A texture-based pixel labeling approach for historical books

Maroua Mehri, Petra Gomez-Krämer, Pierre Héroux, Alain Boucher, Rémy Mullot

Pattern Analysis and Applications manuscript No.
(will be inserted by the editor)

A Texture-based Pixel Labeling Approach for Historical Books

Maroua Mehri · Petra Gomez-Krämer · Pierre Héroux · Alain Boucher · Rémy Mullot

Received: date / Accepted: date

Abstract Over the last few years, there has been tremendous growth in the automatic processing of digitized historical documents. In fact, finding reliable systems for the interpretation of ancient documents has been a topic of major interest for many libraries and the prime issue of research in the document analysis community. One important challenge is to refine well-known approaches based on strong *a priori* knowledge (*e.g.* the document image content, layout, typography, font size and type, scanning resolution, image size, *etc.*). Nevertheless, a texture analysis approach has consistently been chosen to segment a page layout when information is lacking on document structure and content. Thus, in this article a framework is proposed to investigate the use of texture as a tool for automatically determining homogeneous regions in a digitized historical book and segmenting its contents by extracting and analyzing texture features independently of the layout of the pages. The proposed framework is parameter-free and applicable to a large variety of ancient of books. It does not assume *a priori* information regarding document image content and structure. It consists of two phases: a texture-based feature extraction step and unsupervised clustering and labeling task based on the consensus clustering, hierarchical ascendant classification, and nearest neighbor search algorithms. The novelty of this work lies in the clustering of extracted texture descriptors to find automatically homogeneous regions, *i.e.* graphic and textual regions, by using the clustering approach on an entire book instead of processing each page individually. Our framework has been evaluated on a large variety of historical books and achieved promising results.

Keywords Digitized historical books · Pixel labeling · Texture · Autocorrelation · Multiresolution · Purity per block

1 Introduction

The development of the Internet and electronic publishing, the prospects offered by the standardization of documentary techniques and broadcast media, and increased storage capacity and transmission rates, raise questions and pose specific challenges concerning the preservation and reproduction of historical collections. Thus, in order to guarantee a lasting preservation of historical collections and to provide a world-wide access to material which needs to be protected from too frequent handling, libraries have conducted large digitization programs with cultural heritage documents. The European[1] and American[2] Ministries of Culture support digitization programs and encourage the development of digital libraries which offer new services such as on-line consulting of ancient documents, fragile

Maroua Mehri · Petra Gomez-Krämer · Alain Boucher · Rémy Mullot
L3I, University of La Rochelle, Avenue Michel Crépeau, 17042, La Rochelle, France
Tel.: +33-5 46 45 82 62
Fax: +33-5 46 45 82 42
E-mail: {maroua.mehri, petra.gomez, alain.boucher, remy.mullot}@univ-lr.fr

Pierre Héroux
LITIS, University of Rouen, Avenue de l'Université, 76800, Saint-Etienne-du-Rouvray, France
Tel.: +33-2 32 95 50 11
Fax: +33-2 32 95 50 22
E-mail: pierre.heroux@univ-rouen.fr

[1] `http://www.culture.gouv.fr/culture/mrt/numerisation/`
[2] `http://www.archives.gov/digitization/`

books and rare collections, information retrieval, *etc.* Thus, new technologies have revolutionized the world of librarianship and printing [1]. Numerous research projects, such as HisDoc[3], DocExplore[4], Europeana[5], DEBORA[6], BAMBI[7], MADONNE[8], NaviDoMass[9], Passe-Partout[10] and GRAPHEM[11] are looking at the digitization of European and American ancient heritage resources. The French digital library Gallica[12], the British library[13] and the John F. Kennedy library[14], have been established for the purpose of preserving and exploiting this cultural heritage. For instance, the European project DEBORA aims to develop networked libraries by improving accessibility to the 16^{th} century books of Italy, France and Portugal [2, 3]. One of the aims of DocExplore is to construct a document analysis framework which provides computer-based access and analysis of historical manuscripts. The aim of the HisDoc project is to design generic processing approaches and tools for historical manuscripts which are independent of the scripting language [4]. The goal of the MADONNE project is to develop a toolkit that can be used to index heritage documents and categorize book pages [5]. A project on indexing handwritten historical manuscripts[15] has been developed by Rath *et al.* and supported by the "Center for Intelligent Information Retrieval" at the University of Massachusetts Amherst[16] and the National Science Foundation[17] [6]. They are using a part of the George Washington collection[18] at the library of Congress[19] to evaluate their technique.

However, a lack of comprehensive and strategic management tools has become an obstacle to optimizing the exploitation of heritage documents. There has been an increase in special needs for information retrieval in digital libraries and document layout analysis [7–9]. Le Bourgeois *et al.* highlighted the need to design "intelligent" digitizers which can limit manual intervention and perform easy and high quality digitization of document images [2]. Therefore, with the support of the ANR (French National Research Agency[20]) and the collaboration of many research laboratories, we are working on a project named DIGIDOC (Document Image diGitisation with Interactive DescriptiOn Capability)[21]. Specifically, the aim of the DIGIDOC project is to develop new ways of interacting with scanners and new tools for analyzing documents throughout the acquisition process, from scanning the document to knowledge representation and management of the digitized ancient document content.

Thus, to achieve better interaction with scanners, we need to design a computer-aided categorization tool, able to classify digitized book pages according to several criteria, mainly the layout structure or typographic characteristics of their content. For this purpose, we propose to characterize digitized pages of ancient books with a set of regions of homogeneous texture and their topological relationships that helps modeling the layout structure, separating text from non-text regions, partitioning or categorizing pre-localized text blocks into columns, headings, paragraphs, lines, words, notes (head-notes and foot-notes) and abstracts, *etc.* Our goal is to extract as automatically as possible textural features that segment an ancient book or a collection of historical documents into spatially disjoint homogeneous regions or similar content regions and characterize its content according to a topological representation of homogenous regions, without formulating a hypothesis concerning the document structure or layout (*e.g.* column layout) or the typographical parameters (*e.g.* font size and type) of the document. By characterizing each digitized page of ancient book with a set of regions of homogeneous texture and their topological relationships, a model or a signature can be designed for each book page. The obtained signatures help deducing the similarities of book page structure or layout and/or content. Indeed, the structure and content of book pages can be compared and subsequently the designed signatures which model structure and content book pages

[3] https://diuf.unifr.ch/main/hisdoc/

[4] http://www.docexplore.eu

[5] http://www.europeana.eu

[6] http://cordis.europa.eu/libraries/en/projects/debora.html

[7] http://www.ilc.cnr.it/viewpage.php/sez=ricerca/id=97/vers=ing

[8] http://madonne.univ-lr.fr

[9] http://navidomass.univ-lr.fr

[10] http://www3.unil.ch/BCUTodai/app/todaiGetIntro.do?uri=todaiInfo&page=todaiLogo.html

[11] http://liris.cnrs.fr/graphem/

[12] http://gallica.bnf.fr

[13] http://www.bl.uk

[14] http://www.jfklibrary.org/

[15] http://ciir.cs.umass.edu/irdemo/hw-demo/

[16] http://ciir.cs.umass.edu/

[17] http://www.nsf.gov/

[18] http://memory.loc.gov/ammem/gwhtml/gwhome.html

[19] http://www.loc.gov/

[20] http://www.agence-nationale-recherche.fr/en/

[21] The DIGIDOC project is referenced under $ANR-10-CORD-0020$. For more details, http://www.agence-nationale-recherche.fr/en/anr-funded-project/?tx_lwmsuivibilan_pi2[CODE]=ANR-10-CORD-0020

can be classified according to their structure and content. Thus, book pages with similar structure/layout or/and content can be classified or grouped. Figure 1 provides an overview of the objectives of this work.

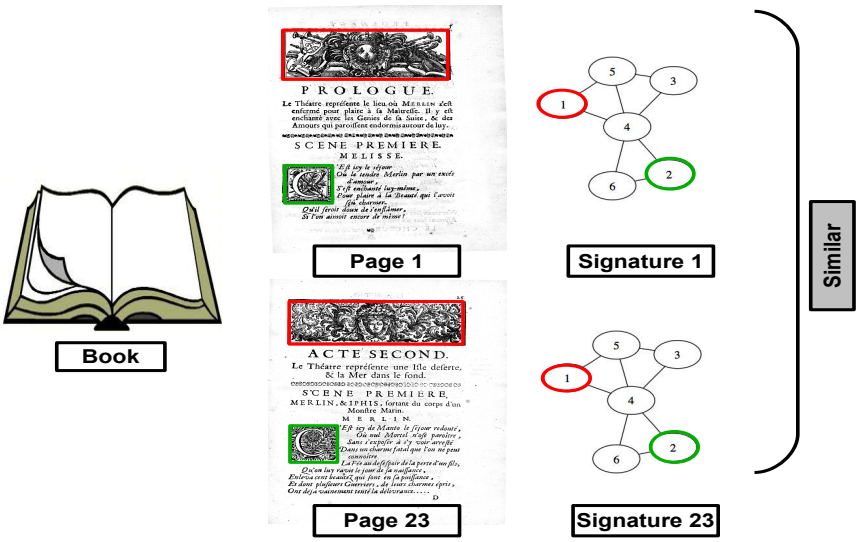


Fig. 1: Overview of the objectives of this work.

Recently, the issues of document image analysis have been considered as texture segmentation and classification [10]. Moreover, some similarities of document content type have been deduced from many book pages [11, 12]. In addition, based on the assumption that texture can characterize a document content type which is usually repeated on many pages of the same book, we propose a framework that works on entire book scale instead of processing each page individually. Thus, in this article by combining several points related to texture-based segmentation that have been reported separately in the literature particularly on synthetic, medical and natural images, we attempt to represent a book page using a set of homogeneous blocks defined by similar texture attributes and their topology. Indeed, a pixel labeling framework for digitized historical books is proposed in this article. The proposed framework ensures the pixel-based characterization of the content of an entire book by extracting and analyzing the texture information from each page. It is automatic, parameter-free and can be adapted to all kinds of books. It is independent of document layout, typeface, font size, orientation, image size, digitizing resolution and intensity, *etc.* It is also insensitive to noise. Moreover, it does not require any manual inspection or *a priori* knowledge regarding document image content and structure or layout.

The originality of our contribution lies in the automatic analysis of some characteristics of book pages (regarding their content and/or layout) to find homogeneous regions (*i.e.* graphic and textual regions) by extracting and clustering texture features on an entire book instead of processing each page individually, with no assumption concerning the book page structure or layout (*e.g.* column layout) or the typographical or graphical properties (*e.g.* font size and type) of the digitized book pages. Indeed, even if the typographical or graphical features are not known in advance, they can be captured by exploiting the regularities of the associated textures through the whole book pages. So, in a first step, a clustering of texture features which are extracted from a subsampling in the entire book aims at identifying the texture information that is present in book pages. The clustering method that is applied has the ability to automatically determine the number of clusters or homogeneous regions. This knowledge is then used in a second step to segment each book page individually.

The remainder of this article is structured as follows: Section 2 reviews related works on historical document segmentation. This section also gives an insight to texture-based image segmentation methods, their use in the context of historical document image segmentation, and clustering approaches when the number of clusters is not known in advance. In Section 3, the proposed framework for the characterization of the content of an entire book by extracting and analyzing the texture information from each page is described. In Section 4, we outline the experimental protocol by describing the experimental corpus, the defined ground truth and the used clustering and classification metrics for an evaluation of accuracy. To evaluate the performance of the proposed framework,

several clustering and classification metrics are computed and discussed in Section 5. Qualitative results are also given to demonstrate its performance. Our conclusions and future work are presented in Section 6.

2 Related works

Historical documents have many particularities such as a large variability of the page layout: noise and degradation (caused by copying, scanning and aging), page skew, complicated layout, random alignment, specific fonts, the presence of embellishments, variations in spacing between the characters, words, lines, paragraphs and margins, overlapping object boundaries, and the superimposition of information layers (stamps, handwritten notes, noise, back-to-front interference, *etc.*) [9, 13]. Figure 2 illustrates some particularities of historical documents.

Fig. 2: Illustration of some particularities of historical documents (*e.g.* superimposition of information layers (stamps, handwritten notes, noise, back-to-front interference, *etc.*), page skew *etc.*)

Thus, processing this kind of document is not a straightforward task and usually includes several stages: pre-processing, analysis, characterization and recognition [14]. A variety of approaches based on *a priori* knowledge of segmentation of ancient documents have been proposed in the literature.

In the context of the Philectre project, André *et al.* extracted drop caps (*i.e.* the first letters at the beginning of a paragraph) and text regions from the foreground layer of the analyzed document using edge detection for dark regions (*i.e.* low mean gray level) followed by a thresholding phase that takes into account the local and global adjacent neighboring pixels [15]. Secondly, they used a vertical and horizontal projection phase based on a few thresholds (average height and line spacing) and specified rules for the extraction of columns and lines. Finally, they performed a connected-component labeling based on a defined projection interval. This approach was based on a knowledge-acquisition phase to determine the relevant characteristics of a sample set of historical documents.

A well-researched survey dedicated to text line segmentation of historical documents was presented by Likforman-Sulem *et al.* [16]. Most of the existing approaches are based on connectivity features, projection (XY-CUT) [17], Run Length Smearing Algorithm (RLSA) [18] and Hough techniques [19] which are suitable for clear lines. These approaches require thresholds to define inter-line or inter-block distances and adjustments for character alignment and line justification. In addition, a pre-processing phase is necessary to remove background noise (superfluous information appearing from the verso) and non-textual regions. Belaïd and Ouwayed proposed a multi-oriented text line extraction approach of ancient Arabic documents based on image meshing technique,

energy distribution of Cohen's class and connected component analysis techniques [20]. They defined a few rules depending on the orientations presented in their documents. Nikolaou *et al.* proposed adaptive RLSA and skeleton segmentation paths for text line, word and character segmentation of historical and degraded machine-printed documents [21]. Although the proposed algorithm worked efficiently for a wide variety of degraded documents, they defined several thresholds in the used segmentation techniques.

Without a given model of the layout for medieval manuscripts, Le Bourgeois *et al.* proposed a data-driven layout segmentation approach based on connected components [2]. Their method required several parameters, estimated thresholds determined by the user and stored in the model, and also required several pre-processing steps: a binarization step, an image noise reduction filter and the frame removal task based on mathematical morphology [22–24]. To localize the main body of the text from Arabic manuscripts, they also estimated the average size of text symbols by computing the average size of all connected components. Then, they computed a text probability value for each connected component. Finally, they estimated an automatic threshold for each profile (horizontal and vertical) obtained from the entire image. They considered their algorithm to be a useful tool to detect the main body of a text, even for Latin manuscripts, but it did not work with large annotation areas in the margins.

Likforman-Sulem presented an overview of different approaches used with ancient documents to separate text and graphic regions [14]. For instance, Granado *et al.* extracted text and graphic regions from ancient books using mathematical morphology [23]. In the case of medieval documents, an accurate morphological analysis of connected components was investigated to separate text/non-text regions [15]. Mengucci and Granado presented a mathematical morphology approach for the segmentation of figures/text characters from pages of Renaissance books (16^{th} century) [25]. These approaches require a selection of morphological parameters and filter thresholds (*e.g.* directional closings, size of the structuring element, *etc.*) based on given heuristics (*e.g.* size and appearance of connected components).

Le Bourgeois and Emptoz, as part of the European project DEBORA, analyzed and segmented ancient books using morphology, texture and a bottom-up model [2, 3]. They succeeded in segmenting the physical layout except for some errors, which appeared when there were lines of text that were touching, due to a lack of *a priori* knowledge and the highly complex layout of the document. They separated text from non-text regions by combining texture, component shapes and alignments. The recognition of drop caps and strips was based on a *a priori* model designed using information about size, location, surrounding neighborhoods, *etc.* Ramel *et al.* evaluated various traditional methods used for segmentation of historical printed documents [26]. They highlighted the limits of the traditional methods to segment historical documents. Thus, they proposed a hybrid segmentation algorithm based on connected components for user-driven page layout analysis of historical printed books. The proposed algorithm used two maps: a shape map for foreground information analysis based on connected component technique and a background map for white area analysis. Then, the classification of the extracted blocks by using connected component analysis technique, was built according to scenarios defined by the user.

Therefore, based on strong *a priori* knowledge such as the repetitiveness of document structure in a corpus (*i.e.* the blocks shape, the uniformity in horizontal and/or vertical spacings, and/or the assumptions about textual and graphical characteristics such as font size, *etc.*), the existing approaches are not effective. In addition, there are certain limitations of these methods. Firstly, several parameters and thresholds must be adjusted. Secondly, those methods are sensitive to noise and not robust to slanted texts. The drawbacks of those approaches are their dependence on the font size, character space, character size, inter-character spacing, document orientation and line and column space, *etc.* Furthermore, the performance of this family of document structure analysis approaches depends on the particular layout and document idiosyncrasies. Finally, for complex and degraded historical document images, it is a difficult task to set empirical rules, domain specific constraints, and thresholds.

Hence, Crasson and Fekete highlighted the real need for automatic processing of digitized historical documents (document layout analysis and text/non-text separation) to facilitate the analysis and navigation in the corpus of ancient manuscripts [27]. Kise stated that the analysis of pages with constrained layouts (*e.g.* rectangular and Manhattan) and clean document images has almost been solved while the analysis of ancient documents is still an open problem due to their particularities (*e.g.* noise and degradation, presence of handwriting, overlapping layouts and great variability of the page layout, *etc.*). He also precised that the most relevant methods used to analyze pages with overlapping or unconstrained layouts are based on signal properties of page components by investigating texture-based features and techniques [28]. Thus, in this work we explore various aspects of the texture features in historical documents in order to assist the analysis of images by characterizing a document layout through a set of homogeneous regions. Given that there is significant degradation and with no hypothesis concerning the document structure/layout or the typographical parameters of the document, the use of texture analysis techniques for historical document segmentation has become an appropriate choice. In order to ensure a distinction between different text fonts and various kinds of graphics, two assumptions are made [29]. First, the textual regions in

a digitized document are considered as textured areas, while its non-text content is considered as regions with different textures. Secondly, text with a different font is also distinguishable.

In this work, essentially we address the problem of the analysis and characterization of historical documents. These phases specifically relate to two important tasks:

- *The feature extraction methods* assign to each analyzed image a visual signature that describes its content.
- *The feature space structuring methods* partition the analyzed image into regions which have homogeneous characteristics and similar properties with respect to the extracted features.

