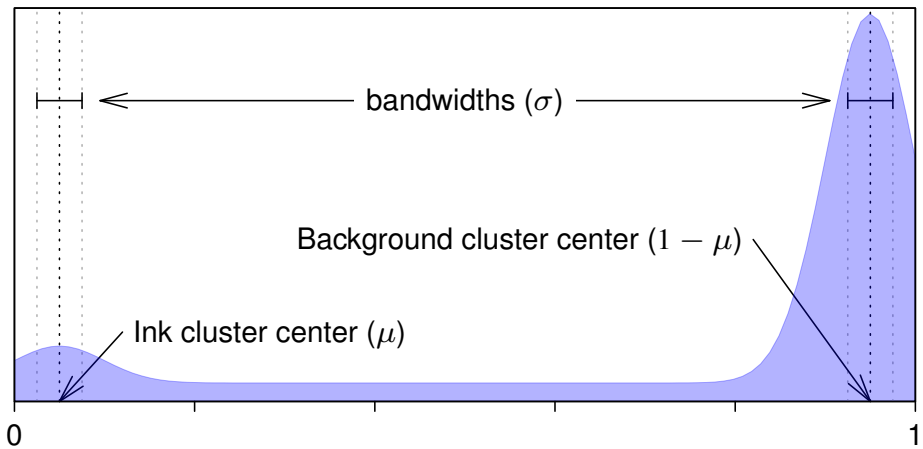
Let the rows of matrix $\boldsymbol{G}_{\mathsf{z}}$ be the output of function $\boldsymbol{g}$ for each of the pixels of the patch z. Then

$$\boldsymbol{H}_{\mathsf{z}} = \frac{1}{|\mathcal{Z}|N} \sum_{\forall z \in \mathcal{Z}} \boldsymbol{G}_z^{\mathsf{T}} \left(\boldsymbol{I} - \frac{\mathbf{1}_{N \times N}}{N} \right) \boldsymbol{G}_z$$

for N the number of pixels in a patch and $\mathcal{Z}$ be either the set of text ($\mathcal{X}$) or bleed-through ($\mathcal{Y}$) training patches.

Discretization and Gamma Correction

- Discretization range based on the values obtained for the clean text training patches.
- Resulting training text pixel value histogram used to guaranty that background corresponds to white and text to black.
- Gamma correction based on least Jensen-Shannon divergence between the histogram of text training patches and a prototypical histogram.



$$y(x) = \frac{\overbrace{\mathcal{N}(x,\mu,\sigma) + A\,\mathcal{N}(x,1-\mu,\sigma) + B}^{z(x)}}{\int_0^1 z(x')\,dx'}$$

Evaluation

Approach evaluated by observing the effect on the performance of a handwritten text recognition (HTR) system.

- Preprocessing:
  - **Bleed-through removal or conversion to grayscale.**
  - Generic noise removal (keeps grayscale information).
  - Slope correction.
  - Size normalization.
- Feature extraction:
  - Sequence of 60-dimensional feature vectors.
- Recognition:
  - Gaussian mixture HMMs.
  - Bi-gram language model with Kneser-Ney back-off smoothing.
  - Viterbi decoding.

Dataset and Experimental Setup

- Dataset: Prologue chapter of *Historia de Las Plantas* by Bernardo de Cienfuegos (17th century).

