UNIVERSITAT POLITÈCNICA DE VALÈNCIA
DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

PATTERN RECOGNITION AND
HUMAN LANGUAGE TECHNOLOGY GROUP

IARFID Master Thesis

Interactive Layout Analysis

Author: Lorenzo Quirós Díaz

Advisor: Carlos D. Martínez Hinarejos
Co-advisor: Alejandro Héctor Toselli

September 19, 2016

Abstract

The amounts of ancient documents transcribed by means of HTR technology have been rising dramatically over the last years. Consequently, the development and enhancement of HTR methods and algorithms have been rising as well. However, Layout Analysis remains a bottleneck in the development and generalization of HTR technology. In this work we present a new Interactive-Probabilistic method that incorporates the user feedback in the Layout Analysis process, in order to provide not only a very accurate layout, but an interactive framework in which user feedback is used to help the user to fix any error.

Resumen

La cantidad de documentos antiguos transcritos por medio de la tecnología HTR se ha incrementado drásticamente en los últimos años. En consecuencia, el desarrollo y la mejora de los métodos y algoritmos de HTR también han aumentado. Sin embargo, *Layout Analysis* sigue siendo un cuello de botella en el desarrollo y la generalización de la tecnología de HTR. En este trabajo se presenta un nuevo método Interactivo-Probabilístico que incorpora la realimentación del usuario en el proceso de *Layout Analysis*, con el fin de proporcionar, no sólo un diseño muy preciso, sino un marco interactivo en el que se utiliza la realimentación de los usuarios para ayudar al usuario a corregir cualquier error.

Resum

La quantitat de documents antics transcrits per mitjà de la tecnologia HTR s'ha incrementat dràsticament als últims anys. Açò ha provocat que el desenvolupament i la millora dels mètodes i algoritmes d'HTR hagen augmentat també. Però el *Layout Analysis* continua sent el coll d'ampolla del desenvolupament i la generalització de la tecnologia HTR. Aquest treball presenta un nou mètode Interactiu-Probabilístic que incorpora la retroalimentació de l'usuari al procés de *Layout Analysis* amb la finalitat de proporcionar no només un disseny molt precís, sinó un marc interactiu al que s'utilitza la realimentació dels usuaris per a ajudar a l'usuari a corregir qualsevol error.

ACKNOWLEDGEMENTS

Many people have helped me to complete this work. Thanks to my family and friends for the support and for making this venture more enjoyable, specially to María Vidal and Adolfo Méndez for their reviews and advice. Finally this work have not been completed without the knowledge and advice of Enrique Vidal, Alejandro Héctor Toselli and Carlos D. Martínez, I am very grateful for their time and patience during this journey.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Chapter Outline

1.1 Introduction

Handwritten Text Recognition can be defined as the problem of finding the most likely word sequence for a given handwritten sequence image [13]. Under this definition, the presence of an input image is needed, but this image should contain only a handwritten sequence; this meas that only one "line" of handwritten text is processed at once. Now, since main goal of HTR systems is to translate not just a single line, but the complete page or even the complete book, a previous system is needed in order to extract those lines from the whole page and, in an upper level, to extract the different zones of the page (paragraph, marginal notes, illustrations, page number, etc.). Notice that, this is a very important step of the HTR process; for example, if we develop a system to recognize text in an image, but we provide an illustration without text, the results of that system are expected to be erroneous. In the best case system is capable of ignore that kind of inputs, but we waste time and computing resources on it. In addition, knowledge of the type of zone is very useful to provide some context to the text in the zone, and since each zone can be semantically different, each zone can be processed differently.

Document Layout analysis (DLA) is the process of identifying and categorizing the regions of interest in an image of a document. Commonly this process is divided in two sub problems [6]: detection and labeling of the different zones in the image (body, illustrations, marginalia) is called *geometric layout analysis*, and the classification of these zones into their logical role (title, caption, footnote, etc.) is called the *logical layout analysis*. Several methods have been developed for DLA [1–3, 5, 9, 16, 18], most of them based on Computer Vision techniques, such as, binarization [11, 14], skew correction [10, 12, 15] and Connected Components labeling [4, 8], among others. All of these methods are designed for an user-free system, as a consequence, any error on the system result must be fixed from scratch by the user.

In this work, an Interactive-Probabilistic approach for image layout analysis is proposed, with the aim of providing, not only a very accurate layout, but an interactive framework in which user feedback is used to fix any error. Conditional Random Fields models have been combined with a prior-probability model and Interactive Pattern Recognition [17] to build the new method.

This document is organized as follows: In the first chapter we introduce the topic under study, and cover the motivation and our approach. In Chapter 2 we cover technical fundaments of our approach along with evaluation measures. Our approach is explained in detail in Chapter 3. Then, experiments and results that help us evaluate the proposed method are presented and discussed in Chapter 4. Finally, in Chapter 5 we present the conclusions extracted from our research, and future work to be covered.

1.2 Motivation

Several manuscripts have been digitized in order to preserve the valuable information from the day-by-day deterioration of the physical document. The amount and importance of the information contained in these manuscripts motivates the development of several techniques and tools to explore, analyze, and read them in a more comprehensive manner. These technologies and tools cover from image quality enhancement to automatic Handwritten Text Recognition (HTR).

One of the most crucial steps in HTR is Document Layout Analysis. This is the process in which zones of interest in a page image are detected and categorized. This process is commonly performed by hand, and sometimes assisted by semi-automatic methods [7, 16, 19] (semi-automatic because the final user always needs to review the results).

With the premise that the user always needs to review[a] the results, and assuming the importance of the DLA in HTR systems, it is highly desirable to study interactive methods to improve the aforementioned systems.

1.3 Related Work

Several methods have been developed on the last few years to extract the correct Layout from digitized pages. Most of them [1–3, 5, 9, 16, 18] follow a similar set of steps:

[a]At the time this paper is written state-of-the-art methods are far away to achieve a completely automatic system.

(i) Image binarization.

(ii) Skew correction.

(iii) Connected Components and/or White spaces analysis.

(iv) Some heuristic to link/merge blocks found in the previous step.

(v) Clean the result (filter out small blocks, remove unsuitable blocks, etc.).

Furthermore, in all of the methods studied, a binarization step is performed, which is a very hard problem by itself. This causes an increment in the difficulty of the original problem and a direct dependence from the binarization method. Also, used heuristics limit the generalization of the methods, since several parameters needs to be tuned by the user based on the features of the input images (page degradation, number of columns, illustrations, quality of the scan, etc.).

1.4 Overview of the Proposal Approach

As mentioned above, methods proposed until now still have errors. Thus, the user needs to go trough all the images in order to review if the layout proposed by the classification method is correct or not; then, if there is any error, the user must fix it. In this work we propose to use a probabilistic methodology to provide not only a good layout but to help the user to easily fix any error provided by the classifier. Under this context, we use the Interactive Pattern Recognition (IPR) framework proposed by Toselli et al. [17] to solve this issue presented in Layout Analysis.

The IPR framework forces, from its definition, to use a probabilistic model in order to take advantage of the information inside the model and the information provided by the user iteratively. Therefore, we do not need just a classification of the zones of the page (i.e. a single hypothesis provided by the classifier), but, the probability of each possible hypothesis, where each hypothesis represents a possible layout under the page restrictions. Of course, this normally means a huge search space, even on a simple small image. In order to guide search algorithms to the best hypothesis, a prior-probability distribution over the limits of the zones should be learned form training data. Assuming that zones are rectangles, limits can be defined by upper-left and bottom-right corners; therefore, prior-probability on these two corners is estimated.

We use Conditional Random Fields to model the conditional distribution of the image pixels in each layout zone ($p(h_k|x_{ij})$[b]) and to provide the main probability distribution to the IPR framework. Also, multi-variable Gaussian mixture models are used to learn the prior-probability distribution of the corners of each zone in the layout.

Finally, a brute force approach is taken to search best hypothesis under the probabilistic model and the user feedback.

1.5 Context of Applications and Assumptions

Many enhancements and test cases should be done on the proposed system before reaching stable version of it. For this reason, this project will be divided into some stages to fulfill quality and time restrictions; in this work we will present a the first step, which is restricted by the following constrains for experimentation:

- Experiments will be conducted over a small set of images.
- Experiments will be limited to only search for the main paragraph in the pages.
- Page images are assumed to have only a main paragraph, marginalia, titles, catch-words, and other types of zones, but not illustrations or figures.

These constrains are appropriate to present the method without loss of generality, because all the theory behind it is not limited by the constrains, but just by the experimentation.

[b]See Section 3.1 for a detailed description of the notation

1.6 Expected Outcomes/Results

The expected results are:

- A formal definition of the new probabilistic method, based on CRFs, GMMs and the IPR framework.
- Evaluation of the performance of proposed method, based on state-of-the-art evaluation methodologies.
- A review of the impact of the new approach in the final user effort.
- A demo to present new method features.
- A solid base for future project steps.

Bibliography

[1] Antonacopoulos, A., Clausner, C., Papadopoulos, C., and Pletschacher, S. (2011). Historical document layout analysis competition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pages 1516–1520.

[2] Antonacopoulos, A., Clausner, C., Papadopoulos, C., and Pletschacher, S. (2013). ICDAR 2013 competition on historical book recognition (HBR 2013). *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pages 1459–1463.

[3] Antonacopoulos, A., Clausner, C., Papadopoulos, C., and Pletschacher, S. (2015). 2015_ICDAR2015 Competition on Recognition of Documents with Complex Layouts -RDCL2015. In *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, pages 1151–1155.

[4] Bieniek, A. and Moga, A. (2000). An efficient watershed algorithm based on connected components. *Pattern Recognition*, 33(6):907–916.

[5] Bukhari, S. S., Breuel, T. M., Asi, A., and El-Sana, J. (2012). Layout analysis for Arabic historical document images using machine learning. *Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR*, pages 639–644.

[6] Cattoni, R., Coianiz, T., Messelodi, S., and Modena, C. M. (1998). Geometric layout analysis techniques for document image understanding: a review. Technical report.

[7] Gatos, B., Stamatopoulos, N., and Louloudis, G. (2011). ICDAR2009 handwriting segmentation contest. *International Journal on Document Analysis and Recognition*, 14(1):25–33.

[8] Grana, C., Borghesani, D., and Cucchiara, R. (2010). Optimized block-based connected components labeling with decision trees. *IEEE Transactions on Image Processing*, 19(6):1596–1609.