2.1 Feature extraction methods

The feature extraction and analysis techniques provide important information about similar and homogeneous content regions that ensure the characterization of the layout and the structure. Chen and Blostein claimed that the choice of document features depends on the document recognition stage and categorized them into three classes [30]:

- *Image features* are extracted from the analyzed image as a signature computed on the entire image and based on color, texture and/or shape.
- *Structural features* are deduced from a physical or logical layout analysis.
- *Textual features* are obtained from Optical Character Recognition (OCR) or from documents.

In this work, we are interested in the image features which are the most suitable for the segmentation of a historical page layout [31–33]. Image feature extraction approaches have been divided into three main types [34]:

- *Global approaches* consider the analyzed image as a signature computed on the entire image and based on color, texture and/or shape.
- *Local approaches* analyze local descriptors around different interest points detected on the analyzed image.
- *Spatial approaches* represent regions and their neighboring relationships by a graph.

Below, the global approaches to feature extraction are detailed because they are the most suitable and widely used for the analysis of digitized historical documents [31–33]. Moreover, they are independent of topological information and the region stability of segmentation algorithms. In this context, a class of segmentation methods based on texture feature extraction and analysis has been proposed recently and is considered as an alternative for complex document structure analysis [35]. Segmentation methods based on texture analysis are pixel-based and do not require either *a priori* knowledge of the document layout or *a priori* information on the semantic and physical characteristics of the document category. These methods generally characterize documents with a complex layout in order to segment their content into homogeneous blocks based on extracted textural descriptors. In addition, they do not assume that information about the layout is available and they can be used with any kind of document (*e.g.* check, manuscript, printed document, journal, newspaper, *etc.*). A general definition of texture has been given as a measure of the variation in intensity, measuring properties such as smoothness, coarseness and regularity [36]. Texture has also been defined as a suitable measure for the analysis of the block contents of the physical layout [35]. Texture analysis methods have been used in image segmentation to extract textural characteristics.

Texture-based segmentation methods can be classified into two categories [37]:

- *Region-based approaches* are used to identify uniform, similar or homogeneous textured regions.
- *Boundary-based approaches* are used to analyze the differences in texture in adjacent or neighborhood surrounding regions.

Since our objective is to find regions with similar textural content, we opt for a region-based approach exploiting textural features. A variety of approaches for characterizing image texture have been proposed. Jain *et al.* demonstrated the effectiveness of a texture-based approach for different types of document image processing tasks (*i.e.* text-graphic separation, address-block location, *etc.*) [38]. Texture feature extraction and analysis methods is classified into four categories [39, 40]:

- *Statistical methods* are used to analyze the spatial distribution of gray levels by computing local indices in the image and deriving a set of statistics from the distribution of the local features. The Grey Level Co-occurrence Matrix (GLCM) is one of the standard statistical segmentation methods that are based on texture analysis [41].
- *Geometrical methods* are used to describe intricate patterns, and to retrieve and describe texture primitives. Texture primitives may be extracted using a difference-of-Gaussian filter, for example [42]. These methods attempt to characterize the primitives and find rules governing their spatial organization.

- *Model-based methods* are used to compute a parametric generative model based on the intensity distribution of texture primitives. A widely used class of probabilistic models is: Conditional Random Fields (CRF) [43], Markov Random Fields (MRF) [44], Gaussian Markov Random Fields (GMRF) [45], fractals [46] and Local Binary Patterns (LBP) [47], *etc.*
- *Frequency methods* are used to investigate the overall frequency content of an analyzed image. The most widely used frequency methods are Gabor filters [38, 48], Fourier transform [49], wavelet transform [49, 50] and moment-based texture segmentation [51].

Some approaches are used to investigate the local properties of an analyzed image (*e.g.* GMRF, LBP, *etc.*). Other methods are based on statistical and/or spatial and/or frequency representations (*e.g.* wavelet transform, Gabor filters, *etc.*).

During the last two decades, several studies have sought to characterize and index ancient documents using their content to explore textural analysis [31, 32, 52, 53]. Some studies looked at the whole ancient document [31], while others examined graphic images such as drop caps [52]. Uttama *et al.* introduced a drop cap segmentation method [52]. The proposed method was based on a combination of different texture analysis approaches (*e.g.* GLCM [41] and the autocorrelation function [54, 55], *etc.*). Others developed texture-based feature extraction algorithms that were designed for the analysis of historical documents such as the ancient document image characterization approach proposed by Journet *et al.* [31]. The computed texture features were based on the autocorrelation function and frequencies. This method provided good information on the principal orientations and periodicities of the texture and could characterize the content of images without making assumptions about the image structure or its properties.

The use of the autocorrelation function is not new in the document analysis community. Numerous studies have identified a number of autocorrelation features for segmenting ancient and contemporary documents images [31, 32, 53, 56–59]. Autocorrelation results have been used to construct a rose of directions [60]. Eglin *et al.* determined the number of Gabor filters by selecting relevant directions, deduced from the rose of directions, in order to select interesting patterns for noise reduction and classification of handwritings in ancient manuscripts [56]. For old document analysis, Journet *et al.* defined three autocorrelation features which some descriptors were derived from the rose of directions. The extracted features computed over the neighborhood of each pixel (foreground and background), were as follows: the main orientation of the rose of directions, the intensity value of the autocorrelation function for the main orientation and the variance in the intensities of the rose of directions, except for the main orientation [31]. Grana *et al.* used the autocorrelation matrix to distinguish between textual and pictorial regions in historical manuscripts [58]. Garz and Sablatnig presented a multi-scale texture-based approach for text region recognition in ancient manuscripts [57]. They extracted the three autocorrelation features proposed by Journet *et al.* [31] by applying three scales by means of overlapping sliding windows. Then, they introduced shifted copies of the proposed textural features proposed by Journet *et al.* [31] such that the main orientation is at 0° to ensure the comparison of different roses of directions and the invariance to skewed text lines. Ouji *et al.* introduced two other texture attributes, also in relation to the autocorrelation function: the mean stroke width and height of an image for contemporary document image segmentation [59].

Texture feature extraction is designed to represent document content through a set of descriptive features that are computed or extracted. There have been a lot of texture analysis studies that use a variety of descriptors based on the statistical and spectral properties of texture. In principle, any texture-based method can be used to extract textural descriptors. Here, we extract texture features from the autocorrelation function for several reasons:

First, a comparative study on the choice of the texture feature category has been elaborated, which ensures the best, most constructive trade-off between best performance, reduced number of parameter settings and thresholds and lowest computation time [61,62]. For instance, the co-occurrence descriptors are statistics computed from the GLCM elements defined in a specified direction and separated by a particular distance. Texture feature methods based on GLCM feature analysis have been proposed in the literature to identify script and language from documents [63, 64]. One of the groups concluded that the GLCM features gave the worst overall performance for script identification [64]. The Gabor features are extracted using the multichannel 2-D Gabor filtering technique. A 2-D Gabor filter is a linear selective band-pass filter that is dependent on two parameters: spatial frequency and orientation. By extracting Gabor features, different studies have investigated to identify scripts, font-faces and font-styles [65, 66]. Jain *et al.* showed the effectiveness of using a multichannel Gabor filtering-based texture segmentation approach for the segmentation and classification of documents [67]. Despite the widespread application of Gabor filters to texture analysis, one of its most serious disadvantages is its high computational cost since it consists of convolving the whole document at each orientation and at each frequency [68].

Secondly, the extraction of the autocorrelation features requires less parameter settings compared to the descriptors computed from Gabor filters and GLCM. Indeed, with no hypothesis concerning the document layout or the typographical parameters of the document, the choice of appropriate thresholds and parameters is a very difficult task.

Finally, the high performance of segmentation experiments that are based on the autocorrelation function used on documents convinces us to work with autocorrelation features in order to reach our objective of determining homogeneous regions in an analyzed document with no hypothesis concerning the document layout or the typographical parameters of the document. The autocorrelation descriptors highlight interesting information on the principal orientations and periodicities of texture allowing characterizing the content of images without any assumption on the page structure and its characteristics. It has also been demonstrated that they work even for skewed images and handwritten text. In addition, they have been proved relevant and robust to noise, unconstrained document layouts and page skew [31, 32, 53, 56–59].

2.2 Feature space structuring methods

The partition task of the set of unlabeled data (obtained from the feature extraction phase) into groups or clusters is necessary to segment the analyzed image into regions which have homogeneous characteristics and similar properties with respect to the extracted features. This task is considered as a feature space structuring technique. Feature space structuring methods involve two phases:

- *The clustering phase* or unsupervised classification partitions a set of unlabeled data into homogeneous groups or clusters. Samples of each cluster share common characteristics, which usually correspond to proximity criteria, defined by introducing measures of distance between clusters and samples.
- *The classification phase* classifies a new object according to a set of predefined classes.

Clustering algorithms can be classified into two categories:

- *Hard clustering methods* distribute data into different clusters, where each data point belongs to exactly one cluster.
- *Fuzzy clustering methods* consider that the allocation of data points to clusters is not binary, *i.e.* each data point may belong to more than one cluster with a set of membership levels. One of the most widely used fuzzy clustering algorithms is the Fuzzy C-Means algorithm (FCM) [69].

In this work, we are interested in the hard clustering algorithms since many parameters must be specified in the case of the fuzzy clustering methods. Several standard hard clustering methods have been proposed in the literature. Hard clustering methods are divided into five categories [70]:

- *Partitioning methods* (*e.g.* k-means clustering (k-means) [71], Partitioning Around Medoids (PAM) [72], CLustering LARge Applications (CLARA) [72], *etc.*) distribute the dataset according to the proximities of feature space deducted from the content of the analyzed image.
- *Hierarchical methods* (*e.g.* AGglomerative NESting (AGNES) [72], DIvisive ANAlysis clustering (DIANA) [72], Hierarchical Ascendant Classification (HAC) [73], *etc.*) are widely used data analysis tools that produce a hierarchy of clusters based on a measure of similarity between groups of data points.
- *Density-based methods* (*e.g.* DBSCAN [74], OPTICS [75], EM [76], *etc.*) are designed to reveal clusters of arbitrary shapes based on the local densities of a point set after introducing the appropriate values of the input parameters (neighborhood radius, *etc.*).
- *Grid-based methods* (*e.g.* STING [77], WaveCluster [78], *etc.*) quantize the space into a finite number of cells without taking into consideration data density and distribution, and then perform clustering operations (neighborhood cells, *etc.*) on the quantized space.
- *Neural network-based methods* (*e.g.* Self-Organizing Maps (SOM) [79], Feed-Forward Network (FFN) [80], *etc.*) partition data into similar subsets with the aid of an artificial neural network [81].

Nevertheless, for a certain class of hard clustering algorithms and conventional clustering techniques [82–85], the number of clusters in a dataset must be specified. Several types of methods can be used to estimate the correct number of clusters. A link-based cluster ensemble framework was used to select the correct number of clusters after evaluating the clustering results of a variety of functional methods based on both internal and external criteria [86]. Ray and Turi determined the number of clusters for color image segmentation in a clustering algorithm by finding the minimum of the intra-cluster and inter-cluster distance [87]. An approach to determine the cluster boundaries

in the hierarchical clustering based on within-class variance and between-class variance has been reported [88]. Another technique has been proposed for the analysis of changes in silhouette values [89] computed from clusters built by using k-means [71] and an optimization technique such as genetic algorithms [90]. Moreover, v-fold cross-validation applied to clustering [91], was performed for a range of numbers of clusters in k-means or EM clustering, then, depending on the average distance of the observations (in the cross-validation or testing samples) from their cluster centers (for k-means clustering), the number of clusters was estimated. Otherwise, by varying all combinations of the number of clusters, distance measures and clustering methods, the changes in various clustering evaluation indices can be examined [92]. Kryszczuk and Hurley proposed a framework for cluster number estimation based on decision-level fusion of multiple clustering validity indices [93]. They proved that no single clustering validity indice consistently outperformed others, particularly for high dimensional datasets. Bolshakova and Azuaje proposed a weighed voting technique based on three clustering algorithms and two cluster validation indices to improve the prediction of the number of clusters [94].

To better represent images and improve the performance of Content-Based Image Retrieval (CBIR), Yu *et al.* suggested to use a classical clustering approach (*e.g.* the k-means algorithm) as a pre-processing step, for image clustering [95]. Recently, clustering of personal album images has been investigated as an instance of application of image clustering [96]. Wang *et al.* proposed a Web-based annotation method flowed by a graph-based semi-supervised learning approach to provide firstly the conceptual labels to personal album image clusters and then to distribute the obtained conceptual labels from image clusters to the whole photo album WNHC12-CSUR. Yu *et al.* presented an adaptive hypergraph learning approach for transductive image classification framework [95]. The proposed approach provided a simultaneous learning of the labels of unlabeled images and the weights of hyperedges based on images and their nearest neighbors. Other graph-based manifold learning proposals have been suggested to improve image clustering and classification such as the Sparse Patch Alignment Framework (SPAF) [97] and High-order Distance-based Multiview Stochastic Learning (HD-MSL) [98]. However, the graph-based manifold learning technique has provided good results in extracting features for image classification, this way of using of the unsupervised/semi-supervised graph-based methods is beyond the scope of our work. Indeed, in the context of our work, we opt for a non-supervised texture analysis approach for meeting the need to segment a book page layout in the conditions of noise presence or significant document image degradation and in the context of lacking information on the document structure such as the document layout and the typographical parameters.

2.3 A short review of texture-based approaches for historical books

A few texture-based segmentation approaches used with historical documents have been developed. To our knowledge, the only non-supervised texture-based approach used with historical books was proposed by Journet *et al.* It was based on an unsupervised clustering technique using extracted texture features which were computed from six pages of the same book. To assign the same label to pixels of six book pages which share similar textural characteristics, the clustering was performed on all extracted texture features of pixels of six book pages [31]. They extracted two different kinds of texture descriptors for each pixel: three autocorrelation features which were derived from the rose of directions and two frequency attributes by using a multi-scale analysis for classifying historical document image pixels into text, graphics and background. The first frequency descriptor computes the ink/paper transitions obtained by performing the average per-line sum of the difference between the pixel intensity value and its left neighbor. The frequency second attribute calculates the white spaces obtained by performing the XY-CUT algorithm and computing the mean of the average per-line and per-column sums of pixel intensities over an analyzed area. Then, by using the CLustering LARge Applications (CLARA) [72], an unsupervised clustering algorithm, the extracted texture descriptors were clustered and pixels were separated into different content clusters. Moreover, the number of homogeneous regions was assumed to be known in advance. They noted 83% and 92% mean good classification rates for the graphical and text pixels, respectively with 180 minutes in total per document as time required to process a page (feature extraction and pixel clustering tasks) [99].

However the Journet *et al.*'s texture-based approach yielded good results on ancient documents containing several textural classes (*e.g.* text, graphics and background), one main disadvantage is that there is a need of user intervention for setting the number of expected clusters, as a consequent of using a classical unsupervised clustering [31]. Then, the most serious disadvantages of their approach is its high computational cost caused by the texture feature extraction step which was processed on all page pixels (*i.e.* the foreground and background pixels). Finally, to assign the same label to pixels of six book pages which shares similar textural characteristics, the clustering approach was only performed on six pages of the same book instead of all book pages, which can lead assigning

different labels to each resulting cluster of two different sets of six pages of the same book (characterizing by similar textural properties).

Thus, in this article a texture-based pixel labeling framework is proposed for the segmentation and characterization of digitized historical book content which addresses the challenges of the existing state-of-the-art methods.

3 Proposed framework

For the segmentation and characterization of digitized historical book content, our goal is to determine a region or group of pixels which share similar properties or characteristics on the basis of which they are grouped. These characteristics may be based on the localization of the pixels and their surroundings, color, intensity or texture. In this article, we will focus only on texture-based features. The use of a texture-based approach in our work has been shown to be effective with skewed and degraded images [100]. We propose a framework which automatically extracts texture descriptors and involves a multiresolution/multiscale approach. This approach can segment and characterize the content of digitized historical book. In particular, it can discriminate between the different classes of the foreground layers of a digitized document based on texture descriptors. The extraction of texture-based features helps to describe the document layout and structure by analyzing the texture feature space computed from digitized historical book content, *i.e.* by mapping the differences in the spatial structures of digitized documents into differences in gray value for each page. The texture features are automatically extracted from the analyzed document at several resolutions. The extracted features are then used in a parameter-free unsupervised clustering approach to determine the homogeneous regions that are defined by similar textural descriptors. The proposed framework is pixel-based and does not require *a priori* knowledge of the document structure/layout or the typographical parameters of the document. Moreover, the number of homogeneous or similar content regions do not need to be known in advance as it is determined automatically. Thus, this framework is automatic, parameter-free and applicable to a large variety of historical books. It is independent of document layout, typeface, font size, orientation, digitizing resolution, *etc.* Moreover, it does not require any manual inspection.

The originality of this framework lies in the texture feature analysis that is used to find homogeneous regions by utilizing a clustering approach on an entire book instead of processing each page individually. The proposed framework is supported by the fact that pages of the same book usually present strong similarities in the organization of the document information (*i.e.* the book page layout or structure), and in the graphical (*e.g.* embellishment, engraving or pictures) and typographical (*e.g.* font size and type) features throughout the digitized book pages. Indeed, the texture information (*e.g.* the typographical or graphical properties) which is often repeated and recurrently present in many book pages, can be deduced by exploiting the regularities of the associated textures through the whole book pages. Thus, a clustering step is performed on texture features which are extracted from a subsampling in the entire book aims at identifying these book characteristics which are then used to help to segment each page image.

The proposed framework starts with a texture feature extraction step. Secondly, a number of foreground pixels from pages of the same book are selected randomly and their textural descriptors are subsequently extracted in order to estimate the number of homogeneous or similar content regions in the book. The estimated number of homogeneous regions in our samples of foreground pixels is determined automatically (block 2, Figure 3). Finally, the textural features for each page are used in a clustering approach by taking into account the estimation of the number of homogeneous or similar content regions (block 1, Figure 3).

Figure 3 illustrates the four main tasks of the proposed framework. Block 2 on Figure 3 is used to estimate the number of homogeneous regions from the extracted textural features analyzed in the whole book. Block 1 on Figure 3 integrates an unsupervised task which automatically labels content pixels with the same cluster identifier as used with the book content in order to determine and characterize the homogeneous regions in the digitized book (block 3, Figure 3).

Figure 4 illustrates the detailed schematic block representation of the proposed framework.

The proposed framework consists of the following three tasks:
1) Texture feature extraction (Section 3.1),
2) Estimation of the number of homogeneous regions (Section 3.2),
3) Pixel clustering and labeling (Section 3.3).