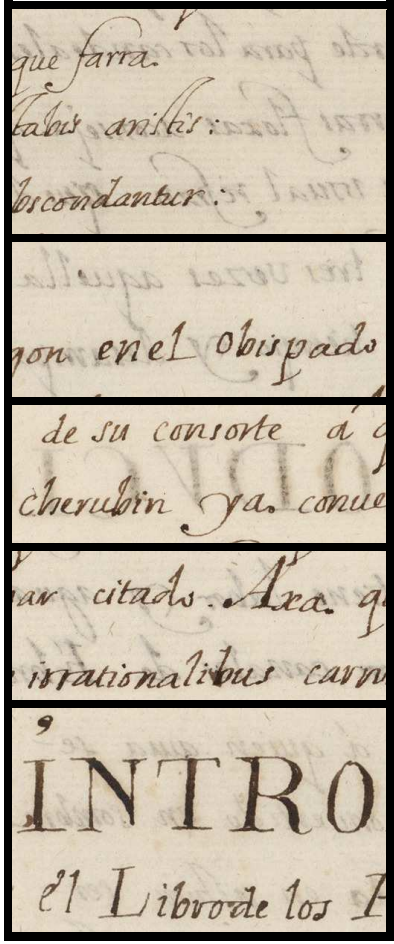
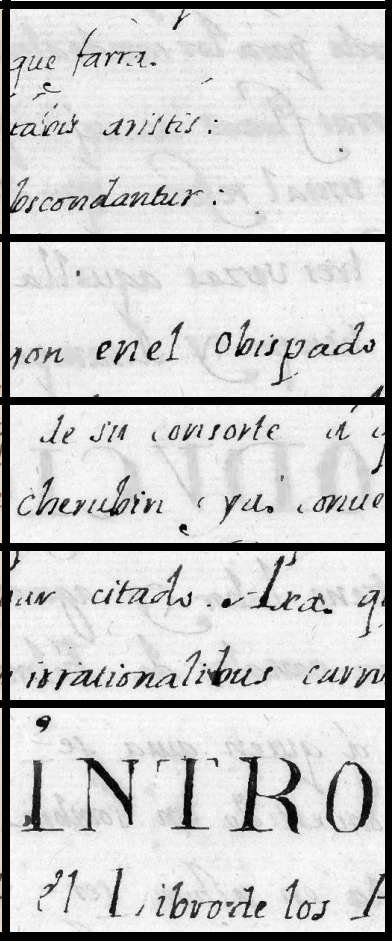
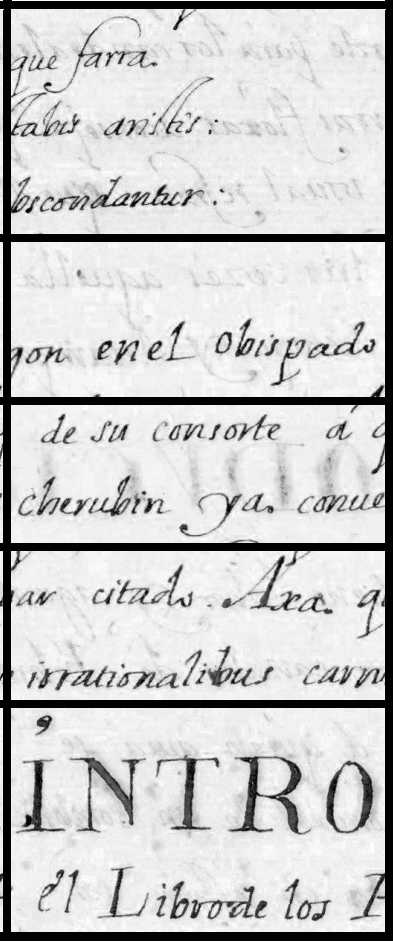
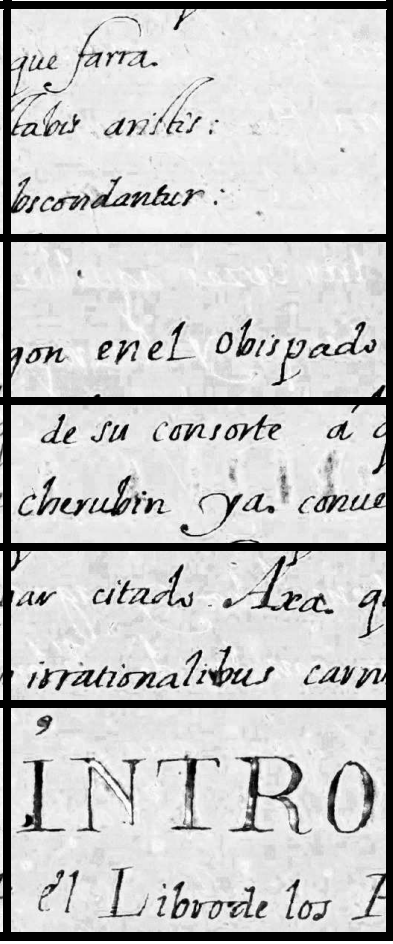
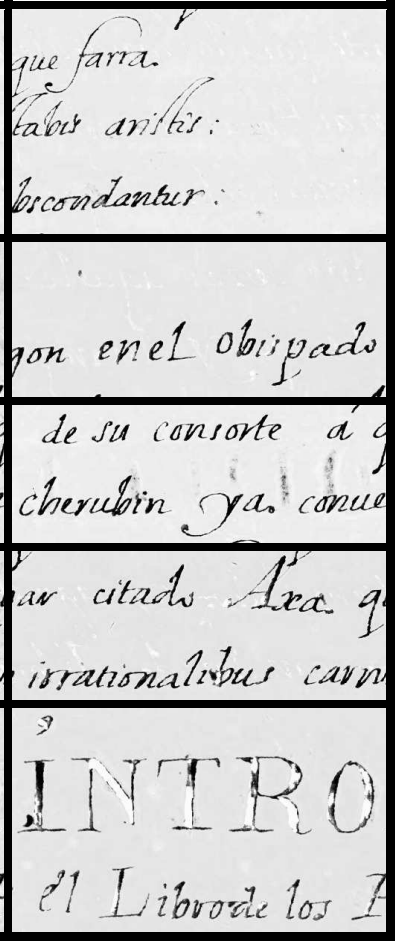
Num. Pages	38
Num. Lines	1,206
Running Words	11,642
Lexicon	3,899
Running Chars	61,973
Num. Chars	71