[9] Lazzara, G., Geraud, T., and Levillain, R. (2014). Planting, growing, and pruning trees: Connected filters applied to document image analysis. *Proceedings - 11th IAPR International Workshop on Document Analysis Systems, DAS 2014*, pages 36–40.

[10] Liu, H., Wu, Q., Zha, H., and Liu, X. (2008). Skew detection for complex document images using robust borderlines in both text and non-text regions. *Pattern Recognition Letters*, 29(13):1893–1900.

[11] Niblack, W. (1985). *An Introduction to Digital Image Processing*. Strandberg Publishing Company.

[12] Papandreou, A., Gatos, B., Louloudis, G., and Stamatopoulos, N. (2013). ICDAR 2013 document image skew estimation contest (DISEC 2013). *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pages 1444–1448.

[13] Romero, V., Serrano, N., Toselli, A. H., Sánchez, J. A., and Vidal, E. (2011). Handwritten Text Recognition for Historical Documents. *Language Technologies for Digital Humanities and Cultural Heritage*, (September):90–96.

[14] Sauvola, J. and Pietikäinen, M. (2000). Adaptive document image binarization. *Pattern Recognition*, 33(2):225–236.

[15] Singh, C., Bhatia, N., and Kaur, A. (2008). Hough transform based fast skew detection and accurate skew correction methods. *Pattern Recognition*, 41(12):3528–3546.

[16] Smith, R. (2009). Hybrid Page Layout Analysis via Tab-Stop Detection. In *Proceedings of the 2009 10th International Conference on Document Analysis and Recognition*, pages 241–245. IEEE Computer Society.

[17] Toselli, A. H., Vidal, E., and Casacuberta, F. (2011). *Multimodal Interactive Pattern Recognition and Applications*. Springer.

[18] Vil'kin, A. M., Safonov, I. V., and Egorova, M. A. (2013). Algorithm for Segmentation of Documents Based on Texture Features. *Pattern Recognit. Image Anal.*, 23(March, 2013):153–159.

[19] Wang, S. Y., Baird, H. S., and An, C. (2009). Document content extraction using automatically discovered features. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pages 1076–1080.

CHAPTER 2

FUNDAMENTS

Chapter Outline

the proposed method is founded on several other methods and algorithms. In this chapter a short overview of those methods and algorithms is presented. Also, some evaluation measures are presented in order to define experiments evaluation criteria.

2.1 Principal Components Analysis

Principal Components Analysis is probably the most known non-supervised technique used to reduce the dimensionality of a data set. Although, PCA does not discriminate between classes, it is very used because most information in the data is preserved after reduction [3].

PCA main goal is to find a projection matrix W, which minimizes the reconstruction error. The solution of this problem is well know [18] and it is to chose those eigenvectors of the covariance matrix, whose eigenvalues are the highest ones; then, the first k eigenvectors, with highest eigenvalues, are used to compose the projection matrix W. The algorithm can be summarized in the following steps:

1. Compute the mean of the data.
2. Subtract the mean to all data.
3. Calculate the covariance matrix on data.
4. Calculate the eigenvectors and eigenvalues of the covariance matrix, ordered by highest eigenvalues.
5. Define projection matrix W by the first k eigenvectors.

For a detailed explanation about how PCA works, please refer to [3, 18]

2.2 Conditional Random Fields

In the case of image labeling, a natural way to represent the dependencies between the pixels of the image is by means of a bidimensional grid-like structure, which can be modeled by a Markov Random Field (MRF) [13]. In this representation, each pixel in the image is represented by a node in the graph and its attributes vector $\boldsymbol{x}$. In this problem the objective is to find the combination of class labels y for each feature vector $\boldsymbol{x}$ that maximizes the probability of the graphical model (Maximum a Posteriori-probability). One way to model this distribution is in terms of the energy associated with a Conditional Random Field (CRF) [12, 20]. If $F = \{\Psi_d\}$ is the set of factors in some factor graph G, where d is an integer index that ranges from 1 to D (being D the number of factors), then the conditional distribution for CRF with respect to a set of computed features is:

$$\mathrm{P}(y|\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} \prod_{\Psi_D \in F} \exp\left\{ \sum_{\zeta=1}^{\mathbb{K}_D} \theta_{d\zeta} \varphi_{d\zeta}(y_d, \mathbf{x_d}) \right\} \tag{2.1}$$

where $\varphi_{d\zeta}$ is a feature function, $\zeta \in \mathbb{K}$; $\mathbb{K}$ the number of features, $\theta_{d\zeta}$ is the set of weights of each factor d. The constant $Z(\boldsymbol{x})$ represents the partition function, a normalization factor to ensure the proper definition of the probability distribution. Typically, the parameters $\theta_{d\zeta}$ will be learned from data by means of some well known methods, namely: BFGS [2], L-BFGS [4], EM [15], SGD [22] or AROW [11].

2.3 Gaussian Mixture Models

Several methods have been developed to estimate the the probability density function (PDF) of any arbitrary distribution from a set of training samples [6, 8, 14]. Among the possible distribution functions estimators, Gaussian mixture models (GMM) are widely used in several tasks and have proved to be a powerful tool.

A Gaussian mixture model is defined as the result of a weighted sum of M Gaussian components:

$$\mathrm{p}(x) = \sum_{i=1}^{M} \pi_i \mathcal{N}(x|\mu_i, \Sigma_i) \tag{2.2}$$

where π_i parameters are called *mixing coefficients* and corresponds with the weight of the component i. These coefficients are restricted to satisfy $\sum_{i=1}^{M} \pi_i = 1$. Each Gaussian density $\mathcal{N}(x|\mu_i, \Sigma_i)$ is called a component of the mixture and has its own mean μ_i and covariance Σ_i. Finally, the estimation of each mixture parameter (π_i, μ_i, Σ_i) is performed by using the Expectation-Maximization (EM) algorithm [8].

2.4 Interactive Pattern Recognition

In the classical Pattern Recognition paradigm, x is an input stimulus, observation or signal and h a hypothesis or output, which the system has to derive from x. Let $\mathcal{M}$ be a model or set of models used by the system to derive its hypotheses. In general, $\mathcal{M}$ is obtained through a "batch" training process from a given training sequence of pairs $(x, h)_i$ from the task being considered [9]. This initial problem statement typically aims at full automation and the "eventual" need for human intervention is overlooked in the mathematical formulation. By ignoring the essential need for human feedback [21], the resulting technologies and systems generally fail to take advantage of the opportunities that could provide the user knowledge.

In traditional PR decision theory it is common to develop techniques that aim at minimizing the cost of wrong hypotheses. In the simplest case, a 0/1 cost function is used, which corresponds to minimizing the number of wrong hypotheses. Under this minimal error criterion, a best hypothesis is shown to be one which maximizes the posterior probability $\mathrm{P}\,r(h|x)$. Using a model $\mathcal{M}$, this is approximated as:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \mathrm{P}\,r(h|x) \approx \arg\max_{h \in \mathcal{H}} \mathop{\mathrm{P}}_{\mathcal{M}}(h|x) \tag{2.3}$$

where $\mathcal{H}$ is the (possibly infinite) set of valid hypotheses. Minimal error is also the main criterion adopted to guide the development of statistical learning approaches to train (the parameters of) $\mathcal{M}$ from the training data.

2.4.1 Definition

Interactive Pattern Recognition (IPR) is a framework for the development of the underlying ideas of feedback, multimodality and adaptation on PR systems. For example, feedback takes the direct advantage of the information provided by the user in each interaction step to improve raw performance.

As in classical PR, x is an input stimulus, observation or signal and h is a hypothesis or output, which the system derives from x. By observing x and h, an operator or user provides some (perhaps null) feedback signal, f , which may iteratively help the system refine or improve its hypothesis until it is finally accepted. $\mathcal{M}$ is a model or set of models which is used by the system to derive its hypotheses. In general, $\mathcal{M}$ is initially obtained in "batch mode", as in traditional PR, from training pairs $(x, h)_i$. Now, during the interactive operation, the valuable user feedback signals produced in each interaction step are advantageously used in an adaptive training process which progressively tunes $\mathcal{M}$ to the specific task and/or to the way the user makes use of the system in this task.

2.4.2 Using the Human Feedback Directly

Human interaction offers a unique opportunity to improve the quality of the system hypothesis h, without varying the model $\mathcal{M}$ [21]. As discussed before, for fixed $\mathcal{M}$ and x, a best hypothesis, $\hat{h}$, is defined by Eq. (2.3). Now interaction allows to add more conditions, that is:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \mathrm{P}_{\mathcal{M}}(h|x, f) \tag{2.4}$$

where f stands for the feedback from the user. Based on that feedback, system must provide a new hypothesis $\hat{h}$ and ask the user for new feedback information. The process continues in this way until the system output is acceptable by the user.

2.4.3 Taking interaction history into account

Since IPR process makes history from previous interaction steps, it would be desirable to use that information to improve the system hypothesis. Let h' be the history. It can be represented by the optimal hypothesis, $\hat{h}$, obtained by the system in its previous interaction steps for the given x. Since previous hypotheses have been supervised/corrected by the user, a part of h' will be correct for the given x. Taking history into account, Eq. (2.4) becomes:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \mathrm{P}_{\mathcal{M}}(h|x, h', f) \tag{2.5}$$

2.4.4 Interaction with Deterministic Feedback

If feedback modality can be considered deterministic (traditional keyboard and mouse), decoding becomes simple, since any feedback can be simplified as a function $d : \mathcal{F} \to \mathcal{D}$, which maps each raw feedback signal, f, into its corresponding unique decoding $d = d(f)$. Hence, we can interchangeably use a feedback signal f and its (trivial and unique) decoding $d = d(f)$. Using d rather than f in Eq. (2.5), applying the Bayes rule, and dropping $\mathcal{M}$ to simplify notation, we can now write the prediction of $\hat{h}$ in more detail:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \mathrm{P}(h|x, h', d) = \arg\max_{h \in \mathcal{H}} \mathrm{P}(x|h', d, h)\, \mathrm{P}(h, h', d) \tag{2.6}$$

2.5 Evaluation Measures

Evaluation methodologies for quantitative and qualitative results in any scientific or engineering study are fundamental in order to measure studied methods results and provide a defined and structured way to be compared withing others. Layout analysis lacks of a single well defined and widespread evaluation methodology; instead, several methods are available in the literature, most of them developed to text line detection. Pixel-wise evaluation (sometimes called Overlap-based) has been used by [1, 5, 16], but pixel-wise evaluation is highly dependent on a strictly defined ground-truth, since the ground-truth for page segmentation is quite ambiguous and may differ between users, other methods have been developed to minimize the dependency on a strictly defined ground-truth on the evaluation metric [7, 19] or just to measure performance based on the zones instead of the pixels [17].
In this work, a general pixel-wise precision/recall methodology is used to evaluate CRFs performance, since no zone is detected but the pixels that belong to a specific zone. For methods were zone is detected, MatchScore [17] and Goal-Oriented Success Rate (GoSR) [19] will be used.