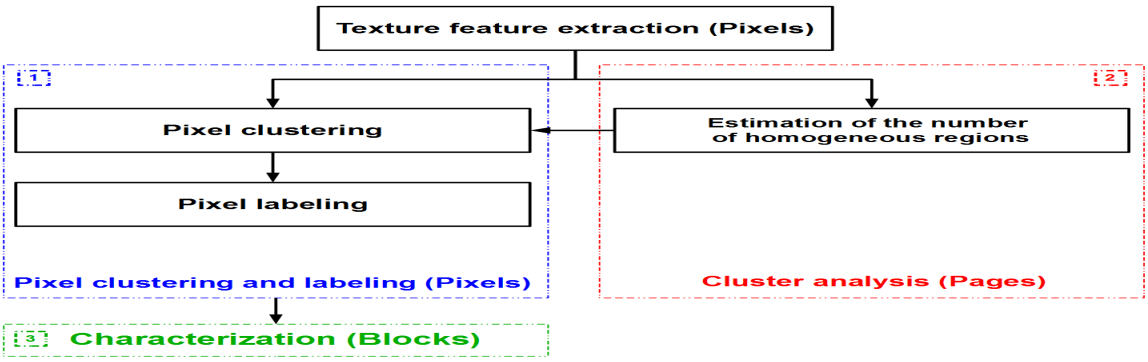


Fig. 3: Flowchart of the proposed pixel labeling framework of digitized historical book content.

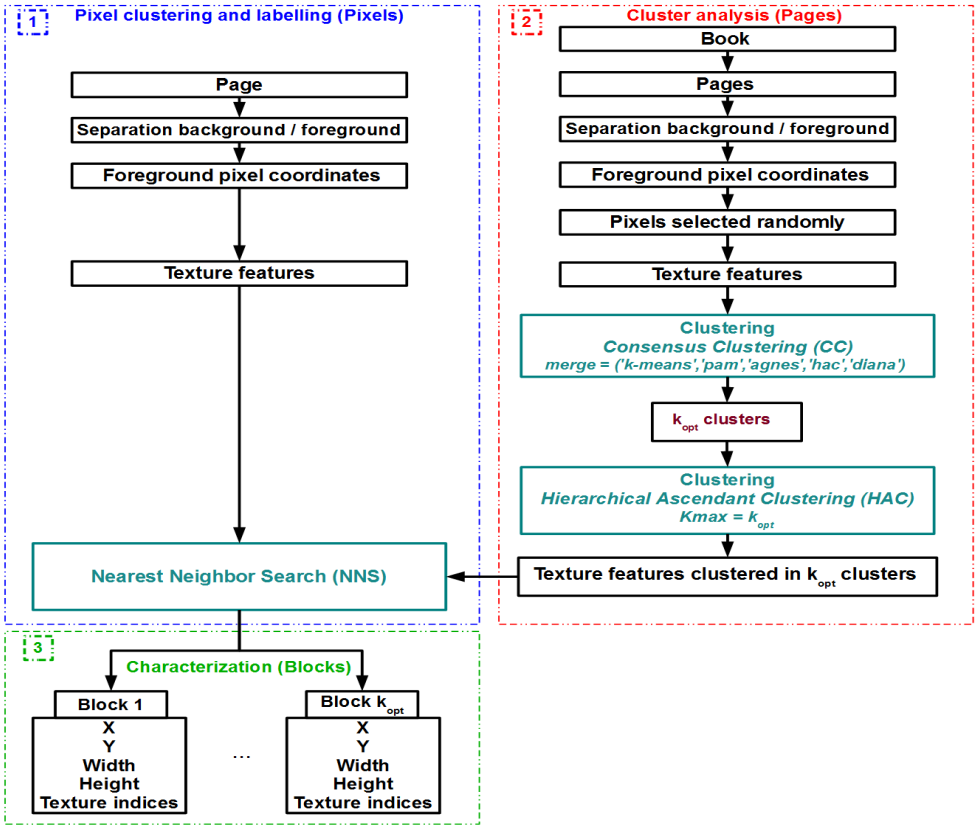


Fig. 4: Detailed schematic block representation of the proposed pixel labeling framework of digitized historical book content.

3.1 Texture feature extraction

The first stage of the framework is to compute the texture features (Figure 3). Due to the drawbacks of approaches that are based on strong *a priori* knowledge, cited in Section 2, approaches based on texture feature extraction and analysis are more suitable for documents with a complex layout. In addition, it has been demonstrated that texture-based approaches work effectively with no hypothesis concerning the document layout (physical structure) or the typographical parameters (logical structure) of document structure [31, 32]. The aim of the feature extraction methods is to extract a set of features from the analyzed image to represent and characterize its content.

3.1.1 Foreground pixel selection

In order to reduce data cardinality and obtain a significant gain in computation time and memory, the autocorrelation descriptors are computed only on the selected foreground pixels. As an example, for a full historical page document (1965×2750 pixels), scanned at 300 dpi, the number of the selected foreground pixels is equal to 26086. Thus, the rate of the selected foreground pixels is over $\frac{1}{200}$ of a document image pixels. In this work, our goal is to have an overview of the page content by finding regions with similar textural content as easily, quickly and automatically as possible rather than a fine characterization. Additionally, the texture of the foreground is more interesting to categorize the type of document content.

The foreground pixel selection step is performed using a standard parameter-free binarization method, the Otsu method, to retrieve only those pixels representing information of the foreground (noise, text fields, graphics, *etc.*) [101]. However, using of the Otsu method is beyond the scope of our work, it has provided good results [64]. They used the Otsu method to segment and extract text regions from a document. Shijian and Tan binarized document images using the Otsu global thresholding method to retrieve character pixels and subsequently identify scripts and languages of noisy and degraded documents [102]. Several comparative studies of segmentation text/background or binarization methods for degraded historical documents have been reviewed [103, 104]. These studies do not agree on the best method and none has been shown to be perfect and suitable for historical documents, even local binarization approaches. Using a global thresholding approach, the Otsu method provides an adequate and fast means of binarization to retrieve only foreground pixels and extract texture features from only the selected foreground pixels.

3.1.2 Multiresolution analysis

The texture feature extraction is performed using analysis windows of varying sizes in order to adopt a multiresolution/multiscale approach. The analysis window can be [37, 105]:

- *Pixel-wise*: each pixel is assigned to an analysis window, which ensures overlapping blocks or regions.
- *Block-wise*: the analyzed document is partitioned into non-overlapping blocks.

The pixel-wise technique is chosen since it gives more reliable values and ensures more accurate determination of texture boundary, however it has a high demand in memory and computational time. Using a multiresolution approach in document image analysis [106–109] and pyramid methods in image processing [110, 111], rich information (*e.g.* for gray level distribution) can be produced since we can perceive differently textural characteristics at varying scales.

A multiresolution analysis was proposed by selecting concentric windows with different sizes in order to characterize the images drawn from historical documents using texture analysis [112]. Another example of the use of multiresolution analysis with ancient documents was proposed by Journet *et al.* who computed their textural features by varying the window sizes [31]. Mehri *et al.* extracted texture feature from the selected foreground pixels of the gray-level document images at four different sizes of sliding windows: (16×16), (32×32), (64×64) and (128×128) to adopt a multiscale approach [61]. In this article, the extraction of autocorrelation descriptors per block is performed at four different sizes of sliding windows: (16×16), (32×32), (64×64) and (128×128).

3.1.3 Autocorrelation descriptors

The autocorrelation function is particularly used with synthetic textural images. It is considered a similarity measure between a dataset and a shifted copy of the data and is used to find periodic patterns and characterize pattern similarity [54, 55]. In a digitized document, the textual and non-text regions (blank spaces, graphics and noise, *etc.*) have different textured areas [29]. The autocorrelation function $R^{I(\alpha,\beta)}_{(x,y)}$ is computed along the horizontal and vertical axes of the analysis window I of an image according to the following equation:

$$\begin{aligned} R^{I(\alpha,\beta)}_{(x,y)} &= \sum_{\alpha\in\Omega}\sum_{\beta\in\Omega} I(x,y)I(x+\alpha,y+\beta) \\ &= FFT^{-1}\left[FFT\left[I(x,y)\right]FFT^{*}\left[I(x,y)\right]\right] \end{aligned} \tag{1}$$

where $I(x+\alpha,y+\beta)$ is the translation of the analysis window of an image $I(x,y)$ by α and β pixels along the horizontal and vertical axes, respectively, defined on the plane Ω. FFT, $(.)^{*}$, and $(.)^{-1}$ denote the Fast Fourier Transform, complex conjugate, and inverse transform, respectively.

The rose of directions, which is a derivative of the autocorrelation function, is deduced from the autocorrelation function [60]. It is a polar diagram derived from the analysis of the autocorrelation results and reveals the significant orientations of the texture in the analyzed image block. It highlights interesting information concerning the principal orientations and periodicities of the texture, characterizing the content of images without any assumption about page structure and its characteristics. The rose of directions has recently been used with historical documents [31,32,53,56,57]. In order to identify the main orientation of the analyzed image, the rose of directions is computed for each orientation by summing up the different values of the autocorrelation function (equation (1)):

$$R^{I}_{(x,y)}(\Theta_i) = \sum_{D_i} R^{I(\alpha,\beta)}_{(x,y)} \tag{2}$$

where $\Theta_i \in [0,180]$ is the selected orientation of the set of possible orientations D_i, which is represented by a straight line passing through (x,y) and the angle Θ_i. The rose of directions is normalized in one of the above studies in order to select only the relative variations of all contributions for each direction [31]. The relative sum $R'^{I}_{(x,y)}(\Theta_i)$ is defined as:

$$R'^{I}_{(x,y)}(\Theta_i) = \frac{R^{I}_{(x,y)}(\Theta_i) - R^{I}_{min}}{R^{I}_{max} - R^{I}_{min}} \tag{3}$$

where $R^{I}_{max} \neq R^{I}_{min}$, R^{I}_{min} and R^{I}_{max} represent the minimum and maximum values of $R^{I}_{(x,y)}(\Theta_i)$, respectively, both of which are computed on the analysis window of an image $I(x,y)$.

To illustrate the performance of the rose of directions in discriminating between textual and graphical regions in the document, and to determine the main orientation of a texture, Figure 5 shows the rose of directions obtained with four different textures. As can be seen, the shape of the rose is different for each type of texture. For textual regions such as (c), the shape of the rose depends on the orientation of the text and the main information. The horizontal orientation (0° and 180°) is clearly identifiable in (g). For drawing (d), the rose of directions (h) is deformed.

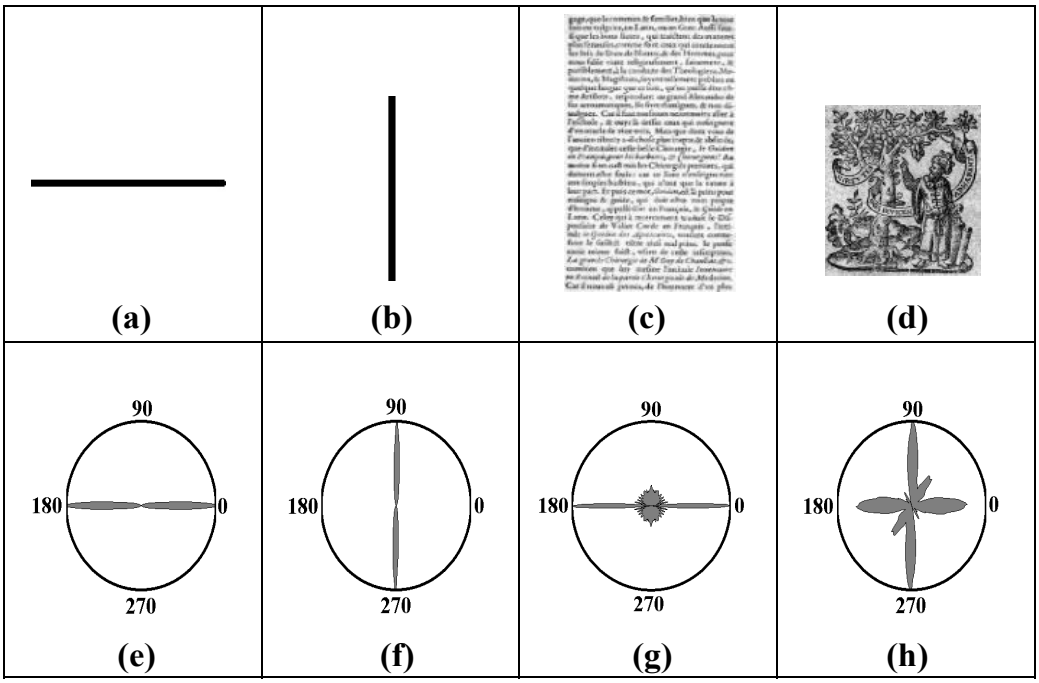


Fig. 5: Examples of the rose of directions. {**(a)**,**(b)**,**(c)**,**(d)**} are the original images and {**(e)**,**(f)**,**(g)**,**(h)**} their respective roses of directions.

The various forms and shapes of the rose of directions which are obtained from the variety of textures contained in ancient grayscale documents do not help us to define a template of the rose of directions for each type of texture. Nevertheless, computing the rose helps us to extract significant and relevant indices for texture features. Journet *et*

al. defined three texture features related to orientation in order to analyze the digitized document and to describe its content [31].

– *Main angle of the rose of directions*

The first texture feature $F^{(1)}_{(x,y)}$ corresponds to the main angle of the rose of directions extracted from its maximal intensity (Figure 6(a)). It is normalized by the deviation from the horizontal angle in order to avoid handling circular data. It is given by:

$$F^{(1)}_{(x,y)} = \left\| 180 - \underset{\Theta_i \in [0,180]}{\operatorname{argmax}} (R'^{I}_{(x,y)}(\Theta_i)) \right\| \tag{4}$$

– *Intensity of the autocorrelation function for the main orientation*

The second texture feature $F^{(2)}_{(x,y)}$ corresponds to the intensity of the autocorrelation function for the main orientation (equation (4)), which is computed on the non-normalized value of the autocorrelation function (equation (2)). This feature evaluates the anisotropy of an image $I(x,y)$ since the rose of directions associates the gray level of pixels in a specific direction. It is computed as:

$$F^{(2)}_{(x,y)} = R^{I}_{(x,y)}(\underset{\Theta_i \in [0,180]}{\operatorname{argmax}} (R'^{I}_{(x,y)}(\Theta_i))) \tag{5}$$

– *Variance of the intensities of the rose of directions*

The third texture index $F^{(3)}_{(x,y)}$ characterizes the overall shape of the rose. $F^{(3)}_{(x,y)}$ is the variance of rose intensities, except for the orientation of maximal intensity. A low $F^{(3)}_{(x,y)}$ means that the main orientation is significantly more prevalent than the other orientations. However, a high variance signifies that the rose is deformed and that there are a large number of orientations that are present to different extents (graphic blocks) (Figure 6(b)). Hence, the third texture descriptor is defined by:

$$F^{(3)}_{(x,y)} = \sigma^2(R'^{I}_{(x,y)}(\Theta_i)) \tag{6}$$

where $\Theta_i \in [0,180] \setminus \{\operatorname{argmax}_{\Theta_i \in [0,180]}(R'^{I}_{(x,y)}(\Theta_i))\}$ and σ represents the standard deviation estimator. The standard deviation estimator σ is computed as:

$$\sigma^2 = \frac{1}{n-1}\sum_{i=1}^{n}(R'^{I}_{(x,y)}(\Theta_i))^2 - \frac{n}{n-1}(\mu)^2 \tag{7}$$

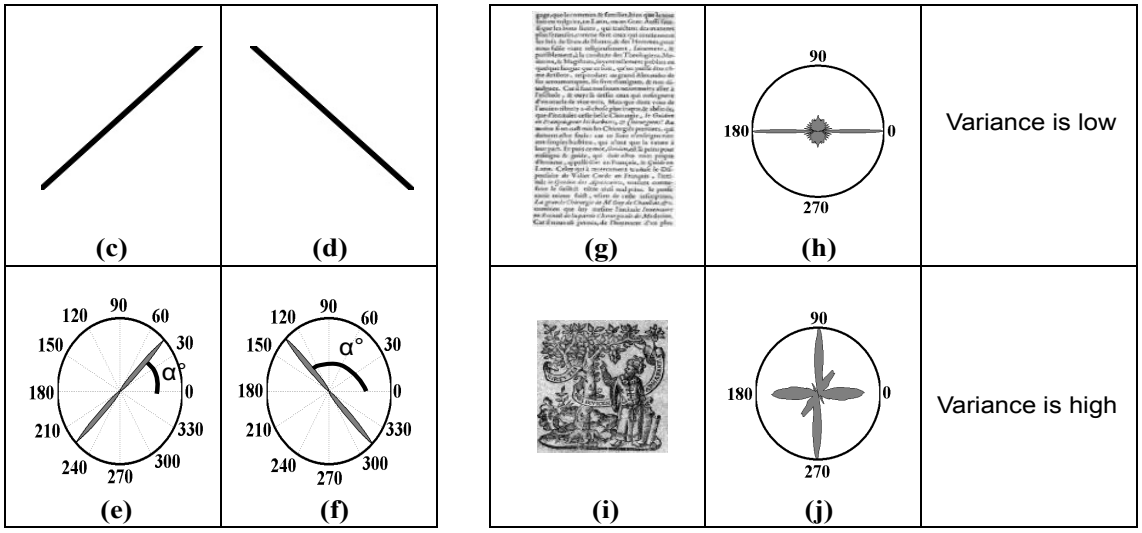
where μ and n are the mean value and the 179 orientation values, respectively.

In addition to the three texture features that are associated with the orientation of the autocorrelation function, we compute two other texture attributes which were first introduced by Ouji *et al.* [59] and seem to be relevant for contemporary documents and specifically with typographic characteristic characterization and chromatic/achromatic decomposition. The two texture descriptors are also related to the autocorrelation function through the mean stroke width and height of an image [59]. Ouji *et al.* computed these features in the horizontal and vertical directions [59]. Mehri *et al.* computed the mean stroke width and height along the axis of the main angle of the rose of directions to accurately estimate the main stroke thickness along specific directions [53]. In this article, the estimation of mean stroke width and height along specific directions is processed according to the work presented by Mehri *et al.* [53].

– *Mean stroke width along specific directions*

The next texture index corresponds to the estimation of mean stroke width along specific directions $F^{(4)}_{(x,y)}$. It is deduced from a derivative of the autocorrelation function along the axis of the main angle of the rose of directions Θ (equation (4)) if $\Theta \in [10,80]$ (equation (8)), otherwise the mean stroke width is estimated along the horizontal axis (equation (9)). If the growth rate of the sequence S^{width} (equations (8) and (9)) is lower than 10%, we estimate the mean stroke width, otherwise we continue to compute the sequence S^{width} until we reach the horizontal borders of the sliding window. S^{width} is defined to be:

$$S^{width} = \sum_{\Theta \in [10,80]} |I(x,y) - T^{\Theta}_{(\alpha,0)}(I(\frac{y}{|\tan(\Theta)|}, y))| \tag{8}$$



(a) Main angle of the rose of directions.