Training regions from other chapters, 11 clean text and 9 bleed-through.

- 10-fold cross-validation.
- Proposed technique (LDCC) for linear, 2^{nd} and 3^{rd} order models.
- Baseline techniques: 1) Grayscale, 2) Double MRF (Random Markov Fields) [Wol10].

[Wol10] Christian Wolf. "Document Ink Bleed-Through Removal with Two Hidden Markov Random Fields and a Single Observation Field". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32.3 (2010), pp. 431–447. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2009.33.

Resulting image examples

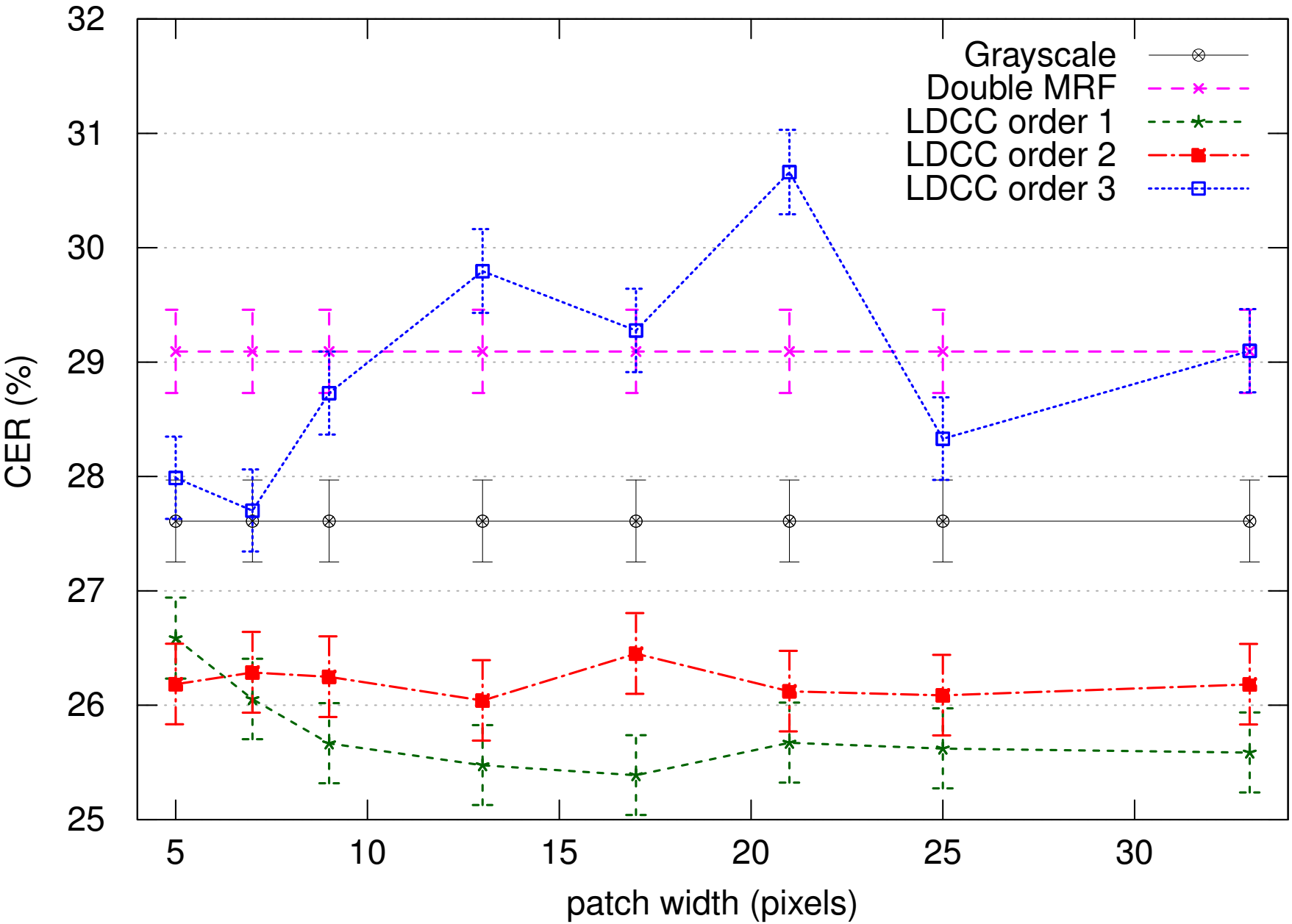
Original	Double MRF	LDCC ord. 1	LDCC ord. 2	LDCC ord. 3
				