2.5.1 Pixel-wise performance

Pixel-wise performance is computed by means of the well known performance (P, Eq. (2.7)), recall (R, Eq. (2.8)) and F1-score (F1, Eq. (2.9)).

$$P = \frac{TP}{TP + FP} \tag{2.7}$$

$$R = \frac{TP}{TP + FN} \tag{2.8}$$

where TP, FP, and FN stand for "True Positives" (pixels inside a zone, classified as belonging to this zone), "False Positives" (pixels outside a zone, classified as belonging to this zone) and "False Negatives" (pixels inside a zone, classified as not belonging to this zone) respectively.

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \tag{2.9}$$

2.5.2 MatchScore

MatchScore was used in the ICDAR (2005 to 2009) competitions [10]. The method is intended, first, to select if a zone pair (h_i^*, h_k) is a one-to-one match (Eq. (2.10)), i.e. the ground-truth zone h_i^* match to result zone h_k based on some threshold T_a; then, if the zones have a one-to-one match performance, metric is computed as shown in Eq. (2.11).

$$\mathrm{MatchScore}(i, k) = \frac{\mathrm{T}(h_i^* \cap h_k)}{\mathrm{T}(h_i^* \cup h_k)} \tag{2.10}$$

where $\mathrm{T}(\cdot)$ a function which counts the pixels inside a zone.

$$FM_{i,k} = \frac{2\mathrm{T}(h_i^* \cap h_k)}{\mathrm{T}(h_i^*) + \mathrm{T}(h_k)} \tag{2.11}$$

Finally, global performance is extracted by calculating the average values of performance metric for all the one-to-one zone pairs.

2.5.3 Goal-Oriented Success Rate

Goal-Oriented performance [19] aims at evaluate how much of the information contained in the ground-truth is also contained in the system result; due to that, only foreground pixels are taken into account. Use of foreground pixels forces us to have a pixel level ground-truth, most of the times in the form of a binarized version of the image. Also, the methodology used to define if some zones have a one-to-one match is changed from MatchScore to Eq. (2.12).

$$I_{ik} = \begin{cases} h_i^* \cap h_k & \text{if } h_i^* \cap h_k \neq \emptyset \text{ and } \frac{\mathrm{T}(I_{ik})}{\mathrm{T}(h_k)} > T_a \\ \emptyset & \text{otherwise} \end{cases} \tag{2.12}$$

Then, success rate (SR) is defined as follows:

$$SR = \frac{\sum_{i=1}^{|h^*|} \sum_{k=1}^{K} w_{ik} \times \mathrm{T}(I_{ik})}{\sum_{i=1}^{|h^*|} \mathrm{T}(h_i^*)} \tag{2.13}$$

where w_{ik} corresponds to a weight for each intersection region I_{ik} ranging the interval $[0, ..., 1]$, which depends on the following conditions[a]:

(i) the ground-truth region h_i^* has been detected correctly

(ii) the ground-truth region h_i^* has been split

(iii) the result region h_k has been overlapped by two or more ground-truth regions (merge)

(iv) non-text elements have been included in the result region h_k

(v) if more than one of the previous conditions is satisfied, the weight with the smaller value is selected

Notice that, in our case, since only one zone is under scrutiny (the main paragraph), any other zone in the layout is considered as a non-text element; therefore, w_{ik} belongs to condition (iv), hence, to Eq. (2.14).

$$w_{ik} = \frac{\mathrm{T}(I_{ik})}{\mathrm{T}(h_k)} \tag{2.14}$$

Bibliography

[1] Baechler, M. and Ingold, R. (2011). Multi Resolution Layout Analysis of Medieval Manuscripts Using Dynamic MLP. In *Proceedings of the 2011 International Conference on Document Analysis and Recognition*, pages 1185–1189, Washington, DC, USA. IEEE Computer Society.

[2] Bertsekas, D. P. (1999). *Nonlinear Programming*. Athena Scientific, 2 edition.

[3] Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 1st edition.

[a] A full list of equations for each condition can be found in [19].

[4] Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM J. Sci. Comput.*, 16(Sept. 1995):1190–1208.

[5] Chaudhury, S., Jindal, M., and Roy, S. D. (2009). Model-Guided Segmentation and Layout Labelling of Document Images using a Hierarchical Conditional Random Field. In *International Conference on Pattern Recognition and Machine Intelligence*, pages 375–380. Springer Berlin Heidelberg.

[6] Chen, S. X. (2000). Probability Density Function Estimation Using Gamma Kernels. *Annals of the Institute of Statistical Mathematics*, 52(3):471–480.

[7] Clausner, C., Pletschacher, S., and Antonacopoulos, A. (2011). Scenario driven in-depth performance evaluation of document layout analysis methods. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pages 1404–1408.

[8] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Source Journal of the Royal Statistical Society. Series B (Methodological) Journal of the Royal Statistical Society. Series B*, 39(1):1–38.

[9] Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification*. Wiley-Interscience, 2nd edition.

[10] Gatos, B., Stamatopoulos, N., and Louloudis, G. (2011). ICDAR2009 handwriting segmentation contest. *International Journal on Document Analysis and Recognition*, 14(1):25–33.

[11] Kulesza, A., Crammer, K., and Dredze, M. (2009). Adaptive Regularization of Weight Vectors. *Advances in Neural Information Processing Systems*, 22:414–422.

[12] Lafferty, J., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning*, 8(Icml):282–289.

[13] Li, S. Z. (1994). Markov Random Field Models in Computer Vision. In *European Conference on Computer Vision*, pages 361–370. Springer-Verlag.

[14] MacQueen, J. B. (1967). Kmeans Some Methods for classification and Analysis of Multivariate Observations. *5th Berkeley Symposium on Mathematical Statistics and Probability 1967*, 1(233):281–297.

[15] McCallum, a., Bellare, K., and Pereira, F. (2012). A conditional random field for discriminatively-trained finite-state string edit distance. *arXiv preprint arXiv:1207.1406*.

[16] Nicolas, S., Paquet, T., and Heutte, L. (2006). A Markovian approach for handwritten document segmentation. *Proceedings - International Conference on Pattern Recognition*, 3:292–295.

[17] Phillips, I. T. and Chhabra, A. K. (1999). Empirical Performance Evaluation of Graphics Recognition Systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):849–870.

[18] Smith, L. I. (2002). A tutorial on principal components analysis. *Cornell University*.

[19] Stamatopoulos, N., Louloudis, G., and Gatos, B. (2015). Goal-Oriented Performance Evaluation Methodology for Page Segmentation Techniques. In *13th International Confrence on Document Analysis and Recognition - ICDAR'15*, pages 281–285.

[20] Sutton, C. and McCallum, A. (2010). An Introduction to Conditional Random Fields. *Machine Learning*, 4(4):267–373.

[21] Toselli, A. H., Vidal, E., and Casacuberta, F. (2011). *Multimodal Interactive Pattern Recognition and Applications*. Springer.

[22] Vishwanathan, S., Schraudolph, N. N., Schmidt, M. W., and Murphy, K. P. (2006). Accelerated training of conditional random fields with stochastic gradient methods. *Proceedings of the 23rd international conference on Machine learning ICML 06*, 148:969–976.

CHAPTER 3

INTERACTIVE LAYOUT ANALYSIS

Chapter Outline

3.1 Interactive Layout Analysis

Document Layout analysis is the process of identifying and categorizing the regions of interest in an image of a document. Commonly, this process is divided into two sub problems [1]: detection and labeling of the different zones in the image (body, illustrations, marginalia) is called geometric layout analysis, and the classification of these zones into their logical role (title, caption, footnote, etc.) is called the logical layout analysis.

As an input we have an image $\mathcal{X} = \{x_{1,1}, x_{1,2}, .., x_{n,m}\}$, which is associated with a rectangular grid G of size $n \times m$. Each image site s is associated to a cell in the grid defined by its coordinates over G and denoted $G_{ij}, \quad 1 \leq i \leq n, \quad 1 \leq j \leq m$. The site set is denoted $S = \{s_1, s_2, ..., s_D\} \quad 1 \leq D \leq n \times m$.

Now lets define $L = \{l_1, l_2, ..., l_\ell\}$ the set of all the possible zones in a Layout, under this definition, L is a unconstrained set, then any combination of zones is allowed, for instance, paragraph, marginalia, illustrations, lines, words, etc. Notice that, since L is unconstrained several zones of the same type are allowed as well. In the other hand, each image is associated to some K-zones Layout defined as $h^* = \{h_1, h_2, ..., h_K\}; h^* \subseteq L$, henceforth, h^* is called the Layout ground-truth of that specific image. Then each site s_d belongs to a single layout zone, i.e., $s_d \implies h_k; \quad h_k \in h^*, \quad s_d \in S$.

In our case, each layout zone is a rectangle defined by its coordinates over G as $h_k = (\boldsymbol{u}_k, \boldsymbol{b}_k), \quad 1 \leq k \leq K$. Where $\boldsymbol{u}_k$ are the upper-left corner, and $\boldsymbol{b}_k$ re the bottom-right ones. Figure 3.1 shows an example of this kind of layout zones.