(b) Variance of the intensities of the rose of directions.

Fig. 6: Examples of main angle and variance of intensities of the rose of directions. {**(c)**,**(d)**} are the original images and {**(e)**,**(f)**} are their rose of directions, respectively. The main orientation on the rose of directions corresponds to the direction of the information contained in the analyzed image. {**(g)**,**(i)**} are the original images and {**(h)**,**(j)**} are their rose of directions, respectively. The variance of intensities for the roses is high for graphic regions and low for text regions.

$$S^{width} = \sum_{\Theta \in [0,9] \cup [81,180]} |I(x,y) - T^{\Theta}_{(\alpha,0)}(I(x,y))| \tag{9}$$

where $T^{\Theta}_{(\alpha,0)}(I(.,.))$ is the translation of the analysis window of an image I by α pixels along the axis of the main angle of the rose of directions $\Theta = F^{(1)}_{(x,y)}$.

The estimation of mean stroke width along specific directions $F^{(4)}_{(x,y)}$ is defined according to the algorithm 1.

– *Mean stroke height along specific directions*

The computation of the last texture attribute is similar to that of the fourth texture index $F^{(4)}_{(x,y)}$. $F^{(5)}_{(x,y)}$ is an estimation of the mean stroke height computed along the axis of the main angle of the rose of directions Θ (equation (4)) if $\Theta \in [10,80]$ (equation (10)), otherwise the mean stroke height is estimated along the vertical axis (equation (11)). If the growth rate of the sequence S^{height} defined in equations (10) and (11) is lower than 10%, the mean stroke height is estimated, otherwise we continue to compute the sequence S^{height} until we reach the vertical borders of the analyzed sliding window. S^{height} is defined to be:

$$S^{height} = \sum_{\Theta \in [10,80]} |I(x,y) - T^{\Theta}_{(0,\beta)}(I(x, x * |\tan(\Theta)|))| \tag{10}$$

$$S^{height} = \sum_{\Theta \in [0,9] \cup [81,180]} |I(x,y) - T^{\Theta}_{(0,\beta)}(I(x,y))| \tag{11}$$

where $T^{\Theta}_{(0,\beta)}(I(.,.))$ is the translation of the analysis window of an image I by β pixels along the axis of the main angle of the rose of directions $\Theta = F^{(1)}_{(x,y)}$.

The estimation of mean stroke height along specific directions $F^{(5)}_{(x,y)}$ is defined according to the algorithm 2.

Figure 7 illustrates the mean stroke width and height differences of two fonts (normal and bold text characters) along the axis of the main angle of the rose of directions.

Algorithm 1 Estimation of mean stroke width along specific directions

$pacc \leftarrow 0$
if $10 \leq \Theta \leq 80$ **then**
  $\boldsymbol{strokeWidth} \leftarrow 1$
  while $\boldsymbol{strokeWidth} < imageWidth$ **do**
    $acc \leftarrow 0$
    $y \leftarrow 0$
    while $y < imageHeight$ **do**
      $tacc \leftarrow 0$
      $x \leftarrow 0$
      $tx \leftarrow \left\lceil \frac{y}{|\tan(\Theta)|} \right\rceil - \boldsymbol{strokeWidth}$
      while $x < imageWidth$ **do**
        $tacc \leftarrow tacc + |I(x,y) - I(tx,y)|$
        $x \leftarrow x + 1$
      $acc \leftarrow acc + tacc$
      $y \leftarrow y + 1$
    if $pacc \neq 0$ **then**
      $seqWidth \leftarrow \frac{acc - pacc}{pacc}$
      if $seqWidth \leq 0.1$ **then**
        return $\boldsymbol{strokeWidth}$
    $pacc \leftarrow acc$
    $\boldsymbol{strokeWidth} \leftarrow \boldsymbol{strokeWidth} + 1$
  return $\boldsymbol{strokeWidth}$
else
  $\boldsymbol{strokeWidth} \leftarrow 1$
  while $\boldsymbol{strokeWidth} < imageWidth$ **do**
    $acc \leftarrow 0$
    $y \leftarrow 0$
    while $y < imageHeight$ **do**
      $tacc \leftarrow 0$
      $x \leftarrow 0$
      while $x < imageWidth$ **do**
        $tx \leftarrow x - \boldsymbol{strokeWidth}$
        $tacc \leftarrow tacc + |I(x,y) - I(tx,y)|$
        $x \leftarrow x + 1$
      $acc \leftarrow acc + tacc$
      $y \leftarrow y + 1$
    if $pacc \neq 0$ **then**
      $seqWidth \leftarrow \frac{acc - pacc}{pacc}$
      if $seqWidth \leq 0.1$ **then**
        return $\boldsymbol{strokeWidth}$
    $pacc \leftarrow acc$
    $\boldsymbol{strokeWidth} \leftarrow \boldsymbol{strokeWidth} + 1$
  return $\boldsymbol{strokeWidth}$

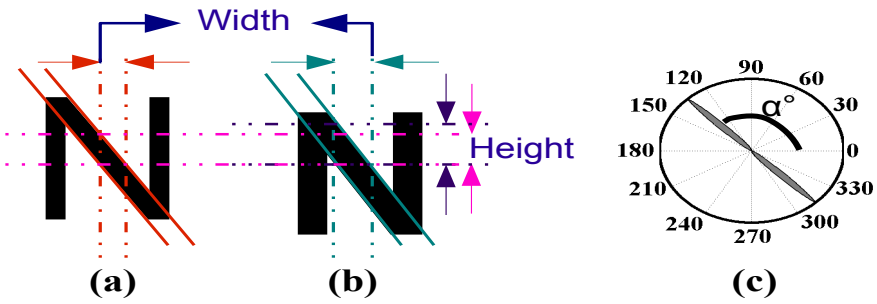


Fig. 7: Estimation of the mean stroke width and height along specific directions. {**(a)**,**(b)**} are the original images and **(c)** their rose of directions. **(a)** depicts a normal text character while **(b)** illustrates a bold text character. As the main orientation of the rose of directions is oblique **(c)**, the mean stroke width and height are estimated along the oblique axis.

Algorithm 2 Estimation of mean stroke height along specific directions

```
pacc ← 0
if 10 ≤ Θ ≤ 80 then
    strokeHeight ← 1
    while strokeHeight < imageHeight do
        acc ← 0
        y ← 0
        while y < imageHeight do
            tacc ← 0
            x ← 0
            while x < imageWidth do
                ty ← ⌈x ∗ |tan(Θ)|⌉ − strokeHeight
                tacc ← tacc + |I(x,y) − I(x,ty)|
                x ← x + 1
            acc ← acc + tacc
            y ← y + 1
        if pacc ≠ 0 then
            seqHeight ← (acc − pacc) / pacc
            if seqHeight ≤ 0.1 then
                return strokeHeight
        pacc ← acc
        strokeHeight ← strokeHeight + 1
    return strokeHeight
else
    strokeHeight ← 1
    while strokeHeight < imageHeight do
        acc ← 0
        y ← 0
        while y < imageHeight do
            tacc ← 0
            x ← 0
            ty ← y − strokeHeight
            while x < imageWidth do
                tacc ← tacc + |I(x,y) − I(x,ty)|
                x ← x + 1
            acc ← acc + tacc
            y ← y + 1
        if pacc ≠ 0 then
            seqHeight ← (acc − pacc) / pacc
            if seqHeight ≤ 0.1 then
                return strokeHeight
        pacc ← acc
        strokeHeight ← strokeHeight + 1
    return strokeHeight
```

3.1.4 Feature extraction process

Extracting these autocorrelation indices using a sliding window gives a total of 20 features, *i.e.* 20 numerical values (5 texture indices $\times$ 4 sliding window sizes) which are computed for each selected foreground pixel from the digitized document. The extraction of the textural descriptors is performed on each grayscale document image. Figure 8 depicts the different pre-processing steps of the proposed step of texture feature extraction, based on the autocorrelation function and multiresolution analysis. The autocorrelation features are computed for analysis windows of different sizes in order to adopt a multiscale approach. The sliding window is shifted horizontally and vertically to scan the entire image. To deal with pixels at image borders when computing texture features on the whole image, a border replication step is used.

Figure 9 illustrates an example of four different sizes of sliding windows: (16×16), (32×32), (64×64) and (128×128), and shows that each window provides additional information on the textural properties.

The optimal size of each sliding window, respecting a constructive compromise between computation time and pixel labeling quality (reliable measurement and texture boundary), is determined experimentally. Computation time is highly dependent on the resolution, the size of the analyzed document and the number of foreground pixels retrieved. As an example, for a full historical page document (1965×2750 pixels), scanned at 300 dpi, it took

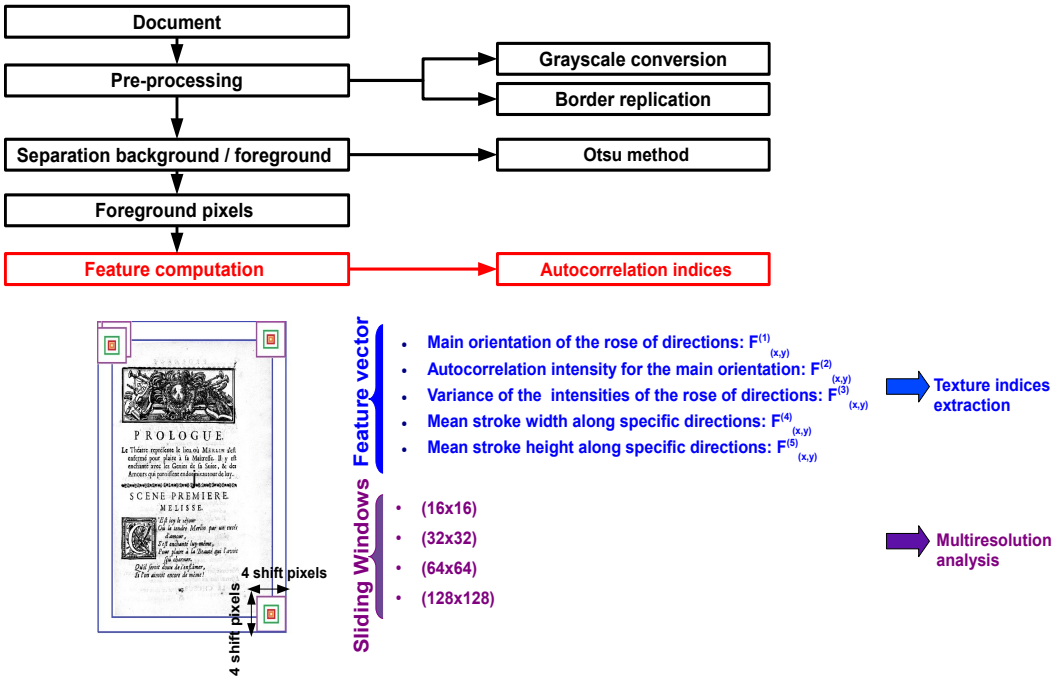


Fig. 8: Texture feature extraction.

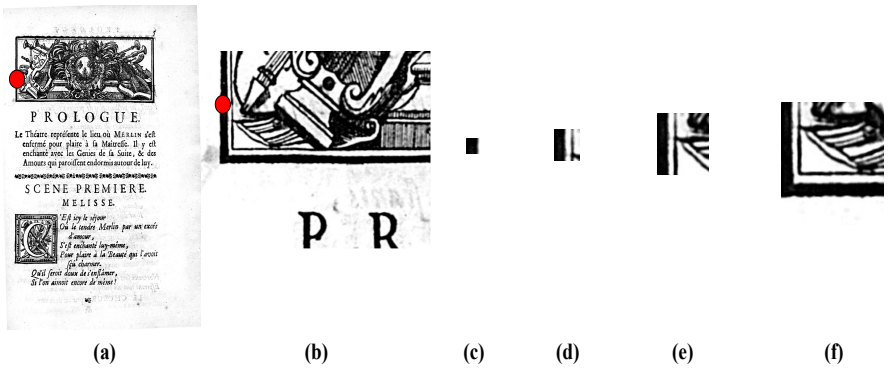


Fig. 9: Example of four different sizes of sliding windows: **(a)** original image with a selected pixel position, **(b)** image zoom, **(c)** (16×16) window, **(d)** (32×32) window, **(e)** (64×64) window and **(f)** (128×128) window.

about 2 minutes to process the feature extraction. The experiment is run on a SGI Altix ICE 8200 cluster (1 CPU and 2 gigabytes allocated memory on a Quad-Core $X5355@2.66GHz$ running Linux).

3.2 Estimation of the number of homogeneous regions

As already seen on the framework figure (Figure 3), our objective is to find homogeneous regions defined by similar texture features. So at this stage (block 2, Figure 3) we need to use a clustering algorithm to partition the analyzed document into regions with similar properties or characteristics as deduced from the analysis of the extracted texture features presented in Section 3.1.

Previous work identified a number of approaches for determining the correct number of clusters in a dataset [113]. Simpson *et al.* have recently proposed an effective method, known as Consensus Clustering (CC), to estimate the optimal number of clusters in biological data [114]. With the help of the CC technique, Mehri *et al.* estimated the number of clusters from a number of samples of foreground pixels to determine the number of homogeneous regions defined by similar autocorrelation indices in an ancient book [32]. Thus, we use the CC in this work to estimate the number of homogenous or similar content regions.

The CC consists of performing a consensus matrix by iterating multiple runs of clustering algorithms with random and re-sampled clustering options [115]. Thus, the consensus matrix analyzes the consistency of the clustering results from five different clustering algorithms: AGglomerative NESting (AGNES) [72], DIvisive ANAlysis clustering (DIANA) [72], Partitioning Around Medoids (PAM) [72], k-means clustering (k-means) [71] and Hierarchical Ascendant Classification (HAC) [73]. So, by weighting the different clustering methods in order to mitigate extremes in consensus values that could result from the sensitivity of some algorithms, a merge consensus matrix is performed which ensures the stability of the obtained clusters. Finally, the optimal number of clusters corresponds to the largest change in area under the cumulative density curve for the merge consensus matrix. It has been shown that hierarchical clustering methods are highly sensitive to outliers while partitioning methods are relatively insensitive. Simpson *et al.* therefore used a merged consensus clustering by applying a weighted averaging of the clustering results to estimate the number of clusters [114].

Thus, the number of clusters in a set of randomly selected foreground pixels is estimated from a few randomly selected pages of a book using the CC method. This method is only used for a set of randomly selected pixels of a few pages selected randomly from the same book. Due to memory constraints and long computational time of the CC method, we first test it on a set of 1000 and 2000 randomly selected pixels from 10 pages selected randomly from the same book.

Variations in clustering for both hierarchical clustering and partitioning methods can be taken into consideration by associating non-uniform weights. With this approach, prior information is introduced into the clustering process by assigning higher weights to the most robust clustering methods. Thus, by weighting different clustering methods, extremes are mitigated in consensus values that can be created by the sensitivity of some algorithms, meaning that outliers can be dealt with differently within datasets, thus improving the quality of classification. So a weight of $\frac{1}{8}$ is assigned to each hierarchical clustering method (AGNES, DIANA and HAC), and a higher weight of $\frac{1}{4}$ is assigned to each partitioning clustering algorithm (PAM and k-means) [114]. By using this merge consensus clustering technique, the consensus matrices are the results deduced from clustering experiments using different algorithms and/or conditions. The merging of clustering results between different methods provides an averaged clustering robustness, *i.e.* a merge consensus matrix M. Hence, the optimal number of clusters k_{opt} in a dataset can be estimated by finding the value of k computed from the merge consensus matrix M across a range $[2,10]$ of possible values of k. The Cumulative Density Function ($CDF(c)$) is computed on the unique elements of the merge consensus matrix M sorted in descending order and defined over the range $c = [0,1]$. Thus, the $CDF(c)$ is defined using equation (12).

$$CDF(c) = \frac{\sum_{i<j} \mathbb{1}_{M(i,j)\leq c}}{\frac{N_s(N_s-1)}{2}} \tag{12}$$

where N_s is the number of selected observations or samples and $\mathbb{1}$ is an indicator or a characteristic function defined on a set $M(i,j) \leq c$.

The Area Under the Cumulative density curve (AUC) is then computed from the CDF (equation (12)) of the consensus matrix across a range $[2,10]$ of possible values of k using equation (13).

$$AUC = \sum_{i=2}^{m} [x_i - x_{i-1}]\, CDF(x_i) \tag{13}$$

where x_i is the current element of the CDF and m is the number of elements [114].

Finally, the optimal number of clusters k_{opt} corresponds to the largest change Δk in the AUC (equation (13)).

3.3 Pixel clustering and labeling

Since the feature extraction phase and the estimation of the optimal number of homogeneous k_{opt} task have been performed, we need to characterize the content of an entire book and find the k_{opt} homogeneous regions defined by similar texture indices in a whole book. The goal of the third task of the proposed framework (block 1, Figure 3) is to structure the texture feature space within a hierarchical or partitioning clustering technique in order to group pixels sharing similar characteristics to identify and characterize similar regions or groups of pixels.

3.3.1 Pixel clustering

In this work, we opt for a standard and reliable hard clustering algorithm, given its optimal trade-off between low complexity, accuracy of the results, reduced number of parameter settings and the requirement for a clustering

technique. This stage (block 1, Figure 3) consists of automatically grouping the pixels into k_{opt} clusters representing homogeneous or similar texture-content regions. Since the main purpose of the CC is to compare, visualize and evaluate the repeatability of the results of clustering experiments, and given the high demand in terms of memory and computational time of the CC algorithm, we perform the HAC algorithm on the computed texture features without taking into account the spatial coordinates to search and extract homogeneous regions for each digitized book page.

As part of an attempt to provide wider access to historical collections, Nguyen *et al.* focused their study on specific graphics called drop caps, and on the extraction of shapes in these graphics [116]. They found interesting classification results which were obtained by performing the HAC algorithm on the stroke features of drop caps. Furthermore, Lai *et al.* stated that the distance computed by the Ward method [117] gave the best results with the HAC method [34].

The HAC algorithm process consists of successively merging pairs of existing clusters where at each cluster grouping step, the choice of cluster pairs depends on the smallest distance, *i.e.* clusters are grouped if the intra-cluster inertia is minimal. This linkage between clusters is performed using the Ward criterion along with the weighted Euclidean distance ($WED(a,b)$) [118]. The WED is defined as:

$$WED(a,b) = \sqrt{n_a n_b \frac{\sum_{k=1}^{N^f} \frac{1}{N^f} \|\overline{x_{ak}} - \overline{x_{bk}}\|}{n_a + n_b}} \tag{14}$$

where $\overline{x_{ak}} = \frac{\sum_{i=1}^{n_a} x_{ai}}{n_a}$ is the centroid of cluster a (resp. b) and n_a (resp. n_b) is the number of elements in cluster a (resp. b). N^f is the number of vector features. The greater the WED (equation (14)) between two clusters, corresponding to two different kinds of texture, the better the discrimination of textural characteristics. The texture feature vectors computed at the selected foreground pixels are not identical and generate k_{opt} clusters in the multidimensional feature space.