CER performance comparison

Method	CER (%)	95% Conf. Int.
Grayscale	27.61	27.25 – 27.97
Double MRF [Wol10]	29.09	28.73 – 29.45
LDCC order 1	**25.39***	25.04 – 25.74
LDCC order 2	**26.04***	25.69 – 26.39
LDCC order 3	27.70	27.34 – 28.06

*Statistically significantly better than Grayscale for a confidence level of 99% using a two-proportion z-test.

Effect of patch size on performance



Conclusions

- Presented a new bleed-through removal technique based on an optimized pixel-by-pixel transformation from color to a single channel.
- The adjustment of parameters is based on an intuitive task to perform by the users. Ideal for an interactive transcription system.
- Could also be used for non-interactive if the conditions of the scanned documents are similar and there is appropriate training data.
- The potential of the proposed technique was demonstrated using a real 17th century manuscript.

Future Work

- Analyze why the performance of the higher models is affected, and propose a new optimization criteria that accounts for it.
- Explore other image features (additional to the pixel color values) to determine which ones can provide an improvement of bleed-through removal performance.
- Do more evaluations: with other datasets, with real users, and integrated into a complete interactive transcription system.
- For other existing bleed-through removal techniques, develop methods for adjusting of parameters based on intuitive tasks that can be easily understood by the end users.

Thank you for your attention!

Questions? Comments?