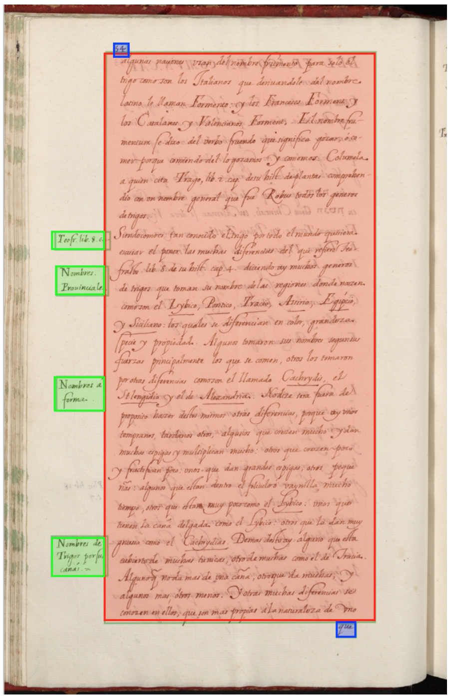


Figure 3.1: Layout zones example. Red=Paragraph, Green=marginalia, Blue=catch-word

3.1.1 Layout Analysis Problem

Let define a *structured hypotheses space* [5] $\mathcal{H} = \{h^1, h^2, ..., h^T\}$[a] over the site set S, where $h^t \subseteq L, \quad 1 \leq t \leq T$. We want the hypothesis $\hat{h}$ which provides the best layout for the site set. Under *minimal error criterion*, a best hypothesis is shown to be the one which maximizes the posterior probability [2] $\mathrm{P}(h|S)$.

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \mathrm{P}(h|S) \tag{3.1}$$

However, in many cases it is difficult to directly estimate $\mathrm{P}(h|S)$ and is better to apply the Bayes rule [5]:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \frac{\mathrm{P}(S|h)\,\mathrm{P}(h)}{\mathrm{P}(S)} = \mathrm{P}(h)\,\mathrm{P}(S|h) \tag{3.2}$$

[a] Based on this definition, $\mathcal{H}$ is finite but huge; in the worst case, each site s belongs to a different layout zone; then, $|\mathcal{H}|$ is defined by the size of the set of sites and the number of different zones as:

$$|\mathcal{H}| = \binom{|L| + D - 1}{D} = \frac{(|L| + D - 1)!}{D!(|L| - 1)!}$$

where the term $\mathrm{P}(S)$ has been dropped since it does not depend on the maximization variable, h. $\mathrm{P}(S|h)$ is the probability of a site set S given the layout hypothesis h, and $\mathrm{P}(h)$ is its prior probability.

Prior probability $\mathrm{P}(h)$ can be modeled by a Gaussian Mixture Model ($\mathcal{M}_g$), which is a Gaussian mixture for each corner of each layout zone, where each corner is assumed independent of the others but constrained to $\boldsymbol{u}_k > \boldsymbol{b}_k \forall k \in h$ (element-wise), by computational reasons.

$$\mathrm{P}(h) \approx \underset{\mathcal{M}_g}{\mathrm{P}}(h) = \prod_{k=1}^{K} \underset{\mathcal{M}_g}{\mathrm{P}}(\boldsymbol{u}_k) \underset{\mathcal{M}_g}{\mathrm{P}}(\boldsymbol{b}_k) \tag{3.3}$$

The likelihood, $\mathrm{P}(S|h)$, can be approached through a simple naïve Bayes decomposition, under spatial independence assumption[b], as follows:

$$\mathrm{P}(S|h) = \prod_{d=1}^{D} \mathrm{P}(s_d|h) \tag{3.4}$$

where each $\mathrm{P}(s_d|h)$ can be modeled, for instance, by K-NN, CRF's, RBMs+linear regression function, NN, etc.

Formally, Eq. (3.2) can be re-written as:

$$\hat{h} \approx \underset{h \in \mathcal{H}}{\arg\max} \prod_{k=1}^{K} \underset{\mathcal{M}_g}{\mathrm{P}}(\boldsymbol{u}_k) \underset{\mathcal{M}_g}{\mathrm{P}}(\boldsymbol{b}_k) \prod_{d=1}^{D_k} \mathrm{P}(s_d|(\boldsymbol{u}_k, \boldsymbol{b}_k)) \tag{3.5}$$

where D_k is the sub-set of sites inside the layout zone k. In order to prevent precision issues on our calculations, we apply $\log$ in both sides of Eq. (3.5).

$$\log \hat{h} \approx \underset{h \in \mathcal{H}}{\arg\max} \sum_{k=1}^{K} \Bigg(\log \underset{\mathcal{M}_g}{\mathrm{P}}(\boldsymbol{u}_k) + \log \underset{\mathcal{M}_g}{\mathrm{P}}(\boldsymbol{b}_k) + \sum_{d=1}^{D_k} \log \mathrm{P}(s_d|(\boldsymbol{u}_k, \boldsymbol{b}_k)) \Bigg) \tag{3.6}$$

[b]Spatial independence: each site is independent from the others in the set.

3.1.2 Interactive Framework

Interactive framework aims to improve the classical pattern recognition paradigm results by taking into account user feedback. As Layout Analysis problem is defined in Eq. (3.6), system will provide the best hypothesis it can, but if there is any error, user commonly will need to remove the prediction and define layout from scratch. This kind of issues could be minimized by the insertion of the feedback (f) and the feedback history (h') on the model [5]. taking Eq. (2.6) into Eq. (3.6) an interactive version of the system is defined as:

$$\log \hat{h} \approx \arg\max_{h \in \mathcal{H}} \sum_{k=1}^{K} \left(\log \underset{\mathcal{M}_g}{\mathrm{P}} (\boldsymbol{u}_k | h', f) + \log \underset{\mathcal{M}_g}{\mathrm{P}} (\boldsymbol{b}_k | h', f) + \sum_{d=1}^{D_k} \log \mathrm{P}(s_d | (\boldsymbol{u}_k, \boldsymbol{b}_k), h', f) \right) \tag{3.7}$$

As a result, the new system must be designed to compute Eq. (3.6) as a first hypothesis, and then Eq. (3.7) will be computed on each iteration until the presented hypothesis satisfies the user.

3.2 System Architecture

The proposed method can be divided into a two stage process:

- **Stage 1:** Traditional Pattern Recognition *batch-processing* paradigm is performed, using a small training set, thus, we use a small training set to learn about the corpus and train the probabilistic model.
- **Stage 2:** Then, user interaction is used to review the results and correct system mistakes.

On each stage, several sub-process must be performed. Figure 3.2 presents a block diagram of the system, and a detailed explanation of each block is presented in further sections in this chapter.

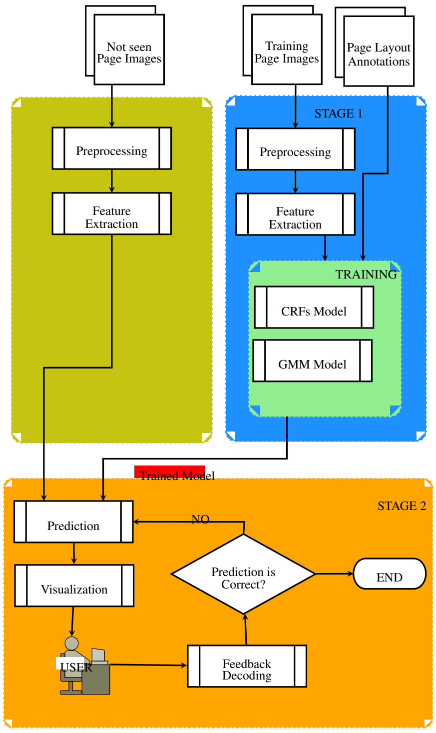


Figure 3.2: System Architecture Diagram.

3.3 Pre-processing

In order to keep this stage of the project as simple as possible, and under the assumption that pre-processing images will provide an enhancement over system performance, images are only transformed to color intensity space and resolution is reduced by feature extraction constrains. This pre-processing steps are focused on reducing the computational cost instead of improving the system performance. Other well know techniques, such as, noise reduction, binarization and skew correction, would be studied on future project stages.

3.4 Feature Extraction

For each site $s_d \in S$ we extract a set of characteristics, this set of characteristics is divided into a two groups: 1. the site color intensity and its neighbors, 2. the site position. First group is extracted from a $(w \times w)$ window, centered on the site s_d as follows (See Figure 3.3 for a visual representation of the process):

$$\nu_d = \mathbf{Y}[\lfloor\frac{d}{c}\rfloor - \lfloor\frac{w}{2}\rfloor : \lfloor\frac{d}{c}\rfloor + \lfloor\frac{w}{2}\rfloor, d - \lfloor\frac{d}{c}\rfloor - \lfloor\frac{w}{2}\rfloor : d - \lfloor\frac{d}{c}\rfloor + \lfloor\frac{w}{2}\rfloor] \quad (3.8)$$

where $\mathbf{Y}[a_1 : a_2, a_3 : a_4]$ stands for a sub-matrix of $\boldsymbol{Y}$ defined by the rows from a_1 to a_2 and the columns from a_3 to a_4; and $\boldsymbol{Y}$ a matrix containing the value of the color intensity of each site in the set S of size $(r \times c); r * c = D$.

Then, depending of w the size of ν_d could be very high, for example, if $w = 33$, the size of ν_d is 1024. In order to reduce the size of the attributes vector, we take each row of ν_d as an data input and apply PCA algorithm over it to reduce data dimensionality to a $(\eta \times w)$ new matrix; finally we apply PCA again over this new matrix to reduce attributes matrix to a $(\eta \times \eta)$ matrix, now this reduced matrix is re-shaped to a $(1 \times \eta^2)$ vector and defined as:

$$\boldsymbol{\beta}_d = \mathrm{vec}(\Upsilon(\Upsilon(\nu_d, \eta)^T, \eta))^T \quad (3.9)$$

where $\mathrm{vec}(\cdot)$ is the vectorization function [3], and $\Upsilon(\boldsymbol{x}, \eta)$ is a function that returns the re-scaled or reduced data of input $\boldsymbol{x}$, by means of PCA algorithm, using the first η eigenvectors.