The texture feature vectors are normalized to zero mean and unit standard deviation in order to avoid a domination of the higher numerical range of a few features. By setting the maximum number of homogeneous regions to the k_{opt} estimated with the CC method, the adapted HAC algorithm with the Ward criterion can be applied to the normalized textural features of the randomly selected samples of a book. This task is essential for finding the k_{opt} homogeneous regions defined by similar texture indices in the whole book. Finally, we obtain k_{opt} clusters for randomly selected foreground samples of a book, *i.e.* k_{opt} clusters of selected autocorrelation vectors computed from a few pages of a book, representing k_{opt} similar content regions.

3.3.2 Pixel labeling

This phase deals with labeling clusters or groups of pixels with respect to the results of the pixel clustering phase. The idea of this task (block 1, Figure 3) is to assign a label to each cluster of pixels which shares similar textural characteristics to the cluster obtained from the selected foreground samples of the book (block 2, Figure 3).

Journet *et al.* performed the clustering stage using CLARA, which is suitable for large scale databases, in the extracted texture features computed from six pages of the same book [31]. Then, if two pixels of two different documents have the same cluster label, they belonged to the same class. However, this technique is characterized by a long processing time and memory complexity. A clustering approach without taking into account the characterization step was proposed by Mehri *et al.* that is designed to determine and assign the same cluster identifier to each similar cluster extracted from the digitized book [53].

In this work, an unsupervised task is integrated that automatically labels content pixels with the same cluster identifier as the book content. For the same book, each cluster (represented by a given color) represents a similar or homogeneous region. Thus, by applying the HAC algorithm, we extract the homogeneous regions defined by similar texture indices, and then we perform the Nearest Neighbor Search algorithm (NNS) [119] in order to assign the same label to each similar cluster extracted from the digitized book. The NNS is used between each texture feature vector with each digitized page of the same book and the k_{opt} clusters of the selected samples of a book in order to find the texture feature vector closest to the cluster of the selected foreground samples of a book, *i.e.* by selecting the minimum distance.

The NNS is used with the Mahalanobis distance to assign the same label for each similar cluster extracted from a digitized book [120]. The Mahalanobis distance (MD) can be defined as a measure of dissimilarity between two vectors. The MD takes into account dataset correlations and is particularly suited to arbitrarily shaped clusters. The MD is computed of each texture feature vector for each digitized page of the same book from the reference

sample defined by each cluster of the selected foreground samples of a book. The Mahalanobis distance $MD(x,y)$ of two multivariate vectors $x = (x_1, x_2, \ldots, x_{N^f})^T$ and $y = (y_1, y_2, \ldots, y_{N^f})^T$ of the same distribution N^f with the covariance matrix S, is defined as:

$$MD(x,y) = \sqrt{\|(x-y)^T S^{-1}(x-y)\|} \tag{15}$$

Since the clustering and labeling phases of the proposed framework have been performed, the homogeneous regions in the digitized book are determined and characterized (block 3, Figure 3).

4 Experiments

To evaluate the performance of our proposed framework and validate our choice of the used techniques on each step of our framework, a set of experiments on a large variety of ancient books and historical document images is detailed in this section. First, our experimental corpus is presented. Secondly, an overview of several internal and external clustering evaluation metrics and different measures of classification accuracy is described. Finally, an assessment of the different steps of our framework is presented and an analysis of the obtained results is subsequently detailed.

4.1 Experimental protocol

As part of a collaboration with the BnF (Bibliothèque nationale de France)[22], a French National Library, as part of the DIGIDOC project, we deal with ancient books collected from the Gallica digital library. The characteristics of our corpus of historical documents are primarily: strong heterogeneity, with differences in layout, typography, illustration style and complex structures. In addition to this specificity, there are degradation properties (yellow pages, ink stains, back-to-front interference, *etc.*) and scanning defects (defects of curvature, light, *etc.*) which complicate the characterization or segmentation of ancient documents and make the processing of this kind of document a difficult task (Figure 2). Two datasets are used in our experiments: *"1st dataset"* and *"2nd dataset"*. *"1st dataset"* is used to assess the robustness of the extracted textural features and *"2nd dataset"* is used to evaluate the results of the different steps of our framework.

The initial evaluation of the framework involves selecting a set of simplified ancient documents (*"1st dataset"*) to assess the performance of the extracted autocorrelation features. By adding some superpositions, *i.e.* white rectangular regions on several component parts or elements of the document content, the document layout is simplified and its complexity is reduced in order to control the analyzed content. Hence, when the document layout is simplified, the corpus distribution can be defined objectively (*i.e.* after simplification of document layout and content, the remaining contents of document images consists of graphics and text with up to three different fonts or only text regions with up to three different fonts). A subset of historical documents containing graphics and text with up to three different fonts is selected from our corpus (Figure 10). From these selected historical documents, a set of 314 simplified historical documents is produced (*"1st dataset"*):

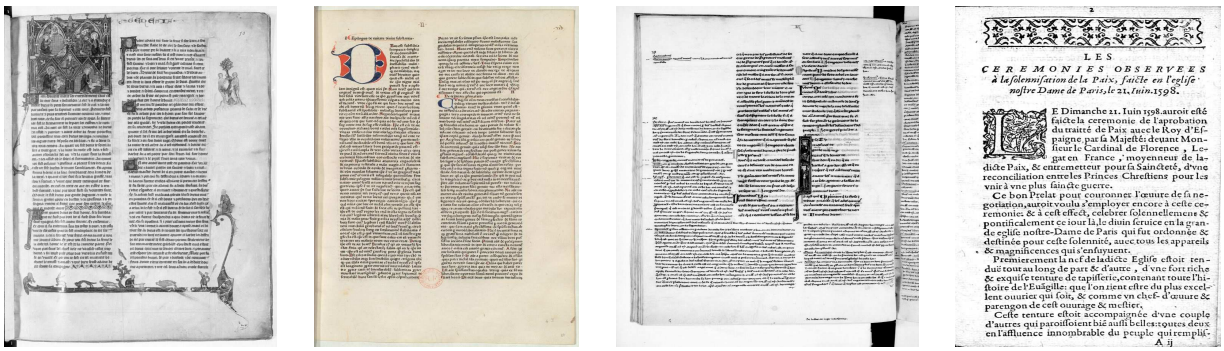
- 89 pages containing only two fonts
- 60 pages containing only three fonts
- 108 pages containing graphics and a single text font
- 57 pages containing graphics and text with two different fonts

We want to know firstly if the distribution of this dataset shows whether or not the extracted autocorrelation features are adequate for segmenting graphical regions from textual ones, and secondly if they can discriminate between texts with a variety of fonts and scales.

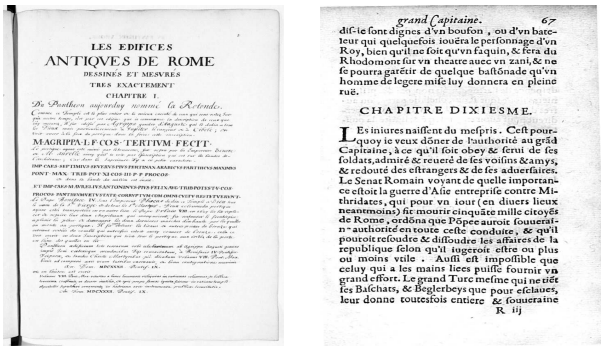
Secondly, another dataset of ancient document images is selected (*"2nd dataset"*) to assess the other phases of our proposed framework. 316 pages without any modification (*"2nd dataset"*) are selected from 13 books in two categories: 7 printed monographs and 6 manuscripts that encompass six centuries of French history (1200-1900). For each category, three kinds of content are selected:

- 110 pages containing only two fonts
- 100 pages containing graphics and a single text font
- 106 pages containing graphics and text with two different fonts

[22] `http://www.bnf.fr/fr/acc/x.accueil.html`



(a) Manuscript-One font and graphics (b) Printed-One font and graphics (c) Manuscript-Two fonts and graphics (d) Printed-Two fonts and graphics



(e) Manuscript-Only two fonts (f) Printed-Only two fonts

Fig. 10: Examples of our ancient document corpus.

Some examples of our corpus are shown in Figure 10. Our corpus is composed of grayscale/color documents which are digitized at 300/400 dpi and saved in the "TIFF" format, which provides a high resolution of digitized images.

An evaluation of segmentation and region classification requires a ground truth which is performed using the ground truthing editor, GEDI (Ground truthing Environment for Document Images)[23], a public domain document image annotation tool that labels spatial boundaries of regions [121]. Thus, by defining our ground truth rectangular regions drawn around each selected zone and different precise labels for regions with different fonts are manually defined (Figure 11-(f)). Furthermore, each rectangular region is characterized by its location on the page, its height, its width and a label.

4.2 Clustering and classification accuracy metrics

Since the texture feature extraction step and the clustering phase have been performed, we need to assess the segmentation method and the clustering results in order to evaluate our approach. Several indices have been presented [122–124]. Two kinds of measures have been proposed in the literature: internal and external measures [34, 125, 126].

- *Internal or unsupervised measures* evaluate the quality of clustering by considering only the intrinsic information concerning the distribution of the observations into clusters.
- *External or supervised measures* compare the distributions of the observations in the clustering results and the ground truth.

Nevertheless, the lack of appropriate quantitative measures for segmentation quality and the difficulty in defining criteria for specific application-dependent segmentation are the shortcomings that limit researchers in an objective unsupervised evaluation of their results. For example, Silva proposed two metrics, completeness and purity, to evaluate performance in document analysis applied specifically to tables [127]. General surveys of

[23] `http://gedigroundtruth.sourceforge.net/`

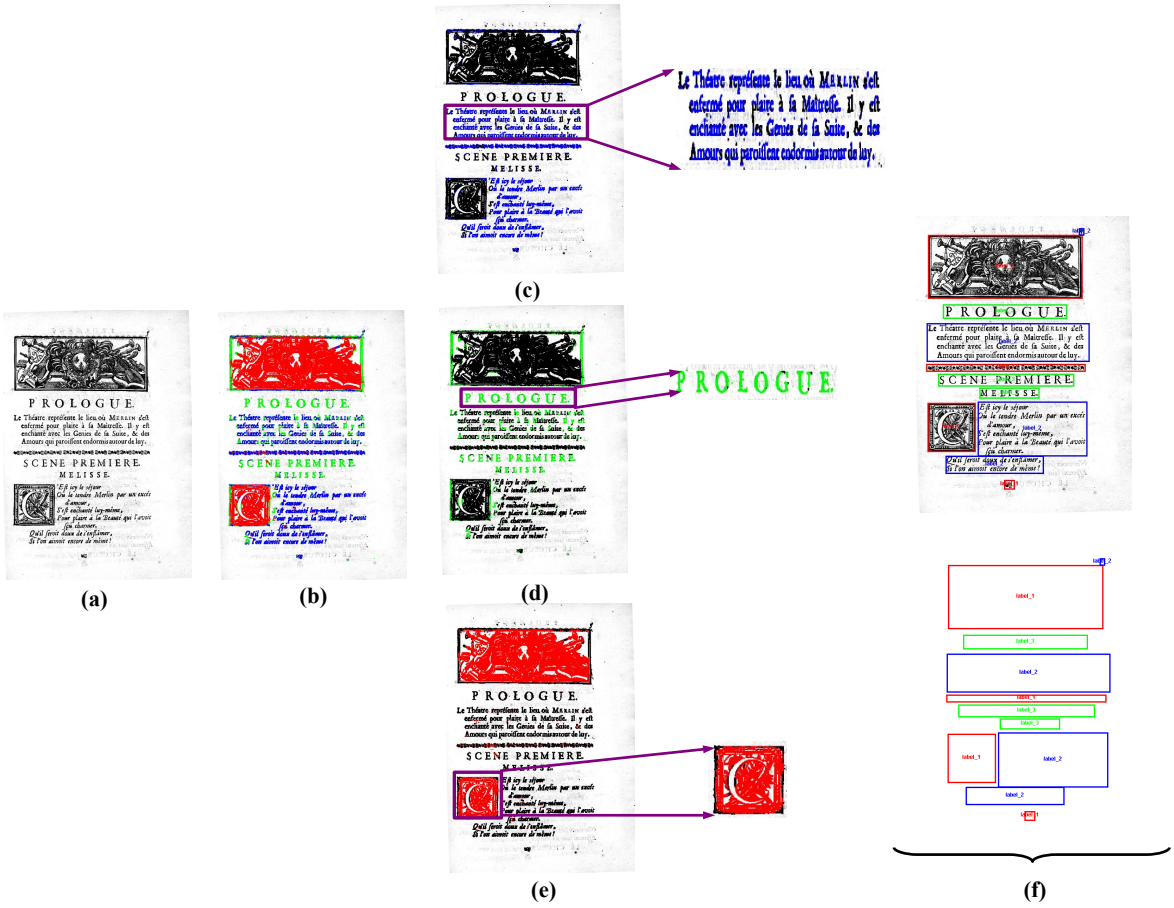


Fig. 11: Example of a pixel labeling result: **(a)** original document image, **(b)** final result of the proposed framework, **(c)** cluster representing the text with normal and italic font, **(d)** cluster representing the uppercase-text font, **(e)** cluster representing the graphics, and **(f)** ground truth.

segmentation method evaluation have been proposed in the literature [122–124]. For instance, one group evaluated their segmentation using the Jaccard coefficient [122]. However, this coefficient is not suitable for assessing the accuracy of our method because our goal is not an accurate pixel-based segmentation; we are more interested in finding homogeneous or similar content regions defined by similar textural indices. Given this objective, an external evaluation metric, Purity Per Block metric ($PPB(B,G)$) was defined by Mehri *et al.*, which evaluated the accuracy of a segmentation approach in terms of matching regions between the ground truth and pixel labeling regions [32, 53]. $PPB(B,G)$ is defined as:

$$PPB(B,G) = \frac{1}{|G|}\sum_{j}\frac{1}{|\{b_i \in g_j\}|}C_j \tag{16}$$

where

$$C_j = \max_{1\leq k\leq k_{opt}}(|b_i,(b_i \in g_j)\wedge(l_{B_i}=k)|)$$

where $|.|$ is the number of pixels in a given block; $B = \{b_1,b_2,...,b_i,...,b_n\}$ and $G = \{g_1,g_2,...,g_j,...,g_m\}$ are the sets of result blocks and rectangular regions of the ground truth, respectively. $L_B = \{l_{B_1},l_{B_2},...,l_{B_i},...,l_{B_n}\}$ corresponds to a set of labels obtained with our clustering methodology.

In order to provide an additional analysis and comparison of our approach and get an insight into the classification accuracy, a confusion matrix (or error matrix or contingency table) [128, 129] is computed. From the confusion matrix, several classification accuracy metrics are deduced, including entropy, purity, precision, recall, F-score or F-measure and classification accuracy rate [130–134]. These accuracy metrics are related to how representative the clusters are of classes and help to determine classes which are not able to segregate groups of data and then note the confusion and misclassification.

To evaluate quantitatively the different results obtained with our framework, internal and external clustering and classification accuracy measures are computed: silhouette width (SW) [89], Jaccard accuracy (J) [135], Fowlkes-Mallows accuracy (FM) [136], purity per block (PPB) [32, 53], purity (PT) [137], entropy (E) [137], precision (P), recall (R) and classification accuracy (CA) [132].

5 Evaluation and results

To prove the robustness of the proposed framework and provide additional insights into its classification accuracy, the clustering results of the extracted textural features on a set of simplified ancient documents are firstly assessed. This evaluation is designed to show the performance of the extracted features (Section 3.1) by using the first dataset of simplified ancient documents (*"1st dataset"*). Then, a second validation step evaluates the other phases of our framework (blocks 1 and 2, Figure 3, Section 3.3) by using the second dataset of ancient documents without any modification (*"2nd dataset"*). Several clustering accuracy metrics and classification accuracy rates are computed to assess the different phases of the framework and subsequently to evaluate its performance.

5.1 Evaluation of the extracted textural features

In order to evaluate the performance of the extracted texture features and assess the discriminating power of the extracted attributes, we first compute the autocorrelation features, on the 314 simplified historical documents of the *"1st dataset"*, as described in Section 3.1. Our objective in the evaluation of the extracted feature by using the *"1st dataset"* is to find out if the extracted autocorrelation features are adequate for segmenting graphical regions from textual ones and for discriminating between texts with a variety of fonts and scales.

A clustering step on the extracted textural attributes from the *"1st dataset"* is performed to validate their robustness. The adapted HAC with the Ward criterion (Section 3.3) is performed on the normalized textural features by setting the maximum number of homogeneous regions to the one defined in our ground truth. The texture feature vectors are normalized to zero mean and unit standard deviation to avoid domination by the higher numerical range of certain features.

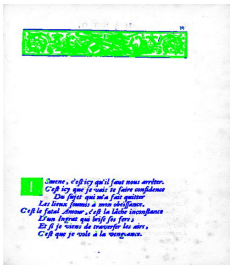
In order to validate our choice of clustering method, *i.e.* adapted HAC with the Ward criterion (block 1, Figure 3, Section 3.3), we compare the clustering results with those obtained using a standard clustering technique, the k-means algorithm. The k-means algorithm is also used with normalized textural features by setting the k clusters, *i.e.* regions of homogeneous or similar content, to that defined in our ground truth. The k-means algorithm partitions the samples into k clusters by using the squared Euclidean distance (SED) [118]. The $SED(x)$ of two multivariate vectors $x = (x_1, x_2, \ldots, x_{N^f})^T$ and $y = (y_1, y_2, \ldots, y_{N^f})^T$ is defined as:

$$SED(x,y) = \sum_{i=0}^{N^f} (y_i - x_i)^2 \tag{17}$$

5.1.1 Qualitative results

The pixel clustering results using the adapted HAC algorithm with the Ward criterion and k-means techniques are illustrated in Figure 12. A visual comparison of the results using the two clustering algorithms indicates that better results are obtained with the HAC technique than with the k-means algorithm (Figure 12). The good results obtained using the HAC algorithm (Figure 12) validate our choice in the clustering phase of the proposed framework (block 1, Figure 3). The pixel clustering results for the extracted texture features obtained with the adapted HAC show a much greater discriminating power for separating text (single font) and graphic regions (Figures 12(a)$\longrightarrow$12(c)) than for distinguishing documents containing graphics and two or more text fonts (Figures 12(d)$\longrightarrow$12(g)). Moreover, the extracted indices distinguishes two different text fonts, the italic and uppercase fonts (Figure 12(h)). On the other hand, in Figures 12(d), 12(e), and 12(g), where the documents contain two fonts and graphics, these features can not separate textual regions with different sizes and fonts and then generate two clusters for graphic regions. This may be explained by the fact that the autocorrelation attributes generally provide the main orientation of a texture (horizontal orientation for textual regions, while many orientations are present to different extents in graphic blocks). Thus, the autocorrelation descriptors are more suitable for documents containing one font and graphics.