Examples after generic noise removal

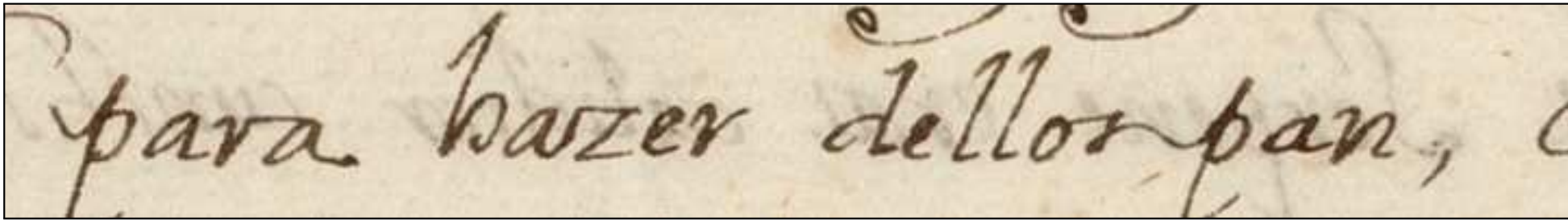
Original

Grayscale, WER=25%

LDCC 1, WER=12.5%

Examples after generic noise removal

Original

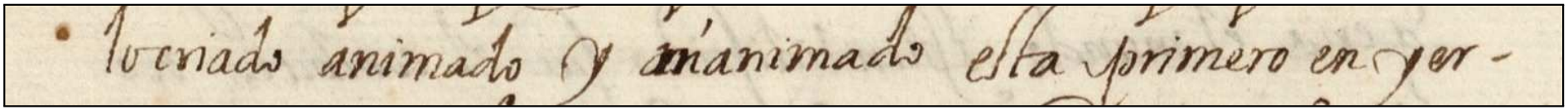


Grayscale, WER=25%

LDCC 1, WER=12.5%

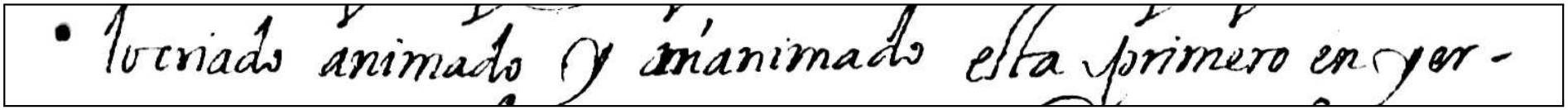
Examples after generic noise removal

Original



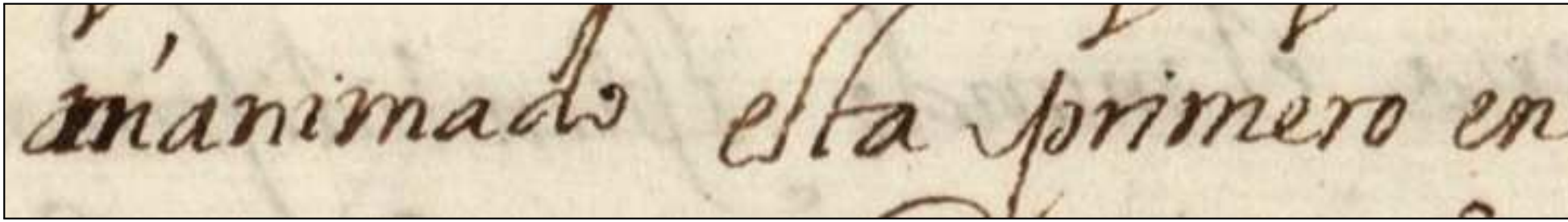
Grayscale, WER=100%

LDCC 1, WER=66.7%

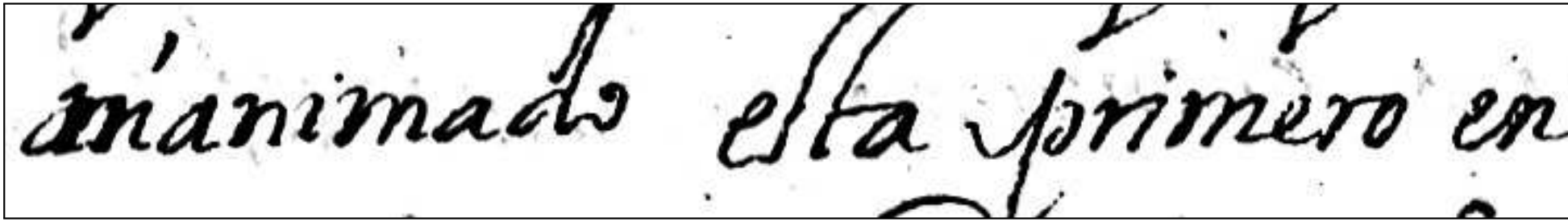


Examples after generic noise removal

Original



Grayscale, WER=100%