Finally a set of eleven features are selected based on previous attributes, also η parameter is anchored to 3, in order to keep $\boldsymbol{\beta}_d$ small (only 9 elements):

$$\begin{aligned}
\varphi_{d0} &= \boldsymbol{\beta}_d \\
\varphi_{d1} &= \boldsymbol{\beta}_{d-1} \\
\varphi_{d2} &= \boldsymbol{\beta}_{d+1} \\
\varphi_{d3} &= \boldsymbol{\beta}_{d-c} \\
\varphi_{d4} &= \boldsymbol{\beta}_{d+c} \\
\varphi_{d5} &= \boldsymbol{\beta}_{d-1}|\boldsymbol{\beta}_d \\
\varphi_{d6} &= \boldsymbol{\beta}_d|\boldsymbol{\beta}_{d+1} \\
\varphi_{d7} &= \boldsymbol{\beta}_{d-c}|\boldsymbol{\beta}_d \\
\varphi_{d8} &= \boldsymbol{\beta}_d|\boldsymbol{\beta}_{d+c} \\
\varphi_{d9} &= \lfloor\frac{d}{c}\rfloor \\
\varphi_{d10} &= d - \lfloor\frac{d}{c}\rfloor
\end{aligned}$$

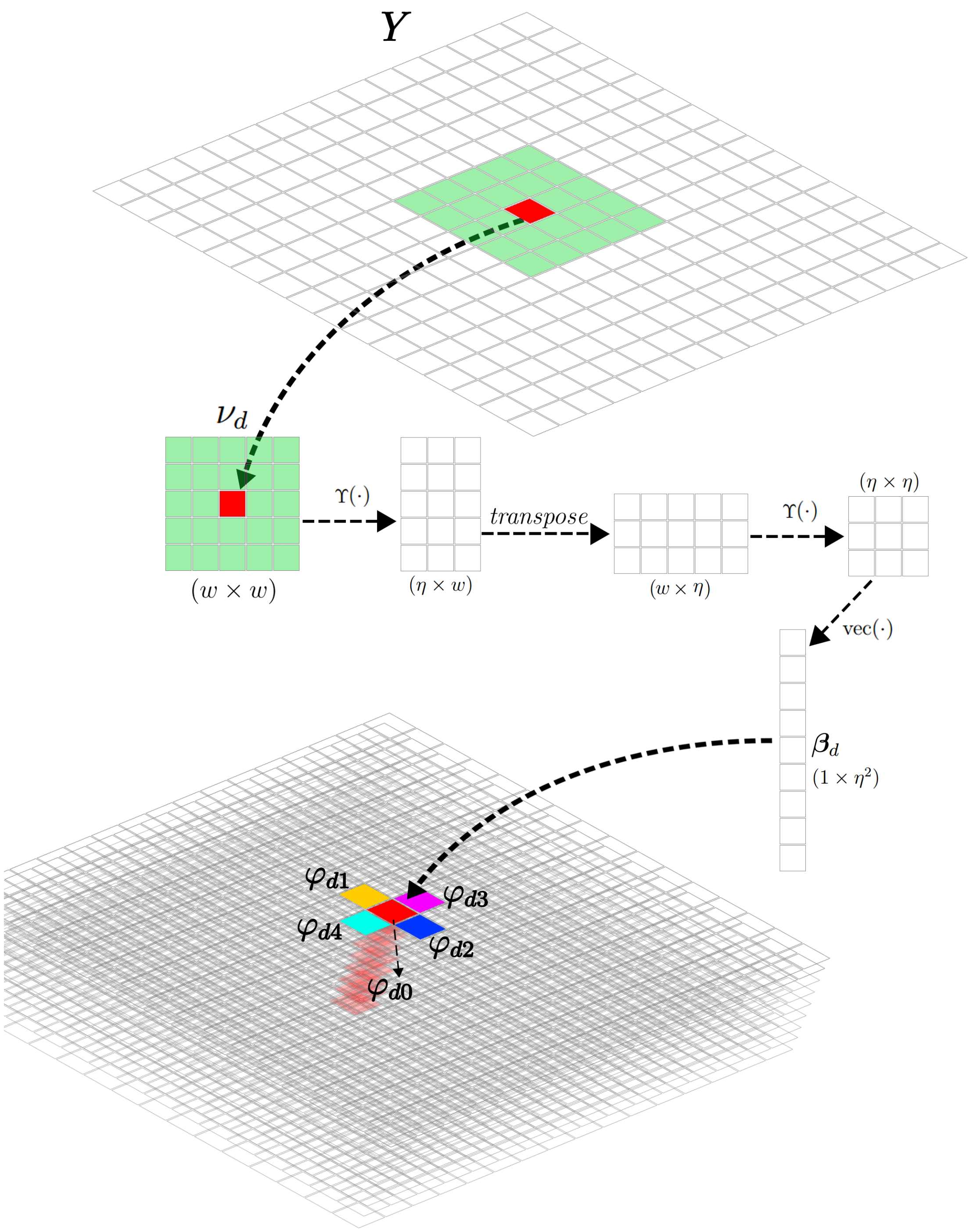


Figure 3.3: Feature extraction process

where first eight features are related to site color intensity and final two are the position of the site in the image. Also, "bi-gram" notation $(\boldsymbol{\beta}_i|\boldsymbol{\beta}_j)$ is used on features five to eight in order to represent sequential relationship between the site s_d and its neighbors.

3.5 CRF's Learning

As mentioned in Section 2.2, CRF's is a powerful tool to estimate a posteriori-probability that maximizes the probability of a graphical model. In this work the `CRFsuite` tool [4] is used to learn a CRF model from features selected in Section 3.4. `CRFsuite` is an implementation of linear-chain CRFs for labeling sequential data; it is a very well know tool for fast training and tagging. It is very fast and, most important, definition of features is very flexible, which allows the user to define many kind of features. The format of the input data is very well described on the tool's web site [4]. Based on the format restrictions of the tool, and in order to keep the number of features low (and the model size as well), each feature is internally converted into strings by adding a "-" character between each value; for instance, the feature $\varphi_{d\zeta} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8]$ is converted to "$\varphi_{d\zeta}$=w_0-w_1-w_2-w_3-w_4-w_5-w_6-w_7-w_8".

Finally, each feature should look like the following example, where TAB character was replaced by new-line in order to keep it more readable, and feature name changed to keep previous notation:

0
φ_{d0}=210−127−127−86−131−127−86−123−127
φ_{d1}=45−127−127−168−131−127−168−123−127
φ_{d2}=44−127−127−169−131−127−169−123−127
φ_{d3}=29−127−127−177−122−127−176−132−127
φ_{d4}=70−127−127−156−130−127−156−124−127
φ_{d5}=29−127−127−177−122−127−176−132−127|210−127−127−86−131−127−86−123−127
φ_{d6}=45−127−127−168−131−127−168−123−127|210−127−127−86−131−127−86−123−127
φ_{d7}=210−127−127−86−131−127−86−123−127|44−127−127−169−131−127−169−123−127
φ_{d8}=210−127−127−86−131−127−86−123−127|70−127−127−156−130−127−156−124−127
φ_{d9}=4
φ_{d10}=79

After training, marginal probability of each class is extracted using the `tag` option, giving $\mathrm{P}(s_d|h_0)$ and $\mathrm{P}(s_d|h_1)$, were h_0 and h_1 means background and paragraph, respectively. In Figure 3.4 we can see an example of the marginal probabilities obtained from the CRF model. A learning algorithm should be selected by experimentation (see Section 4.3.1 for details).

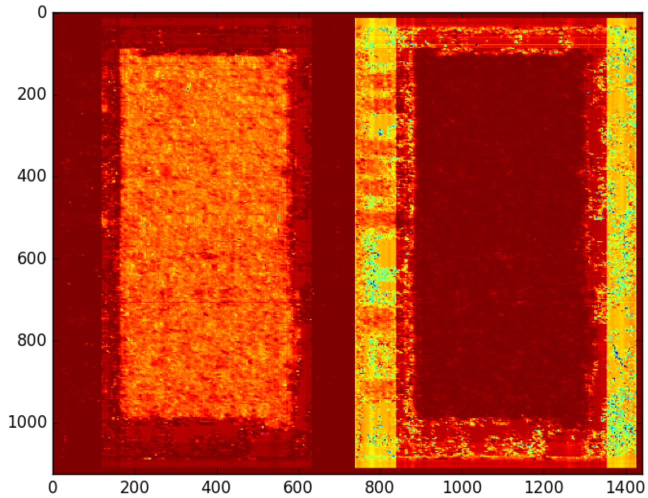


Figure 3.4: Marginal probability maps extracted from CRF model results. Left image is the map for background and right one is the map for paragraph zone. Dark red means high, and blue/yellow means low.

3.6 GMM Learning

The prior-probability model of corner location of the zones in the image can be estimated by a multi-variable GMM. Thus, corners in the ground-truth (h^*) are used to estimate a GMM model over each corner. For this stage of the project (only the main paragraph is taken into account), upper-left corner of the zone ($\boldsymbol{u}_k$) is modeled by a Gaussian mixture of two components, whereas bottom-right corner ($\boldsymbol{b}_k$) is modeled by three components; in both of them a diagonal covariance matrix is used, in order to keep mixture axis parallel to image axis.

Many tools have been developed to estimate GMMs from data, using the EM algorithm. For Python a library called `scikit-learn` is a particularly well known library which implements GMM estimation, among many other algorithms. It is very straightforward to use and efficient; hence, it is used in this work to estimate GMMs of each corner of each layout zone. In Figure 3.5, log-probability of main paragraph corners is drawn for all possible $\boldsymbol{u_k}$ and $\boldsymbol{b}_k$ (k equal paragraph in this case) in an image.

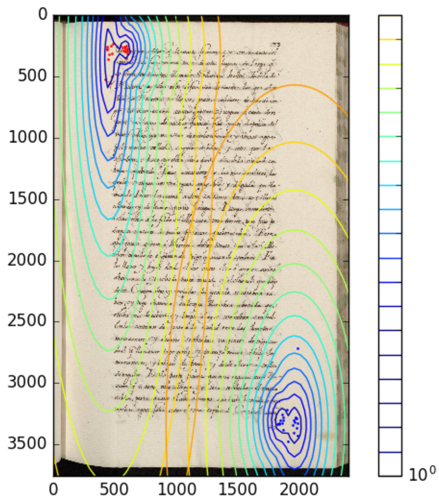


Figure 3.5: Gaussian Mixture Models over main paragraph corners.

3.7 User Interaction

As defined in previous sections, the user will review all the pages and provide some feedback to the system in order to fix any error, i.e., user will interact with the system only through the mouse following deterministic rules:

(i) Left-click over the image: mouse pointer coordinates are a correct corner of some zone in h^*.

(ii) Right-click over the image: layout is correct (i.e. "OK signal")

(iii) Click outside the image: close the system.