HAC

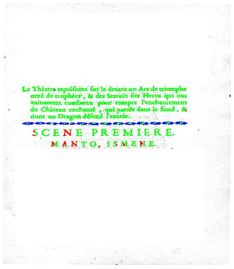


(a) $PPB = 0,96$ $P = 0,95$ $R = 0,93$ $CA = 0,94$

(b) $PPB = 0,97$ $P = 0,97$ $R = 0,96$ $CA = 0,96$

(c) $PPB = 0,90$ $P = 0,96$ $R = 0,96$ $CA = 0,96$

(d) $PPB = 0,73$ $P = 0,63$ $R = 0,92$ $CA = 0,91$



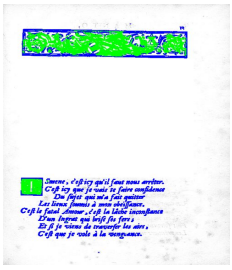
(e) $PPB = 0,85$ $P = 0,68$ $R = 0,91$ $CA = 0,80$

(f) $PPB = 0,82$ $P = 0,83$ $R = 0,81$ $CA = 0,85$

(g) $PPB = 0,91$ $P = 0,66$ $R = 0,87$ $CA = 0,85$

(h) $PPB = 0,75$ $P = 0,66$ $R = 0,64$ $CA = 0,82$

k-means

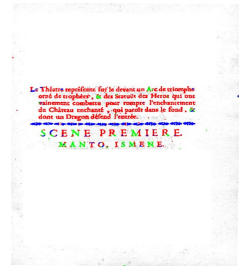


(i) $PPB = 0,87$ $P = 0,84$ $R = 0,80$ $CA = 0,78$

(j) $PPB = 0,90$ $P = 0,90$ $R = 0,93$ $CA = 0,91$

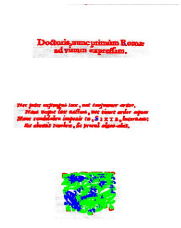
(k) $PPB = 0,87$ $P = 0,94$ $R = 0,94$ $CA = 0,94$

(l) $PPB = 0,73$ $P = 0,58$ $R = 0,81$ $CA = 0,76$



(m) $PPB = 0,78$ $P = 0,76$ $R = 0,82$ $CA = 0,82$

(n) $PPB = 0,94$ $P = 0,94$ $R = 0,86$ $CA = 0,91$



(o) $PPB = 0,88$ $P = 0,66$ $R = 0,87$ $CA = 0,84$

(p) $PPB = 0,65$ $P = 0,58$ $R = 0,85$ $CA = 0,83$

Fig. 12: Examples of results of the clustering of the extracted texture features from simplified historical documents "1st *dataset*" using the adapted HAC with the Ward criterion and the k-means algorithms. The HAC and k-means algorithms are used with the normalized textural features by setting the maximum k clusters to that defined in our ground truth. Figures 12(a)⟶12(c) and 12(i)⟶12(k) represent documents containing graphics and a single text font (2 clusters) analyzed with the HAC and k-means algorithms, respectively. Figures 12(d)⟶12(g) and 12(l)⟶12(o) represent documents containing graphics and two fonts (3 clusters) analyzed with the HAC and k-means algorithms, respectively. Figures 12(h) and 12(p) represent documents containing only three fonts (3 clusters) analyzed with the HAC and k-means algorithms, respectively. Since the process is unsupervised, the colors attributed to text or graphics may differ from one document image to another.

5.1.2 Quantitative results

This way of assessing the effectiveness of a segmentation method is inherently a subjective evaluation and we need to assess the robustness of the clustering results of the autocorrelation features using an appropriate quantitative metric. Thus, we compute several clustering accuracy measures: silhouette width (SW), Jaccard accuracy (J), Fowlkes-Mallows accuracy (FM) and purity per block (PPB) in order to quantitatively evaluate the different results. Therefore, an additional analysis and comparison with the classification accuracy metrics is needed to evaluate the performance of the extracted textural features, first to separate text and graphic regions, and secondly to segment text with different fonts. The confusion matrix (or error matrix or contingency table) is therefore computed for each document to get an insight into the accuracy of our framework. Several classification accuracy metrics are calculated from the confusion matrix, including purity (PT), entropy (E), precision (P), recall (R) and classification accuracy (CA). These metrics help to determine how good the clustering is and to compare cluster and class memberships [133]. As an evaluation clustering criterion, it is assumed that preference would be given to higher PT, P, R, CA and PPB and to lower E. Measures of PPB and classification accuracy metrics (P, R and CA) are presented at the bottom of each image in Figure 12.

Tables and Figure 1 show the clustering and classification accuracy results with 314 simplified historical documents *"1st dataset"* using the k-means and the adapted HAC algorithms. The higher the values, the better the results (except E, where lower values are better). The computed accuracy clustering and classification values are very promising. In the tables there are two *"Overall"* values. The *"Overall*"* value is obtained by averaging all the respective column values except the value of *"Two fonts and graphics**"*. The *"Overall**"* value is obtained by averaging all the respective column values except the value of *"Two fonts and graphics*"*. *"Two fonts and graphics*"* represents the case when every font in the text has a different label in the ground truth, and clustering is performed by setting the number of types of content regions to 3 (graphics and two different text fonts). *"Two fonts and graphics**"* represents the case when all fonts in the text have the same label in the ground truth, and clustering is performed by setting the number of types of content regions equal to 2 (graphics and text). This distribution indicates out which texture features can be more suitable for segmenting a document containing two text fonts and graphics into two/three classes, *i.e.* separating two different text fonts when a document contains graphics.

We conclude from Tables and Figure 1 that the best performances for the clustering evaluation metrics are obtained for documents containing graphics and single text font: 78%(J), 86%(FM), 91%(PPB) using the HAC algorithm, and $0,56$(SW) when using the k-means. However, the best performances (91%(PT) and $0,30$%(E)) are obtained with the *"Only two fonts/HAC"* document category, whereas the best performances are obtained for the *"Two fonts and graphics**"* document category with the following clustering evaluation metrics: 89%(P), 88%(R) and 90%(CA) when using the HAC algorithm. We conclude that each category of evaluation metrics is in agreement with a specific category of document, depending on the objective, the process and the specificity of each computed accuracy clustering and classification, *i.e.* "J and PPB", "PT and E", and "P, R and CA". The second best results (90%(PT), $0,36$(E), 86%(P), 86%(R) and 88%(CA)) are obtained for documents containing graphics and single text font using the HAC algorithm. The lowest values of the most computed clustering and classification accuracy metrics are obtained for documents containing only three fonts when the k-means and HAC algorithms are used. In conclusion, by computing numerous clustering and classification evaluation metrics and by using the adapted HAC on the extracted autocorrelation descriptors, we prove that it is possible to separate textual from graphic regions in historical documents.

5.2 Evaluation of other framework phases

In this section, the evaluation of the other phases of the proposed framework (blocks 1 and 2, Figure 3, Sections 3.2 and 3.3) is described: the estimation of the number of homogeneous (Section 3.2), pixel clustering (Section 3.3.1) and pixel labeling (Section 3.3.2) using the second dataset of ancient documents without any modification: *"2nd dataset"*.

5.2.1 Evaluation of the estimation of the number of homogeneous regions

The estimated number of homogeneous regions is obtained using the merge CC method with the extracted features of a number of selected foreground pixels chosen randomly from the pages of a book (block 2, Figure 4, Section 3.2).

An example of Δk is shown in Figure 13(b). In this experiment, k_{opt} is equal to 3 and is estimated from the peak in Δk values of the merge curve.

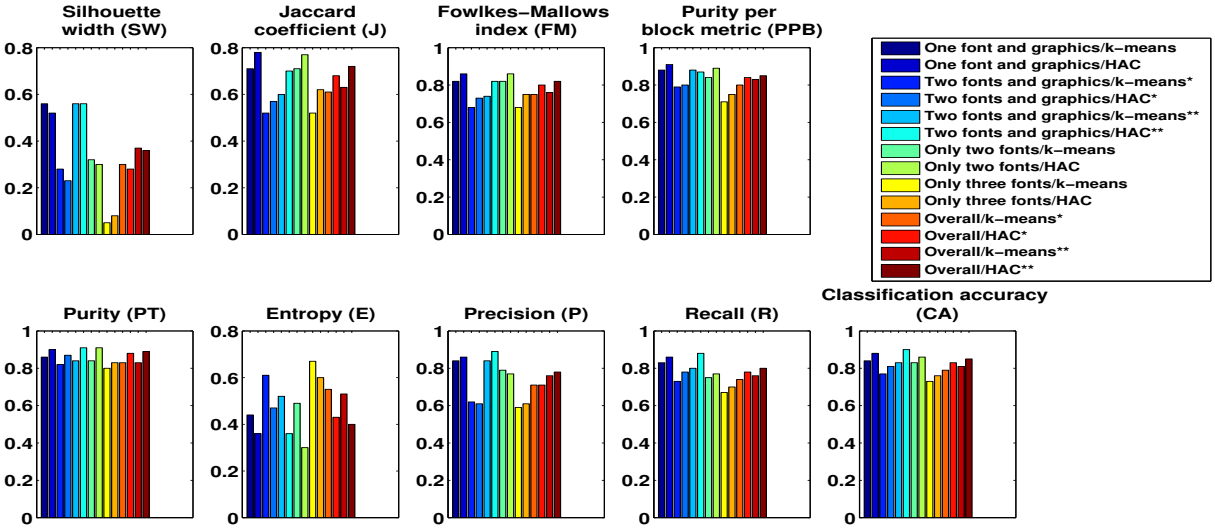
Table 1: Evaluation of the extracted features by clustering and classification accuracy measures on 314 simplified historical documents *"1st dataset"* using the HAC and k-means algorithms: silhouette width (SW), Jaccard coefficient (J), Fowlkes-Mallows index (FM), purity per block metric (PPB), purity (PT), entropy (E), precision (P), recall (R) and classification accuracy (CA). $\mu(.)$ and $\sigma(.)$ are the mean and standard deviation of $(.)$, respectively. The higher the values, the better the results (except E, where lower values are better). The *"Overall*"* value is obtained by averaging all the respective column values except the value of *"Two fonts and graphics**"*. The *"Overall**"* value is obtained by averaging all the respective column values except the value of *"Two fonts and graphics*"*. *"Two fonts and graphics*"* represents the case when every font in the text has a different label in the ground truth, and clustering is performed by setting the number of types of content regions to 3 (graphics and two different text fonts). *"Two fonts and graphics**"* represents the case when all fonts in the text have the same label in the ground truth, and clustering is performed by setting the number of types of content regions equal to 2 (graphics and text).

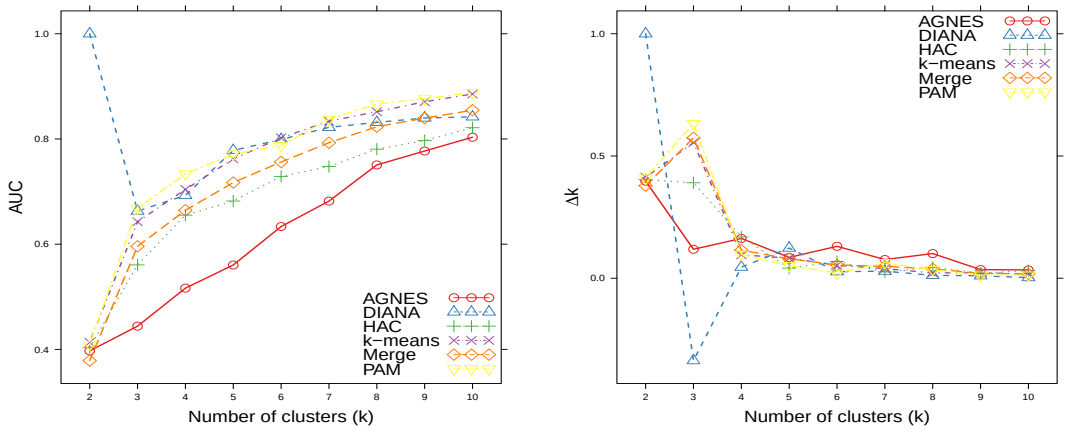
	Document content	$\mu(SW)$	$\sigma(SW)$	$\mu(J)$	$\sigma(J)$	$\mu(FM)$	$\sigma(FM)$	$\mu(PPB)$	$\sigma(PPB)$
k-means	One font and graphics	0,56	0,21	0,71	0,18	0,82	0,13	0,88	0,09
	Two fonts and graphics*	0,28	0,13	0,52	0,14	0,68	0,11	0,79	0,08
	Two fonts and graphics**	0,56	0,20	0,60	0,14	0,74	0,11	0,88	0,08
	Only two fonts	0,32	0,24	0,71	0,17	0,82	0,12	0,84	0,11
	Only three fonts	0,05	0,14	0,52	0,17	0,68	0,14	0,71	0,12
	Overall*	**0,30**	**0,18**	**0,61**	**0,16**	**0,75**	**0,12**	**0,80**	**0,10**
	Overall**	**0,37**	**0,20**	**0,63**	**0,16**	**0,76**	**0,12**	**0,83**	**0,10**

	Document content	$\mu(SW)$	$\sigma(SW)$	$\mu(J)$	$\sigma(J)$	$\mu(FM)$	$\sigma(FM)$	$\mu(PPB)$	$\sigma(PPB)$
HAC	One font and graphics	0,52	0,21	0,78	0,18	0,86	0,12	0,91	0,09
	Two fonts and graphics*	0,23	0,17	0,57	0,15	0,73	0,11	0,80	0,08
	Two fonts and graphics**	0,56	0,17	0,70	0,15	0,82	0,11	0,87	0,07
	Only two fonts	0,30	0,21	0,77	0,17	0,86	0,11	0,89	0,08
	Only three fonts	0,08	0,19	0,62	0,18	0,75	0,14	0,75	0,08
	Overall*	**0,28**	**0,19**	**0,68**	**0,17**	**0,80**	**0,12**	**0,84**	**0,08**
	Overall**	**0,36**	**0,19**	**0,72**	**0,17**	**0,82**	**0,12**	**0,85**	**0,08**

	Document content	$\mu(PT)$	$\sigma(PT)$	$\mu(E)$	$\sigma(E)$	$\mu(P)$	$\sigma(P)$	$\mu(R)$	$\sigma(R)$	$\mu(CA)$	$\sigma(CA)$
k-means	One font and graphics	0,86	0,14	0,44	0,28	0,84	0,15	0,83	0,16	0,84	0,16
	Two fonts and graphics*	0,82	0,13	0,61	0,32	0,62	0,16	0,73	0,15	0,77	0,12
	Two fonts and graphics**	0,84	0,11	0,52	0,25	0,84	0,12	0,80	0,14	0,83	0,13
	Only two fonts	0,84	0,14	0,49	0,30	0,79	0,17	0,75	0,15	0,83	0,14
	Only three fonts	0,80	0,16	0,67	0,38	0,59	0,11	0,67	0,15	0,73	0,16
	Overall*	**0,83**	**0,14**	**0,55**	**0,32**	**0,71**	**0,15**	**0,74**	**0,15**	**0,79**	**0,14**
	Overall**	**0,83**	**0,14**	**0,53**	**0,30**	**0,76**	**0,14**	**0,76**	**0,15**	**0,81**	**0,15**

	Document content	$\mu(PT)$	$\sigma(PT)$	$\mu(E)$	$\sigma(E)$	$\mu(P)$	$\sigma(P)$	$\mu(R)$	$\sigma(R)$	$\mu(CA)$	$\sigma(CA)$
HAC	One font and graphics	0,90	0,13	0,36	0,27	0,86	0,17	0,86	0,16	0,88	0,15
	Two fonts and graphics*	0,87	0,11	0,47	0,29	0,61	0,19	0,78	0,14	0,81	0,12
	Two fonts and graphics**	0,91	0,08	0,36	0,21	0,89	0,08	0,88	0,10	0,90	0,09
	Only two fonts	0,91	0,10	0,30	0,20	0,77	0,18	0,77	0,17	0,86	0,15
	Only three fonts	0,83	0,10	0,60	0,29	0,61	0,09	0,70	0,13	0,76	0,12
	Overall*	**0,88**	**0,11**	**0,43**	**0,26**	**0,71**	**0,16**	**0,78**	**0,15**	**0,83**	**0,13**
	Overall**	**0,89**	**0,10**	**0,40**	**0,24**	**0,78**	**0,13**	**0,80**	**0,14**	**0,85**	**0,13**





(a) (b)

Fig. 13: Consensus clustering: **(a)** plot of Area Under the Cumulative density curve (AUC) for the consensus matrix for each clustering experiment against number of clusters k. **(b)** plot of Δk changes in AUC for the consensus matrix for each clustering experiment against number of clusters k. Using the three hierarchical clustering methods: AGNES, DIANA and HAC give an estimate of 2 as the optimal number of clusters, while an estimate of 3 is obtained with k-means, PAM and Merge (the orange curve representing the merge consensus clustering peaked at $k = 3$).

Table 2 shows 10 examples of the estimation of the number of homogeneous regions. These examples illustrate 10 estimations computed from 2 different books containing graphics and single text font using CC and different clustering techniques. For each set of 1000 randomly selected foreground pixels from 10 pages also selected randomly from the same book, we compare the estimated number of homogeneous regions using 5 clustering methods and the merge CC technique with the number of clusters defined in our ground truth. For most of these estimations, carried out using the two partitioning clustering methods (PAM and k-means), the estimated number of clusters is similar to that defined in our ground truth. However, there is a slight variability in the number of clusters estimated by the three hierarchical clustering methods: AGNES, DIANA and HAC. This may be explained by the presence of noise in the analyzed historical documents, which can have an impact on the estimated number of homogeneous regions. Although a slight variability in the estimated number of homogeneous regions is observed when the merge CC technique is used, the results are relatively consistent since noise and degradation information are taken into consideration. Noise and degradation information can be considered as particular textured areas, *i.e.* characterized by different texture features vectors, and which can subsequently constitute a separate cluster. In addition, the results estimate using the PAM method are relatively similar to those obtained with the merge CC technique and also the number of clusters defined in the ground truth. Thus, this confirms our hypothesis that the partitioning clustering methods are relatively robust and justifies the higher weight of $\frac{1}{4}$ assigned to each partitioning clustering algorithm (PAM and k-means).