LDCC 1, WER=66.7%

Examples after generic noise removal

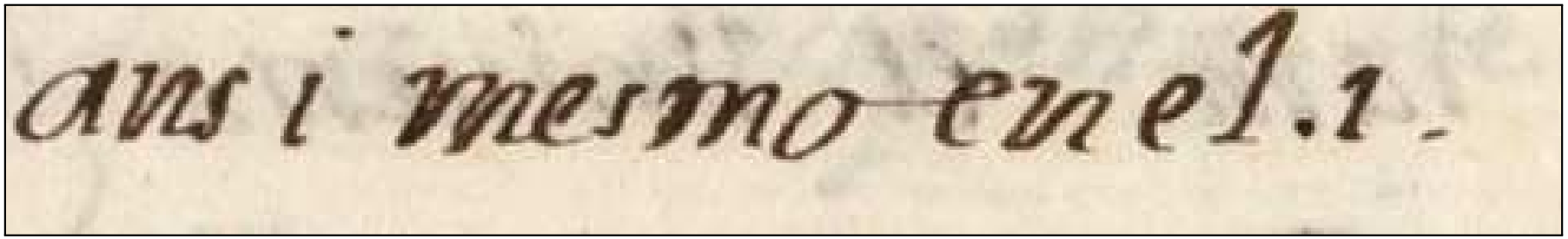
Original

Grayscale, WER=70%

LDCC 1, WER=40%

Examples after generic noise removal

Original

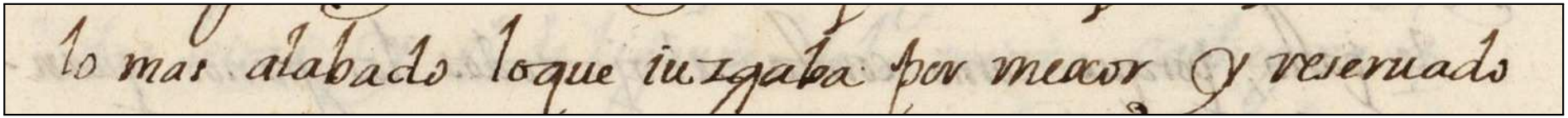


Grayscale, WER=70%

LDCC 1, WER=40%

Examples after generic noise removal

Original

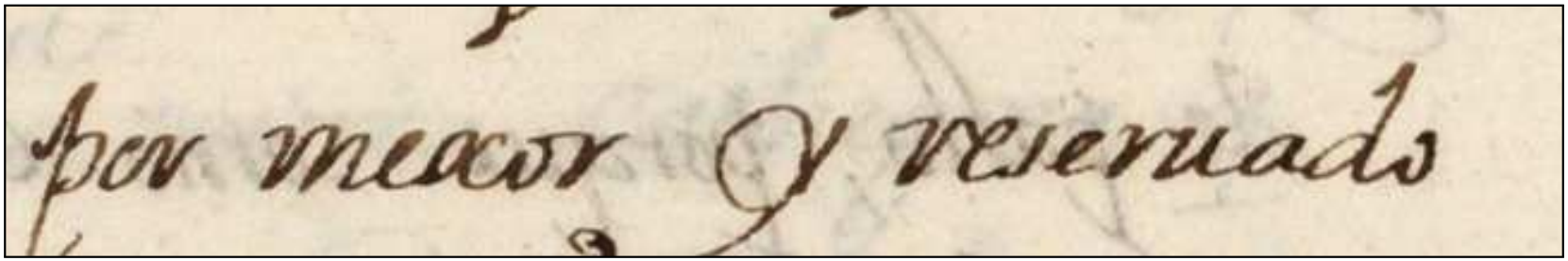


Grayscale, WER=40%

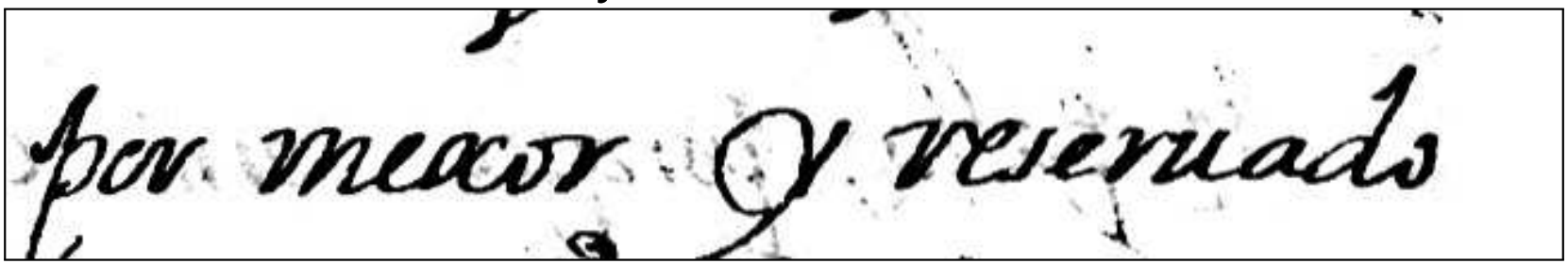
LDCC 1, WER=50%

Examples after generic noise removal

Original



Grayscale, WER=40%



LDCC 1, WER=50%