As a consequence of using a deterministic feedback, the signal does not need to be "decoded" and it can be used directly [5] on Eq. (3.7) as the feedback f. Feedback under this context means a constrain under the search space $\mathcal{H}$, i.e., rule (i) limits the search space to only those hypotheses where the selected coordinates in the image are part of h; then, that coordinate will be considered as "anchored". Also, history is important to constrain the search space to only those zone corners in the hypothesis which are not "anchored" by the user in previous iterations. User interaction is defined in Algorithm 1; note that this algorithm encompass completely Stage 2 presented in Figure 3.2.

Algorithm 1: Interactive Algorithm

Data: a set of images $\mathcal{X}$, Prob site model $\mathcal{P}$, Prob layout Model $\mathcal{Q}$
Result: best, under the model, array of layout hypothesis $\hat{\boldsymbol{h}}$
for $x \in \mathcal{X}$ **do**
  $\hat{h}_x$ = Eq. (3.6);
  while $f \neq OK$ **do**
    $h' = \hat{h}_x$;
    $f =$ decodeUserFeedback();
    $\hat{h}_x$ = Eq. (3.7);
return $\hat{\boldsymbol{h}}$

Bibliography

[1] Cattoni, R., Coianiz, T., Messelodi, S., and Modena, C. (1998). Geometric layout analysis techniques for document image understanding: a review. *ITC-irst Technical Report*, pages 1–68.

[2] Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification*. Wiley-Interscience, 2nd edition.

[3] Magnus, J. R. and Neudecker, H. (1999). *Matrix differential calculus with applications in statistics and econometrics*. J. Wiley & Sons, Chichester, New York, Weinheim.

[4] Okazaki, N. (2007). CRFsuite: a fast implementation of Conditional Random Fields (CRFs).

[5] Toselli, A. H., Vidal, E., and Casacuberta, F. (2011). *Multimodal Interactive Pattern Recognition and Applications*. Springer.

CHAPTER 4

EXPERIMENTS AND RESULTS

Chapter Outline

4.1 Corpus Description

To develop and test the new method proposed in this document a manuscript under the following requirements is preferred to be selected:

- Reasonably good state of preservation.
- Available in some digital format.
- Reasonably large.
- Well-defined layout zones.
- Layout should be complex enough to exemplify method strengths and weaknesses.
- Ground-truth available or easy to build.

Under these constrains, the manuscript chosen for the present work is the first tome of a seven volume manuscript entitled "Historia de las Plantas" PLANTAS for short, a XVII century handwritten botanical specimen book compiled by Bernardo Cienfuegos, one of the most outstanding Spanish botanists in the XVII century. The first volume has 49 pages at the beginning comprising indices, reference tables, a botanical glossary in different languages, and a 36-page preface written by Cienfuegos. This is followed by 887 numbered pages that contain 152 chapters about cereals and related plants, including 126 botanical illustrations. All in all, the first volume has 1 035 pages, containing about 20,000 handwritten text lines. This corpus is already digitized at 300ppi in 24 bit RGB color, available as JPG images along with their respective ground-truth layout in PAGE XML format [7] compiled by PRHLT group [1] using seven categories, namely: catch-word, heading, marginalia, page-number, paragraph, signature-mark, and float (illustrations); see Figure 4.1 for reference.

In this stage of the presented work only a sub-set of 39 pages was considered, in order to accomplish with stage and time restrictions. Pages that contains indexes, reference tables, and illustrations were excluded since goals of this stage are restricted to the main paragraph. 22 of these pages were selected for training the model, and the remaining 17 for test (see Section 6.1 for reference).

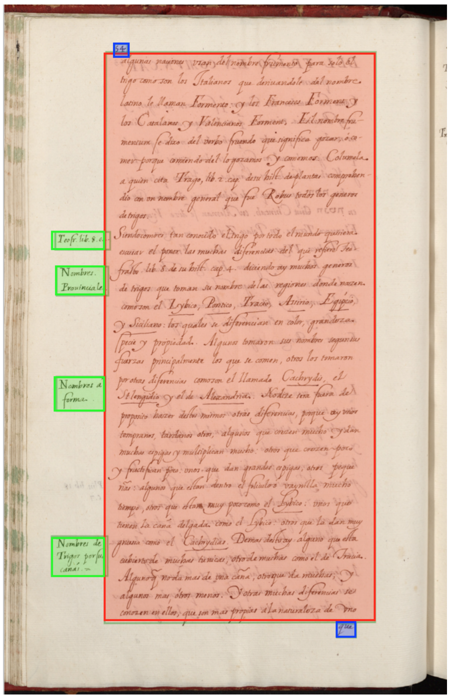


Figure 4.1: PLANTAS layout zones; green for marginalia, red for paragraph, and blue for catch-word.

4.2 Implementation Notes

System was implemented mainly in Python 2.7 due to the facilities provided by the language (N-dimensional array structures, image processing tools, plotting tools, etc) and how quickly is possible to develop a new piece of software. Besides, `CRFSuite` is used to handle CRF training and tagging steps. In this chapter some implementation details will be explained for clarification; most of them are related to non-direct implementation of the theory explained in Chapters 2 and 3, and implemented towards code optimization.

4.2.1 Integral Image

Integral Image (or sometimes called Summed Area Table) is a data structure and algorithm first used by Crow [3] in Computer Graphics, and introduced by Viola and Jones [8] to Computer Vision. It is widely used for quickly and efficiently generating the sum of values in a rectangular subset of a grid. The Integral Image $\boldsymbol{I}$ of an $M \times N$ input image A is defined as a 2D cumulative sum of A:

$$\boldsymbol{I}[x,y] = \sum_{x' \leq x, y' \leq y} A(x',y') \quad x \leq M, y \leq N \tag{4.1}$$

Then, using the integral image any rectangular sum can be computed in four array references [8] (see Figure 4.2), i.e. four accesses to the I matrix instead of $(x\dot{y})$ accesses to A using the direct approach.

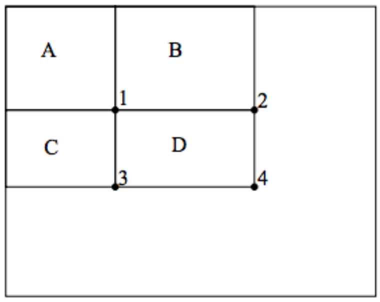


Figure 4.2: The sum of the pixels within rectangle D can be computed with four array references. The value of the Integral Image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is A + B, at location 3 is A + C, and at location 4 is A + B + C + D. The sum within D can be computed as $4 + 1 - (2 + 3)$.[8]

Since the CRF model provides the probability of each site in the site set S, we can see it as a matrix and use Integral Image for all related calculations. For example in Eq. (3.7) we need to compute the sum of the probability of each site inside each k layout-zone; this can be computed as:

$$\sum_{d=1}^{D_k} \log \mathrm{P}(s_d | (\boldsymbol{u}_k, \boldsymbol{b}_k)) = \boldsymbol{I}_k[\boldsymbol{b}_k] + \boldsymbol{I}_k[\boldsymbol{u}_k] - \boldsymbol{I}_k[\boldsymbol{u}_{kr}, \boldsymbol{b}_{kc}] - \boldsymbol{I}_k[\boldsymbol{b}_{kr}, \boldsymbol{u}_{kc}] \tag{4.2}$$

with vectors $\boldsymbol{u}_k$ and $\boldsymbol{b}_k$ defined in Section 3.1.

4.2.2 Element-wise to matrix evaluation

User interaction requires a system fast enough to keep real time feeling to the user, i.e. less than 1.0 seconds [6]. In this system user interacts as explained on Section 3.7, but direct implementation of Eq. (3.7) using `for` loops is just not fast enough to keep real time feeling (≈ 25 seconds per click). In order to reduce the time consumed by the CPU to compute Eq.

(3.7) a matrix-like version of the equation is implemented. This allows the system to use `numpy` multidimensional and broadcasting features to reduce delay.
Then, Eq. (3.7) is transformed into Eq. (4.3), where sums have been expanded to the specific case of this work (only main paragraph); $\boldsymbol{I}_k$ means for the Integral Image of the probabilities of the layout zone k, $\boldsymbol{P}_{\boldsymbol{u}_1}$ and $\boldsymbol{P}_{\boldsymbol{b}_1}$ are the probability matrix from $\boldsymbol{u}_1$ and $\boldsymbol{b}_1$ GMM models respectively, f_r and f_c are the decoded feedback from the user as the row and column selected respectively.

$$\begin{aligned}
\hat{h}_{\boldsymbol{b}_1} = {} & \arg\min \boldsymbol{I}_0[n,m] \\
& - (\boldsymbol{I}_0[f_r:n, f_c:m] + \boldsymbol{I}_0[f_r-1, f_c-1] - \boldsymbol{I}_0[f_r-1, f_c:m] - \boldsymbol{I}_0[f_r:n, f_c-1]) \\
& + (\boldsymbol{I}_1[f_r:n, f_c:m] + \boldsymbol{I}_1[f_r-1, f_c-1] - \boldsymbol{I}_1[f_r-1, f_c:m] - \boldsymbol{I}_1[f_r:n, f_c-1]) \\
& + \boldsymbol{P}_{\boldsymbol{u}_1}[f_r, f_c] + \boldsymbol{P}_{\boldsymbol{b}_1}[f_r:n, f_c:m] \\
\hat{h}_{\boldsymbol{u}_1} = {} & \arg\min \boldsymbol{I}_0[n,m] \\
& - (\boldsymbol{I}_0[f_r, f_c] + \boldsymbol{I}_0[0:f_r-1, 0:f_c-1] - \boldsymbol{I}_0[0:f_r-1, f_c] - \boldsymbol{I}_0[f_r, 0:f_c-1]) \\
& + (\boldsymbol{I}_1[f_r, f_c] + \boldsymbol{I}_1[0:f_r-1, 0:f_c-1] - \boldsymbol{I}_1[0:f_r-1, f_c] - \boldsymbol{I}_1[f_r, 0:f_c-1]) \\
& + \boldsymbol{P}_{\boldsymbol{u}_1}[1:f_r, 1:f_c] + \boldsymbol{P}_{\boldsymbol{b}_1}[f_r, f_c]
\end{aligned} \tag{4.3}$$

where $I_k[a_1:a_2, a_3:a_4]$ stands for a sub-matrix of I_k defined by the rows from a_1 to a_2 and the columns from a_3 to a_4; and I_k is of size $(n \times m)$.