– 1000 *vs.* 2000 pixels are used in the CC technique

To analyze the robustness of the estimation obtained with the merge CC technique in our framework, the number of selected foreground pixels introduced as input is varied. Table 2 shows the evaluation of the estimation of the number of homogeneous or similar content regions for an analysis 10 sets of 2000 randomly selected pixels from the same 2 different historical books. The results are in agreement with the number of clusters defined in the ground truth. We conclude that the effectiveness of the merge CC technique depends on the number of observations. The higher the number of observations, *i.e.* the number of randomly selected foreground pixels, the better the estimation results. Nevertheless, when numerical complexity is taken into account, a high number of observations requires 16 times the computation time to obtain half of the observations.

– The merge CC technique *vs.* internal clustering evaluation measures

In the following tests, the results of the estimated number of homogeneous regions are compared using the merge CC method and various internal clustering evaluation measures. For each estimation approach the sum of the

Table 2: Examples of the estimation of the number of homogeneous regions by analyzing 10 sets of 1000 and 2000 randomly selected pixels from 2 different historical books containing graphics and single text font with the merge CC and different clustering techniques.

		Set number											Set number									
		Book 1					Book 2						Book 1					Book 2				
	1000 pixels	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}	7^{th}	8^{th}	9^{th}	10^{th}	2000 pixels	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}	7^{th}	8^{th}	9^{th}	10^{th}
AGNES		2	2	2	2	2	6	4	6	4	5		5	3	2	3	2	5	4	6	4	5
DIANA		2	4	2	2	2	3	3	2	2	3		2	2	2	2	2	3	2	3	2	3
HAC		2	3	2	3	2	3	3	3	3	4		2	3	2	2	2	4	2	3	3	4
k-means		2	2	2	2	2	2	2	2	2	3		2	2	2	2	2	2	2	2	2	3
PAM		2	2	2	2	2	2	2	2	2	3		2	2	2	2	2	2	2	2	2	2
Merge		2	2	2	2	2	3	3	3	3	3		2	2	2	2	2	2	2	2	2	2
Ground truth		2	2	2	2	2	2	2	2	2	2		2	2	2	2	2	2	2	2	2	2

differences between the number of homogeneous regions defined in the ground truth and the estimated number of homogeneous regions is computed in order to quantify the difference between the number of clusters *vs.* classes. The lower the value for the difference between the number of clusters *vs.* classes, the better the results. The difference between the number of clusters *vs.* classes (D_k) is defined as:

$$D_k(k_{est}, k_{gt}) = \sum_i |k^i_{gt} - k^i_{opt}| \tag{18}$$

where k_{est} is the estimated number of clusters, k_{gt} is the number of clusters defined in the ground truth, and i represents the i^{th} execution of the CC algorithm on the selected foreground pixels from the digitized book.

Figure 14 represents for each category of book the difference between the number of clusters *vs.* classes (D_k) for the CC method and 21 different internal clustering evaluation measures (Krzanowski-Lai index [138], Hartigan index [139], Calinski-Harabasz index [140], Cubic Clustering Criterion [141], Scott index [142], Marriot index [143], TraceCovW index [144], TraceW index [144], Friedman index [145], Rubin index [146], C-index [147], Davies-Bouldin index [148], silhouette width index [89], Ratkowsky index [149], Ball index [150], PtBiserial index [151], Frey index [152], McClain index [153], Dunn index [154], SDindex [155] and SDbw validity index [156]). Furthermore, D_k is presented for the CC technique and the unsupervised evaluation measures for all categories. The lowest differences between the number of clusters *vs.* classes for all of our document corpus are 17, 17 and 16 computed using respectively the CC, Frey index and McClain index, respectively. Figure 14 shows a null value for the difference between the number of clusters *vs.* classes for the Frey index and the McClain index for the printed document category (one font and graphics) but this increases for the other categories. The CC method provides good results for all categories of our page corpus, as shown in Figure 14. The different experiments detailed in this section prove that the merge CC technique provides good results by comparing its estimation of the optimal number of clusters with the ground truth and the internal clustering evaluation indices. However, the CC method is relatively long so is not particularly effective for a very large dataset.

5.2.2 Evaluation of the pixel clustering phase

The next evaluation step consists of assessing the clustering method, *i.e.* the adapted HAC algorithm with the Ward criterion, used for the pixel clustering task of the proposed framework (block 1 on Figure 3, Section 3.3.1) on the *"*2^{nd} *dataset"*: 316 ancient documents without any modification.

Indeed, the clustering results obtained after analyzing the extracted autocorrelation features for the *"*1^{st} *dataset"* are encouraging and indicate many interesting perspectives (Figure 12). The same proposed clustering approach is therefore used on the corpus of ancient book pages. The pixel clustering results with the historical documents obtained using the HAC algorithm are illustrated in Figure 15. The results obtained with the *"*2^{nd} *dataset"* strengthen our previous observations with the *"*1^{st} *dataset"* (Figure 12). Thus, we confirm that the clustering task used with the extracted autocorrelation features have a much greater discriminating power for separating text (single font) and graphic regions (Figures 15(a) and 15(b)) than for distinguishing documents containing graphics and two or more text fonts (Figure 15(c)). The results also confirm that it is more difficult to separate two text fonts (Figure 15(d)).

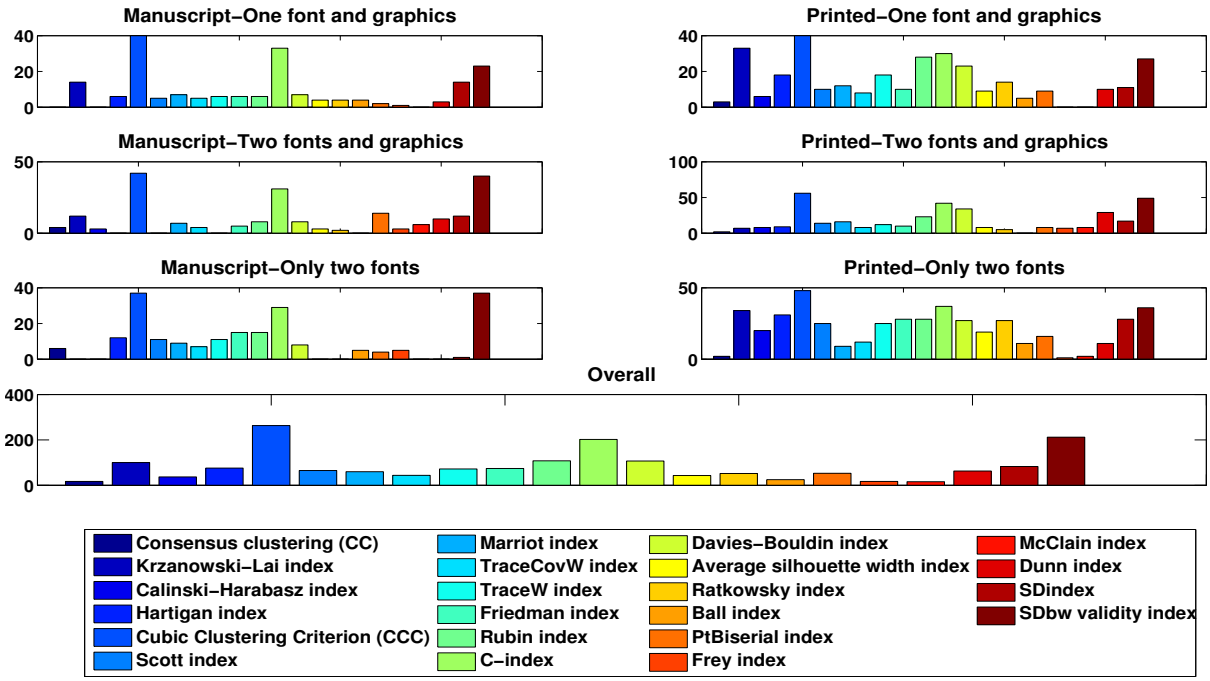
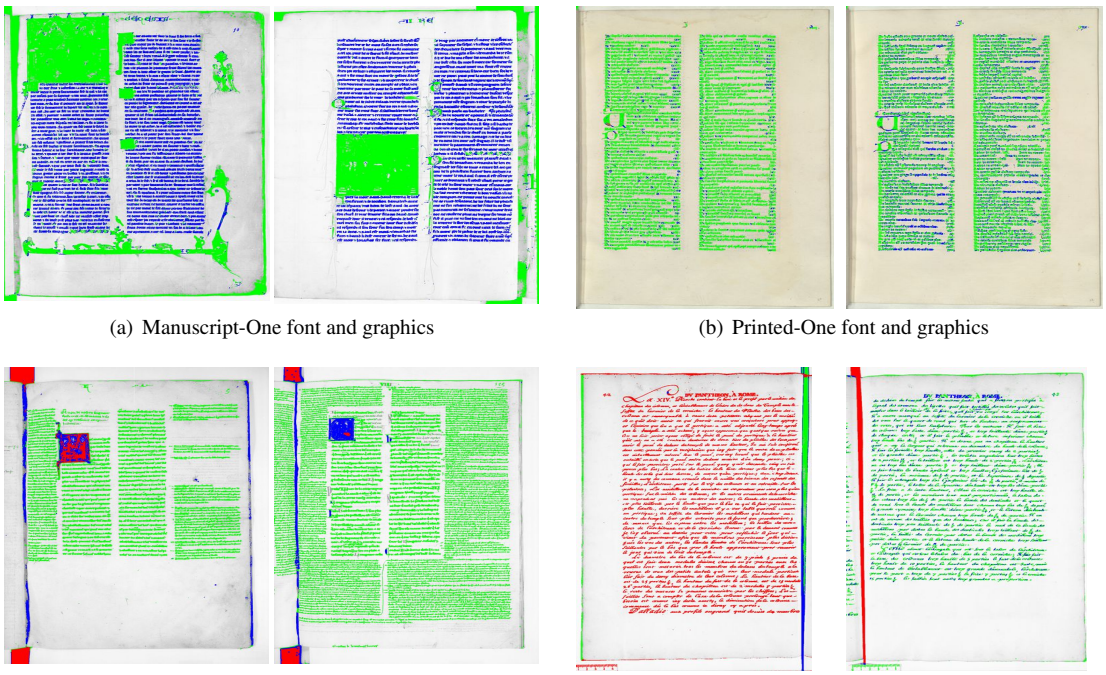


Fig. 14: Evaluation of the estimation of the number of homogeneous regions by using the CC method *vs.* various internal clustering evaluation measures: the estimated number of homogeneous regions is computed from 21 different internal clustering evaluation measures (Krzanowski-Lai index, Hartigan index, Calinski-Harabasz index, Cubic Clustering Criterion, Scott index, Marriot index, TraceCovW index, TraceW index, Friedman index, Rubin index, C-index, Davies-Bouldin index, silhouette width index, Ratkowsky index, Ball index, PtBiserial index, Frey index, McClain index, Dunn index, SDindex and SDbw validity index) and are compared with the estimated number using the CC method for each category of book (*"Manuscript-One font and graphics"*, *"Printed-One font and graphics"*, *"Manuscript-Two fonts and graphics"*, *"Printed-Two fonts and graphics"*, *"Manuscript-Only two fonts"* and *"Printed-Only two fonts"*). The difference between the number of clusters *vs.* classes for all our corpus is also shown. The lower the difference between the number of clusters *vs.* classes, the better the results.

5.2.3 Evaluation of the pixel labeling phase

Pixel labeling (block 1 on Figure 3, Section 3.3.2) is used to determine and assign the same cluster identifier to each similar cluster extracted from the digitized book. This step of the framework uses the NNS technique. This technique is used between each texture feature vector of each digitized page of the same book and the k_{opt} clusters of the selected foreground samples of a book in order to find the closest texture feature vector to the cluster of the selected foreground samples of a book, *i.e.* by selecting the minimum Mahalanobis distance (*MD*). In order to validate this task, the *MD* is compared with the Euclidean distance (*ED*) [157] when the NNS technique is used.

The results of this pixel labeling step using the *ED* and *MD* are illustrated in Figures 16 and 17, respectively. The success of the pixel labeling framework is demonstrated by visual inspection of the segmented documents (Figures 16 and 17). The proposed framework gives better results with the *MD*-based approach (Figures 17) and finds homogeneous regions in the content of digitized ancient books, *i.e.* for example on Figure 17(a) the graphic regions (green) and textual regions (blue) are similarly labeled in two different pages of the same book. It is clear from the four figures 17(b), 17(a), 17(d) and 17(c) that the document images are segmented into graphic regions, which correspond to an ornament and a drop cap, and textual regions. For the printed document category (two fonts and graphics) in Figure 17(d), the proposed approach distinguishes two different fonts, the normal (blue) and uppercase (green) fonts. On the other hand, Figure 17(e) shows that for the manuscript category (only two fonts), our method discriminates between the noise on the document image borders and the textual regions and separates textual regions with different sizes and fonts, italic and uppercase. However, Figure 17(f) suggests that for the printed document category (only two fonts), the proposed approach can not discriminate between the normal and uppercase fonts when the *MD* is used.



(a) Manuscript-One font and graphics

(b) Printed-One font and graphics

(c) Manuscript-Two fonts and graphics

(d) Manuscript-Only two fonts

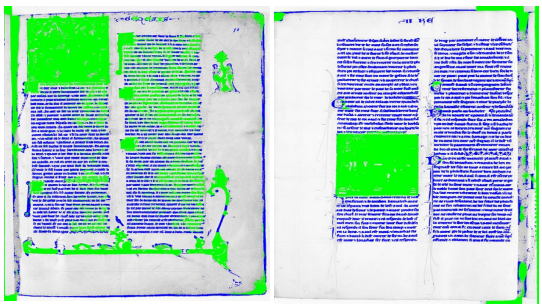
Fig. 15: Examples of the pixel clustering task used with the extracted texture features of historical book pages. Since the pixel labeling task is not processed, the colors attributed to text or graphics may differ from one document image to another.

- **Purity per block metric (*PPB*)**

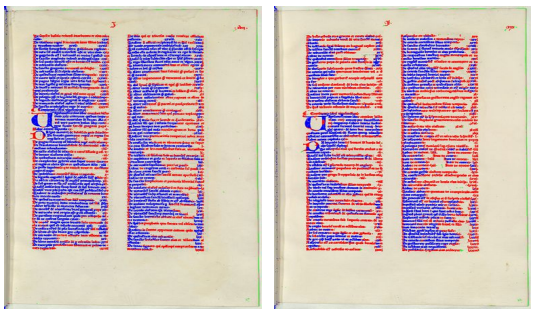
In this section, the purity per block metric (*PPB*) (equation (16)), which was defined by Mehri *et al.*, is computed to validate and evaluate a set of experiments [32, 53].

  - Euclidean distance (*ED*) *vs.* Mahalanobis distance (*MD*) is used in the NNS technique

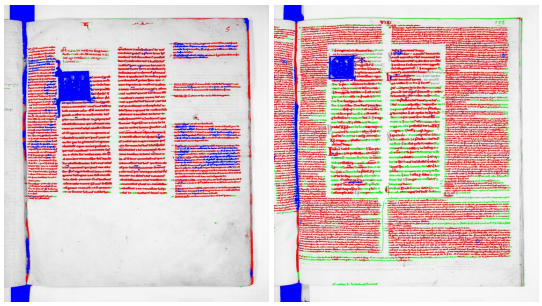
The results of *PPB* (equation (16)) are presented in Table 3 when the *ED* and the *MD* are used in the pixel labeling task (block 1 on Figure 3, Section 3.3.2) and after setting the number of randomly selected pixels from 10 pages selected randomly from the same book to 1000 (block 2 on Figure 3, Section 3.2). We obtain $87\% \pm 0.04$ and $85\% \pm 0.04$ mean *PPB* when using the *ED* and *MD* distances are used in the pixel labeling task, respectively. The overall results are quite satisfying, especially for the manuscript category which, contains textual (one or two fonts) and non-textual regions. The mean *PPB* when using the *ED* and *MD* are 91% and 92%, respectively, for the manuscript category (one font and graphics). It can be assumed that manuscripts contain graphic regions that are more compact and homogeneous than printed documents. 94% and 90% mean *PPB* are obtained with the *ED* and the *MD*, respectively, for the category of printed documents containing only two text fonts. The high mean *PPB* obtained for the printed document category (only two fonts) (Figure 17(f)) does not signify a good segmentation of the document content according to different types of content regions, *i.e.* different text font. However, it does give an idea about the level of region homogeneity. Hence, further analysis is required with numerous clustering and classification evaluation metrics. Mehri *et al.* found 80%(*PPB*) for the printed category (only two fonts) when using the *MD* [53], while in this work 90%(*PPB*) is obtained. This may be explained by the fact that a foreground pixel selection step using the Otsu method [101] is introduced in this work which helps improving the result. Indeed, this stage is more suitable for distinguishing the foreground and the background clusters. An overall slight improvement in the average value of *PPB* is noted in addition to a significant gain in computation time compared to the method proposed by Mehri *et al.* [53]. By comparing the average values of *PPB* for different document categories, a higher mean *PPB* is obtained for pages containing graphics and single text font when using the *ED* and the *MD*. This suggests that the extracted autocorrelation features can distinguish textual and graphical regions. When using the *ED*, an overall higher value of *PPB* is observed for printed documents than when using the *MD* for manuscripts. However, a higher standard deviation of *PPB* is observed when using the *ED* compared to the *MD*. Thus, our



(a) Manuscript-One font and graphics:
$PPB = 0,91$ $P = 0,74$ $R = 0,72$ $CA = 0,74$

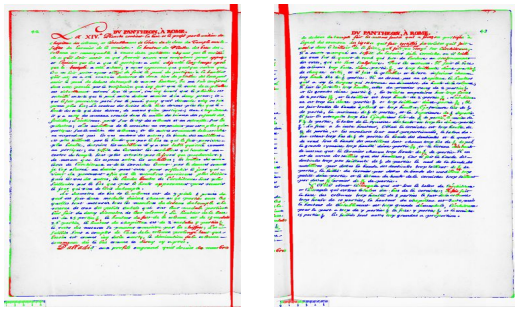


(b) Printed-One font and graphics:
$PPB = 0,88$ $P = 0,51$ $R = 0,42$ $CA = 0,22$

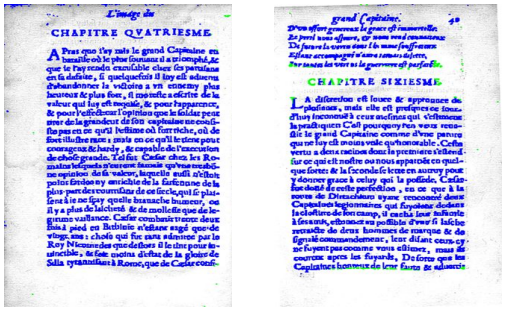


(c) Manuscript-Two fonts and graphics:
$PPB = 0,90$ $P = 0,75$ $R = 0,69$ $CA = 0,80$

(d) Printed-Two fonts and graphics:
$PPB = 0,82$ $P = 0,70$ $R = 0,70$ $CA = 0,66$



(e) Manuscript-Only two fonts:
$PPB = 0,81$ $P = 0,66$ $R = 0,57$ $CA = 0,78$



(f) Printed-Only two fonts:
$PPB = 0,94$ $P = 0,51$ $R = 0,52$ $CA = 0,73$

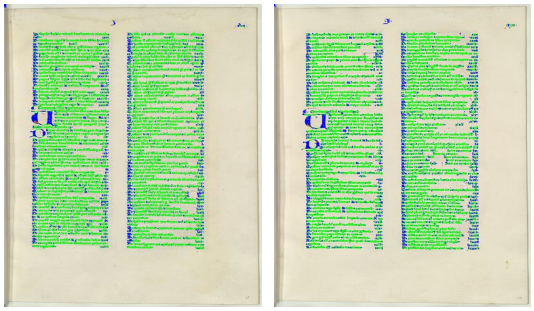
Fig. 16: Examples of the result with the proposed pixel labeling framework for digitized historical book content performed by introducing 1000 pixels into the CC technique and using the Euclidean distance in the pixel labeling task. For the same book, each cluster (represented by a given color) represents a similar or homogeneous region. Since the process is unsupervised, the colors attributed to text or graphics may differ from one book to another.

comparative analysis of the two distances (*ED* and *MD*) demonstrates that the synthesis of *PPB* values for different document categories gives different results depending on the context (text *vs.* graphics, manuscript *vs.* printed document). The analysis of *PPB* is not sufficient and an additional comparative synthesis is needed, with numerous clustering and classification evaluation accuracies. Nevertheless, if we compare the visual results when using the *ED* (Figure 16) and the *MD* (Figure 17), it is clear that the best results are obtained with the *MD*.