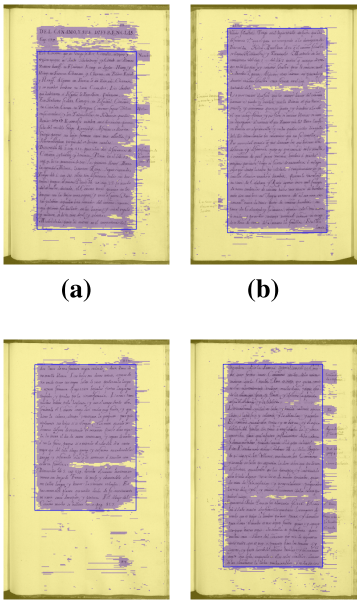
Under this approach, time to compute min value is reduced from ≈ 25 seconds to ≈ 0.06 seconds, which is enough for current application.

4.3 Experiments

Experiments have been conducted over the selected corpus to obtain the model parameters, such as features to train CRF models, window size of those features, and training algorithm, among others. Parameters search is not exhaustive since main goal of this stage is not to get the best model, but to demonstrate the interactive algorithm features. All the results have been evaluated using the methods presented in Section 2.5.

4.3.1 Conditional Random Fields

CRF performance is highly dependent on features selection; because of that, some values of image zoom (Z), window size (W), grid size (G) were tested. AROW [5] training algorithm was selected to train the CRF model because is proved to be very fast and results are similar to L-BFGS [2] or others. Results are presented in Appendix 6.2, Table 6.2 for reference. In view of the results, the following parameters are selected to train the final model: Z=0.3, W=33 and G=3. Quantitative results are shown in Table 4.1 and some examples of qualitative results in Figure 4.3, where all pixels inside the blue rectangle are labeled as "paragraph" in the ground-truth.



(a) **(b)** **(c)** **(d)**

Figure 4.3: CRF's qualitative results, pixels classified as background in yellow, pixels classified as paragraph in purple. Ground-truth rectangle in blue. Pages a) 0944, b) 0945, c) 0948 and d) 0958.

Table 4.1: CRF's site level quantitative results.

Page	Precision	Recall	F1
0944	0.908	0.914	0.907
0945	0.953	0.953	0.953
0946	0.958	0.956	0.956
0947	0.951	0.952	0.951
0948	0.949	0.951	0.950
0956	0.894	0.896	0.891
0957	0.946	0.945	0.945
0958	0.931	0.927	0.927
0959	0.952	0.952	0.952
0960	0.955	0.954	0.954
0961	0.950	0.950	0.950
0962	0.909	0.921	0.910
0963	0.919	0.914	0.916
0964	0.909	0.910	0.908
0965	0.942	0.942	0.942
0966	0.947	0.947	0.947
0967	0.945	0.946	0.945
Average	0.936	0.936	0.936

4.3.2 Connected Components Labeling (CCL) Approach

Connected components algorithms based on morphological operations are a simple method to detect connected objects or regions in binary images. A simple version of this kind of algorithms [4] is implemented as a point of comparison for the proposed method. Thus, all adjacent sites classified as "paragraph" by the CRF model are grouped; then, we search for the minimum rectangle where all sites of the same group fits; and finally, based on user experience, only the biggest rectangle is selected. See Figure 4.4 for some qualitative examples and Table 4.2 for quantitative results.

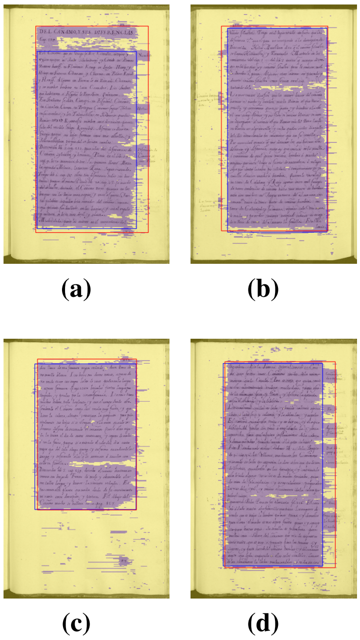


Figure 4.4: CCL results examples, red line. Ground-truth added for reference, blue line. Pages a) 0944, b) 0945, c) 0948 and d) 0958.

Table 4.2: CCL quantitative results.

Page	MatchScore	GoSR
0944	0.862	0.768
0945	0.958	0.926
0946	0.929	0.877
0947	0.969	0.944
0948	0.965	0.937
0956	0.873	0.782
0957	0.956	0.921
0958	0.923	0.863
0959	0.984	0.971
0960	0.932	0.879
0961	0.967	0.942
0962	0.861	0.765
0963	0.970	0.944
0964	0.878	0.787
0965	0.948	0.908
0966	0.929	0.873
0967	0.950	0.912
Average	0.933	0.882

4.3.3 Prior-Probability Approach

Prior-Probability is used to estimate the best "paragraph" coordinates, i.e. we maximize Eq. (3.6) over all $(\boldsymbol{u}_k, \boldsymbol{b}_k)$ in the image range using a brute force approach and the methods explained in Section 4.2. See Figure 4.5 for some qualitative results examples, and Table 4.3 for quantitative results.

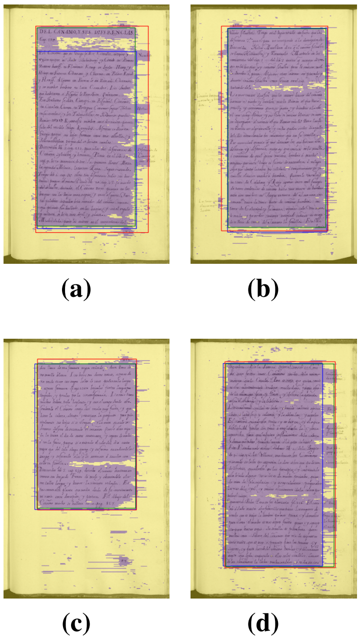


Figure 4.5: Proposed method results example, blue line for ground-truth, red line for connected components labeling approach, green line for proposed approach. Pages a) 0944, b) 0945, c) 0948 and d) 0958.

Table 4.3: Proposed method quantitative results.

Page	MatchScore	GoSR
0944	0.924	0.862
0945	0.975	0.956
0946	0.988	0.977
0947	0.968	0.943
0948	0.983	0.969
0956	0.882	0.791
0957	0.978	0.960
0958	0.930	0.875
0959	0.969	0.945
0960	0.988	0.978
0961	0.973	0.952
0962	0.927	0.869
0963	0.937	0.888
0964	0.934	0.878
0965	0.969	0.944
0966	0.974	0.951
0967	0.966	0.940
Average	0.957	0.922

4.3.4 Interactive Approach

User is allowed to change system hypothesis by a simple click over the image. This feedback is decoded and a new hypothesis is presented to the user. Number of clicks needed by the user to define the main paragraph and new hypothesis are recorded. Quantitative results are presented on Table 4.4 along with the number of clicks performed by an user.

Table 4.4: Post-user-feedback quantitative results.

Page	MatchScore	GoSR	# Clicks
0944	0.978	0.960	1
0945	0.975	0.956	0
0946	0.988	0.977	0
0947	0.968	0.943	0
0948	0.983	0.969	0
0956	0.974	0.952	2
0957	0.978	0.960	0
0958	0.983	0.969	1
0959	0.969	0.945	0
0960	0.988	0.978	0
0961	0.973	0.952	0
0962	0.975	0.955	1
0963	0.956	0.921	1
0964	0.971	0.947	1
0965	0.969	0.944	0
0966	0.974	0.951	0
0967	0.966	0.940	0
Average	0.975	0.954	—

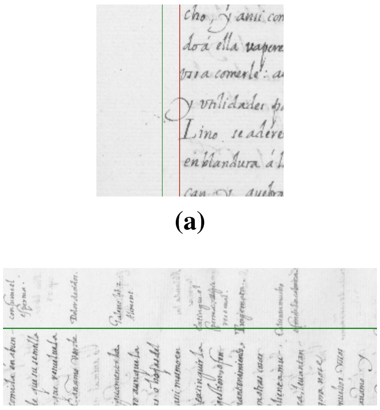
4.4 Results Discussion

CRF model performed an average of 93.6% F1-score. Although the set of features selected were very elemental (only site color intensity and position) and a non-exhaustive parameter search was done, results are very promising for further stages. Misclassification of marginalia and title zones could be because color intensity is a feature more to identify text than to identify the different layout zones. That means that layout zone classification lays mostly on position feature. Also, only two classes have been used for training the CRF model. Although this model provides posterior probability needed for the interactive model, feature selection needs to be improved for further stages of this project.

CCL is highly dependent on geometric distribution of posterior probability from the CRF model and, after segmentation, there is no easy and general way to improve raw results. For this reason, this approach is used only as a point of comparison for the proposed method. Performance computed by GoSR and MatchScore are pretty different (93.2% and 83.2% respectively on average), because the first method is most a measure of how similar are the polygons, but the second method finds how much of the information is on both polygons. Notice that almost all marginalia and title zones have been classified as paragraph. This is due to the high dependence on geometric distribution.

Prior-probability plays a main role in next approach; most of the marginalia zones are not longer classified as a paragraph, and rectangle boundary is stretched to text boundary, which relies on a average 4% GoSR improvement over the CCL approach. This is up to a 13% improvement in complex cases (see Table 4.3).

Finally, methods studied in the bibliography and the method proposed in this work still have errors that must be fixed manually by some human. The interactive approach based on the proposed method provides the framework to help user to fix those errors. Under that premise, not only the system performance must be computed, but the user effort as well. On this seventeen pages corpus, 65% of the times user made no changes on the first provided hypothesis. Thus, hypothesis is good enough to identify the paragraph. On the other hand, 30% of the times the user performs only one click in order to fix errors in the hypothesis. Finally, only in one case two clicks were needed. Performance average is 97.5% and 95.3% on MathScore and GoSR methods respectively; notice that 100% was not reached in any case because users have a different concept of how much the border of the zone needs to be to the border of the text, or cases where two o more blocks cannot be divided by a single line. See examples on Figure 4.6.



(a)

(b)

Figure 4.6: Examples of different zone boundaries provided by different users. a) small section of a character is out due to an adjustment to text border, b) single horizontal line cannot divide upper and bottom zones.

Bibliography

[1] Bosch, V., Bordes-Cabrera, I., Muñoz, P. C., Hernández-Tornero, C., Leiva, L. A., Pastor, M., Romero, V., Toselli, A. H., and Vidal, E. (2014). Computer-assisted Transcription of a Historical Botanical Specimen Book : Organization and Process Overview Categories and Subject Descriptors. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, pages 125–130, Madrid, Spain.

[2] Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM J. Sci. Comput.*, 16(Sept. 1995):1190–1208.

[3] Crow, F. C. (1984). Summed-area tables for texture mapping. *ACM SIGGRAPH Computer Graphics*, 18(3):207–212.

[4] Gonzalez, R. C. and Woods, R. E. (2008). *Digital Image Processing*. Prentice Hall, 3rd edition.

[5] Kulesza, A., Crammer, K., and Dredze, M. (2009). Adaptive Regularization of Weight Vectors. *Advances in Neural Information Processing Systems*, 22:414–422.

[6] Nielsen, J. (1993). *Usability Engineering*. Morgan Kaufmann Publishers Inc.

[7] Pletschacher, S. and Antonacopoulos, A. (2010). The PAGE (Page Analysis and Ground-truth Elements) format framework. *Proceedings - International Conference on Pattern Recognition*, pages 257–260.

[8] Viola, P. and Jones, M. (2001). Robust real-time object detection. *International Journal of Computer Vision*, 57(2):137–154.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

Chapter Outline

5.1 Conclusions

For the human beings is very easy to identify a document structure, but the number of variations in the documents (quality, author style, type of paper used, etc.) and the complexity of the structure itself makes Layout Analysis a very hard problem to solve by a computer system. Contributions of this work in the aim to solve Layout Analysis problem and other conclusions are presented as follows:

- A new method for Layout Analysis for ancient documents is presented. It was also shown that the method works, at least, to extract the main paragraph of the page, which is commonly where most of the information remains.
- Inclusion of the prior-probability in the model shown a direct improvement over the CCL method, without any heuristics.
- The interactive approach provides to the user the ability to fix any error produced in the classification stage. The number of clicks needed to fix the errors has been reduced; consequently, time expended in the task is reduced as well.
- *Geometric* and *logical* layout sub-problems have been merged in the proposed method; this allows HTR steps to take into account syntactic differences between different layout-zones.
- The proposed method does not need a binarization step, this makes the method independent of one of the most difficult steps of state-of-the-art methods. In consequence, the new method decrements the potential sources of error.

5.2 Future Work

As mentioned in Chapter 1, in this paper only the first stage of the project is presented; then, the following extensions could be performed:

- **Remove single zone constrain:** currently only main paragraph zone is detected; further stages of the project require to include any type of layout zones, indeed, marginalia, titles, catch-words, etc.

- **Improve CRF's features:** features used to train the CRF model are very weak, and this reduces system performance and generalization; a good starting point would be to explore the use of Convolutional Neural Networks to extract these features automatically.
- **Include non-deterministic feedback:** although deterministic feedback is useful under Layout Analysis conditions, a non-deterministic feedback would make the system more flexible.
- **Replace brute force algorithm:** even though, the brute force algorithm performs properly to search best hypothesis on a single-zone case, in order to include any type of layout zones we need to use a more efficient algorithm, like Gradient Descent.
- **More experiments on complex corpora:** in order to test completely the proposed methodology and compare it to state-of-the-art methods, an extensive set of experiments needs to be conduced; thus, complementary experiments on complete book corpus [2] and the latest contest corpus [1] are planned.

Bibliography

[1] Antonacopoulos, A., Clausner, C., Papadopoulos, C., and Pletschacher, S. (2015). ICDAR2015 Competition on Recognition of Documents with Complex Layouts –. *13th International Confrence on Document Analysis and Recognition - ICDAR'15*, pages 1151–1155.

[2] Bosch, V., Bordes-Cabrera, I., Muñoz, P. C., Hernández-Tornero, C., Leiva, L. A., Pastor, M., Romero, V., Toselli, A. H., and Vidal, E. (2014). Computer-assisted Transcription of a Historical Botanical Specimen Book : Organization and Process Overview Categories and Subject Descriptors. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, pages 125–130, Madrid, Spain.

CHAPTER 6

APPENDIX

Chapter Outline

6.1 Corpus Distribution Table

Table 6.1: Corpus Pages Distribution

Name	ID	used to
Mss_003357_0944_pag-811[843]	0944	test
Mss_003357_0945_pag-812[844]	0945	test
Mss_003357_0946_pag-813[845]	0946	test
Mss_003357_0947_pag-814[846]	0947	test
Mss_003357_0948_pag-815[847]	0948	test
Mss_003357_0956_pag-823[855]	0956	test
Mss_003357_0957_pag-824[856]	0957	test
Mss_003357_0958_pag-825[857]	0958	test
Mss_003357_0959_pag-826[858]	0959	test
Mss_003357_0960_pag-827[859]	0960	test
Mss_003357_0961_pag-828[860]	0961	test
Mss_003357_0962_pag-829[861]	0962	test
Mss_003357_0963_pag-830[862]	0963	test
Mss_003357_0964_pag-831[863]	0964	test
Mss_003357_0965_pag-832[864]	0965	test
Mss_003357_0966_pag-833[865]	0966	test
Mss_003357_0967_pag-834[866]	0967	test
Mss_003357_0158_pag-053[057]	0158	train
Mss_003357_0159_pag-054[058]	0159	train
Mss_003357_0161_pag-056[060]	0161	train
Mss_003357_0162_pag-057[061]	0162	train
Mss_003357_0163_pag-058[062]	0163	train
Mss_003357_0176_pag-071[075]	0176	train
Mss_003357_0177_pag-072[076]	0177	train
Mss_003357_0178_pag-073[077]	0178	train
Mss_003357_0179_pag-074[078]	0179	train
Mss_003357_0180_pag-075[079]	0180	train
Mss_003357_0181_pag-076[080]	0181	train
Mss_003357_0182_pag-077[081]	0182	train
Mss_003357_0183_pag-078[082]	0183	train
Mss_003357_0184_pag-079[083]	0184	train
Mss_003357_0185_pag-080[084]	0185	train
Mss_003357_0186_pag-081[085]	0186	train
Mss_003357_0187_pag-082[086]	0187	train
Mss_003357_0188_pag-083[087]	0188	train
Mss_003357_0189_pag-084[088]	0189	train
Mss_003357_0190_pag-085[089]	0190	train
Mss_003357_0191_pag-086[090]	0191	train
Mss_003357_0192_pag-087[091]	0192	train

6.2 CRF's parameter search.

Table 6.2: CRF's parameters search results. Z=zoom, W=window, G=granularity, FT= feature extraction time, TrT= Training time, TeT= Test time, P= Precision, R= Recall, F1= F1-score.

#	Z	W	G	FT[s]	TrT[s]	TeT[s]	P	R	F1
1	0.1	9	3	4.916	16.822	1.150	0.899	0.900	0.897
2	0.1	9	9	1.151	0.074	0.117	0.868	0.865	0.859
3	0.1	9	12	0.937	0.035	0.062	0.878	0.874	0.872
4	0.1	17	3	20.448	16.054	0.962	0.894	0.884	0.882
5	0.1	17	9	3.619	0.074	0.092	0.895	0.886	0.884
6	0.1	17	12	2.727	0.019	0.052	0.852	0.833	0.828
7	0.1	33	3	23.603	0.997	0.792	0.843	0.811	0.817
8	0.1	33	9	3.920	0.044	0.084	0.858	0.796	0.804
9	0.1	33	12	2.815	0.015	0.043	0.850	0.714	0.714
10	0.1	65	3	32.006	1.050	0.704	0.684	0.659	0.668
11	0.1	65	9	5.078	0.064	0.067	0.820	0.667	0.687
12	0.1	65	12	3.528	0.029	0.037	0.863	0.712	0.742
13	0.2	9	3	18.714	142.845	4.559	0.866	0.868	0.866
14	0.2	9	9	2.714	0.825	0.523	0.852	0.852	0.847
15	0.2	9	12	2.127	0.290	0.261	0.882	0.885	0.882
16	0.2	17	3	88.976	122.980	4.235	0.866	0.863	0.858
17	0.2	17	9	11.068	5.450	0.422	0.863	0.858	0.851
18	0.2	17	12	6.758	0.219	0.225	0.895	0.896	0.892
19	0.2	33	3	105.599	81.756	4.396	0.905	0.901	0.901
20	0.2	33	9	13.263	1.048	0.410	0.895	0.888	0.887
21	0.2	33	12	8.095	0.246	0.214	0.896	0.889	0.888
22	0.2	65	3	173.887	76.256	3.759	0.853	0.848	0.850
23	0.2	65	9	20.629	0.477	0.393	0.837	0.812	0.819
24	0.2	65	12	12.392	0.287	0.220	0.829	0.803	0.808
25	0.3	9	3	41.491	457.521	12.270	0.829	0.832	0.829
26	0.3	9	9	5.174	19.041	1.207	0.859	0.860	0.859
27	0.3	9	12	3.271	1.414	0.554	0.866	0.868	0.867
28	0.3	17	3	195.334	341.149	10.868	0.881	0.882	0.880
29	0.3	17	9	23.479	17.754	1.117	0.880	0.883	0.880
30	0.3	17	12	13.886	2.578	0.563	0.885	0.888	0.884
31	0.3	33	3	24.153	20.972	9.582	0.936	0.937	0.936
32	0.3	33	9	29.088	15.112	0.973	0.898	0.898	0.896
33	0.3	33	12	16.788	1.933	0.527	0.895	0.892	0.890
34	0.3	65	3	396.537	190.954	9.099	0.901	0.898	0.899
35	0.3	65	9	45.363	2.987	0.927	0.891	0.881	0.883
36	0.3	65	12	26.538	6.580	0.502	0.881	0.870	0.870

NOMENCLATURE

AROW	Adaptive Regularization of Weights
BFGS	Broyden–Fletcher–Goldfarb–Shanno
CCL	Connected Components Labeling
CRF	Conditional Random Field
DLA	Document Layout Analysis
GMM	Gaussian Mixture Model
IPR	Interactive Pattern Recognition
L-BFGS	Limited-memory BFGS
PCA	Principal Components Analysis
PDF	Probabilistic Distribution Function
PR	Pattern Recognition