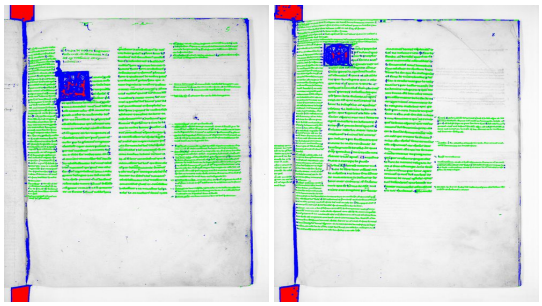
– 1000 *vs.* 2000 pixels are used in the CC technique

Since the use of the Mahalanobis distance (*MD*) in the proposed framework is validated using the NNS technique in the pixel labeling task (block 1 on Figure 3, Section 3.3.2), we will assess in this section the pixel labeling results with a variable number of selected pixels introduced as input in the estimation of the number of homogeneous regions and similar content regions (block 2 on Figure 4, Section 3.2). Thus, the difference in *PPB* accuracy metric is computed for the results of the pixel labeling task by setting the number of randomly selected pixels to 1000 and 2000 in the framework phase: the estimation of the number of the homogeneous regions and similar content regions

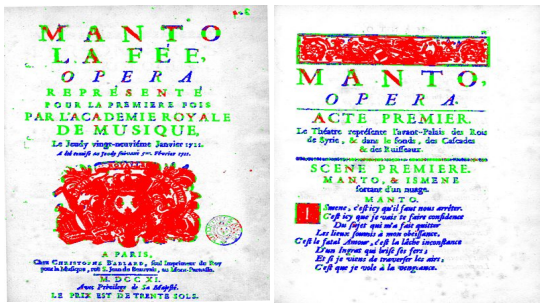
(a) Manuscript-One font and graphics:
$PPB = 0,92$ $P = 0,95$ $R = 0,92$ $CA = 0,94$



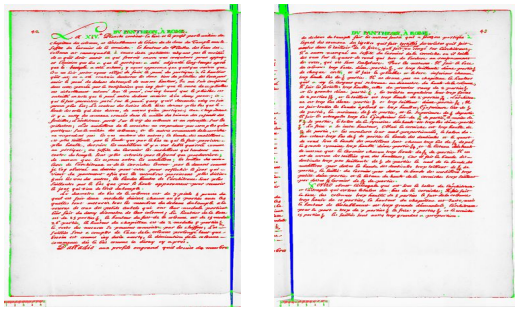
(b) Printed-One font and graphics:
$PPB = 0,77$ $P = 0,58$ $R = 0,51$ $CA = 0,75$



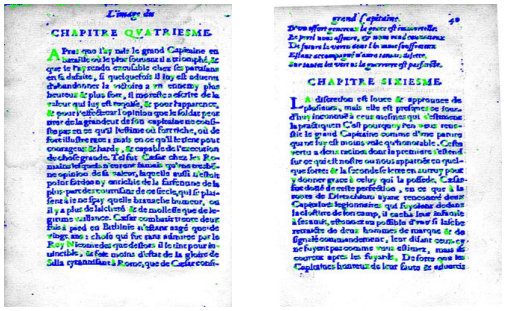
(c) Manuscript-Two fonts and graphics:
$PPB = 0,88$ $P = 0,73$ $R = 0,72$ $CA = 0,75$



(d) Printed-Two fonts and graphics:
$PPB = 0,76$ $P = 0,83$ $R = 0,82$ $CA = 0,82$



(e) Manuscript-Only two fonts:
$PPB = 0,85$ $P = 0,57$ $R = 0,70$ $CA = 0,77$



(f) Printed-Only two fonts:
$PPB = 0,90$ $P = 0,52$ $R = 0,50$ $CA = 0,72$

Fig. 17: Examples of the result with the proposed pixel labeling framework for digitized historical book content performed by introducing 1000 pixels into the CC technique and using the Mahalanobis distance in the pixel labeling task. For the same book, each cluster (represented by a given color) represents a similar or homogeneous region. Since the process is unsupervised, the colors attributed to text or graphics may differ from one book to another.

(block 2 on Figure 4, Section 3.2). On average, a similar level of pixel labeling is obtained when 1000 *vs.* 2000 pixels are used in the CC technique (Table 4) with an overall difference of 4%. We conclude that the fixed number of randomly selected pixels introduced in the estimation of the number of homogeneous regions and similar content regions does not have a negative impact on the results of the pixel labeling phase of our framework. Thus, we can limit the number of randomly selected pixels for this estimation to 1000.

- **Other clustering evaluation accuracies**

An additional analysis with different internal and external clustering evaluation indices is needed in order to evaluate our approach, to validate the external evaluation metric, the purity per block measure (*PPB*) (equation (16)), and the choice of computed distance. In this context, 12 clustering evaluation indices are computed: five internal (DaviesBouldin index [148], Dunn index [154], CalinskiHarabasz index [140], Hartigan index [139] and KrzanowskiLai index [138]) and seven external (Rand index [158], adjusted Rand index [159], mutual information

Table 3: Purity per block metric (*PPB*) results. μ and σ are the mean value and standard deviation values of *PPB*, respectively. *"⋆"*: NNS with the Euclidean distance (*ED*); *"⋆⋆"*: NNS with the Mahalanobis distance (*MD*).

Document category	Document content	Number of pages	$\mu^{\star}(PPB)$	$\sigma^{\star}(PPB)$	$\mu^{\star\star}(PPB)$	$\sigma^{\star\star}(PPB)$
Manuscript	One font and graphics	50	0,91	0,05	0,92	0,01
	Two fonts and graphics	56	0,90	0,07	0,88	0,04
	Only two fonts	50	0,81	0,09	0,85	0,05
	Overall	**156**	**0,87**	**0,07**	**0,88**	**0,03**
Printed	One font and graphics	50	0,88	0,14	0,77	0,05
	Two fonts and graphics	50	0,82	0,07	0,76	0,04
	Only two fonts	60	0,94	0,05	0,90	0,08
	Overall	**160**	**0,88**	**0,09**	**0,81**	**0,03**
Overall		**316**	**0,87**	**0,08**	**0,85**	**0,04**

Table 4: Difference in *PPB* for pixel labeling when 1000 *vs.* 2000 pixels are used in the CC technique.

Document category	Document content	*PPB*
Manuscript	One font and graphics	0,01
	Two fonts and graphics	0,06
	Only two fonts	0,12
	Overall	0,06
Printed	One font and graphics	0,01
	Two fonts and graphics	0,02
	Only two fonts	0,05
	Overall	0,03
Overall		0,04

measure [160], adjusted mutual information measure [161], *J* [135], *FM* [136] and *PPB* (equation (16)) [32, 53]). The higher the values, the better the results (except the Davies-Bouldin index, where lower values are better). Numerous clustering evaluation indices are computed using the results of the pixel labeling phase of our framework with the two distances, *ED* and *MD*.

Figure 18 shows that the best clustering results are obtained with the most computed evaluation accuracy metrics obtained for the manuscript category (one font and graphics) when using the *ED* and *MD*. *J* and *FM* are congruent when using *MD* in pixel labeling for the following categories of book: *"Manuscript-One font and graphics"*, *"Manuscript-Two fonts and graphics"*, *"Printed-Two fonts and graphics"*, *"Manuscript-Only two fonts"*, *"Printed-Only two fonts"* and *"Overall"*. However, the results obtained using the two computed distances vary and this is observed with the other evaluation accuracy metrics. Moreover, the second best result is obtained for the manuscript category (only two fonts) with the following six accuracy clustering metrics: DaviesBouldin index, CalinskiHarabasz index, Rand index, *J*, *FM* and *PPB*. This may be explained by the fact that the autocorrelation features discriminate between the noise in the document image and the textual regions (Figure 17(e)). Three measures (adjusted Rand index, mutual information measure and adjusted mutual information measure) give the second best clustering result for the printed category (two fonts and graphics) (Figure 17(d)). We can confirm that the probability and information theory based accuracies are relatively concordant. The *J*, *FM* and rand index show that the lowest values are obtained for manuscript documents (two fonts and graphics). On the other hand, the lowest outcomes for both the *PPB* and Davies-Bouldin indices are observed in the printed document category (two fonts and graphics). The three probability and information theory based accuracies (adjusted Rand index, mutual information measure and adjusted mutual information measure) and the CalinskiHarabasz index, the Hartigan index and the KrzanowskiLai index, suggest that the printed document category (only two fonts) has the lowest outcomes (Figure 17(f)). It should remembered that the various supervised and unsupervised measures do not evaluate and assess the same aspects. We conclude that the results of the *PPB* are relatively similar to the various internal and external clustering evaluation indices presented previously. The best clustering results are obtained for the manuscript category (one font and graphics) with the different clustering evaluation metrics. This strengthens our previous results and confirms our assumption that the autocorrelation attributes generally provide the main orientation of a texture (horizontal orientation for textual regions, although there are many orientations that are present to different extents in graphic blocks). The slight variability in the ranking of clustering performance using

numerous internal and external clustering measures together with ED or MD can be explained by the specificity of each clustering accuracy measure. For instance, the information and probabilistic theoretical measures compare the distribution of samples in the clustering result and ground truth by computing the variation in mutual information.

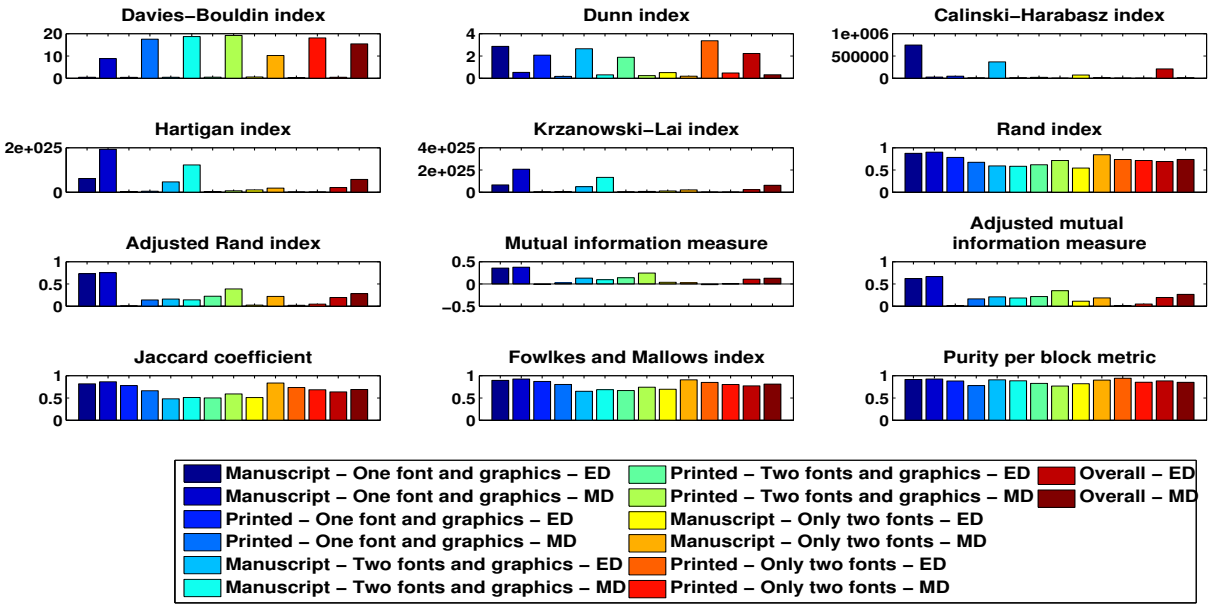


Fig. 18: Evaluation of the proposed pixel labeling approach to digitized historical book content by internal and external clustering accuracy measures performed with the Euclidean distance (ED) and the Mahalanobis distance (MD) in the pixel labeling task. 12 clustering evaluation indices are used: five internal (DaviesBouldin index, Dunn index, CalinskiHarabasz index, Hartigan index and KrzanowskiLai index) and seven external (Rand index, adjusted Rand index, mutual information measure, adjusted mutual information measure, Jaccard coefficient, Fowlkes-Mallows index and purity per block measure). The higher the values, the better the results (except the Davies-Bouldin, where lower values are better).

- **Classification accuracy metrics**

To ensure that each pixel is classified correctly and to provide additional insight into the classification accuracy, the confusion matrix for each document category is used to compute five measures of classification accuracy: PT, E, P, R and CA. To evaluate documents containing two fonts and graphics (Figures 17(c) and 17(d)) using clustering and classification accuracy metrics, all fonts in the text have the same label in the ground truth. Tables and Figure 5 shows the results of these five classification accuracy measures obtained using the ED and MD in the pixel labeling task.

We conclude from Tables and Figure 5 that the results obtained by the numerous computed clustering evaluation measures are coherent with the different classification accuracy results since a considerable improvement in pixel labeling is obtained when using the MD, with overall gains of $6\%(P)$, $10\%(R)$ and $14\%(CA)$. However, we observe slight drops in the average of $1\%(PT)$ and $0,5\%(E)$. This can be explained by the particular inconsistency of the two classification accuracy metrics, which can not indicate precisely the level of accuracy of the results. The best classification for manuscripts containing one font and graphics, especially when using the MD in pixel labeling, is a gain of $1\%(PT)$, $0,5\%(E)$, $21\%(P)$, $20\%(R)$ and $20\%(CA)$. However, relatively low classification accuracy metrics ($58\%(P)$, $51\%(R)$ and $75\%(CA)$) are seen for the printed document category (one font and graphics). This low values are unexpected for this category since we have demonstrated a PPB of 77% (Table 3) without taking into account the topographical relationships. This may raise questions about the defined ground truth, which is to a certain extent subjective. Figure 19 indicates the difference between the ground truth (Figure 19(e)) and the clustering results (Figure 19(c,d)). The ground truth is defined by considering the drop caps as graphic regions while the small letters at the beginning of each text line are considered as text regions (Figure 19(e)). Nevertheless, the results of pixel labeling show that the textural characteristics of each small letter at the beginning of each text line is different from the other text content (Figure 19(c,d)). Thus, to deal with this classic problem, it might be possible to

refine the definition of the ground truth by capturing the subjective average of many users' impressions of their given ground truth. Our previous conclusions on the difficulty of the extracted textural attributes to separate two or more text fonts (Figures 12(d), 12(e), 12(g) and 17(f)), are demonstrated by computing quantitative clustering accuracy, including external and internal measures. Moreover, calculating the classification accuracy metrics confirms that the extracted textural indices can not discriminate between two different fonts in particular, italic and normal fonts (Figure 17(f)). Nevertheless, a slight improvement is observed for classification accuracy measures with manuscripts containing only two fonts characterized by different sizes (Figure 17(e)). This confirms our assumption that the autocorrelation features mainly provide the major orientation of the information, *i.e.* the main orientation of the italic font is different from the uppercase one. However, satisfying results are obtained for printed documents (two fonts and graphics) (Figure 17(d)). Figure 17(d) shows the good results obtained for the segmentation of different kinds of information in the content of printed documents containing graphics (red) and two different fonts: italics (blue) and uppercase (green) fonts. Figure 17(c) shows that it is not possible to distinguish two different fonts characterized by different sizes, although the proposed framework separates the graphic (blue), noise (red) and text (green) regions. The high values for mean P, mean R and mean CA for the printed documents (two fonts and graphics) indicate that our pixel labeling framework tends to misclassify fewer pixels than for manuscripts (two fonts and graphics) and indicates that the quality of segmentation and classification depends on the characteristic information content of the analyzed documents. We obtain 73%(P), 72%(R) and 75%(CA) for manuscripts (two fonts and graphics) and 83%(P), 82%(R) and 82%(CA) for printed documents. This confirms our assumption that the manuscripts contain graphic regions that are more compact and homogeneous than the printed documents. The overall results are quite encouraging since we obtain 70%(P), 70%(R) and 79%(CA) for a large variety of ancient books that have many of the particularities of historical documents. These results are based on the extracted texture features, without taking into account the topographical or spatial relationships and with no hypothesis concerning the document layout or the typographical parameters of the document. High values are obtained for the classification accuracy metrics (75%(P), 78%(R) and 82%(CA)) with the manuscript category compared to printed documents, *i.e.* a difference of 7%(PT), 0,30%(E), 11%(P), 17% (R) and 6%(CA). This can be justified by the fact that manuscripts are characterized by a particular style which generates structured textural features, *i.e.* manuscripts contain drawing regions that are more compact and homogeneous than the printed documents.

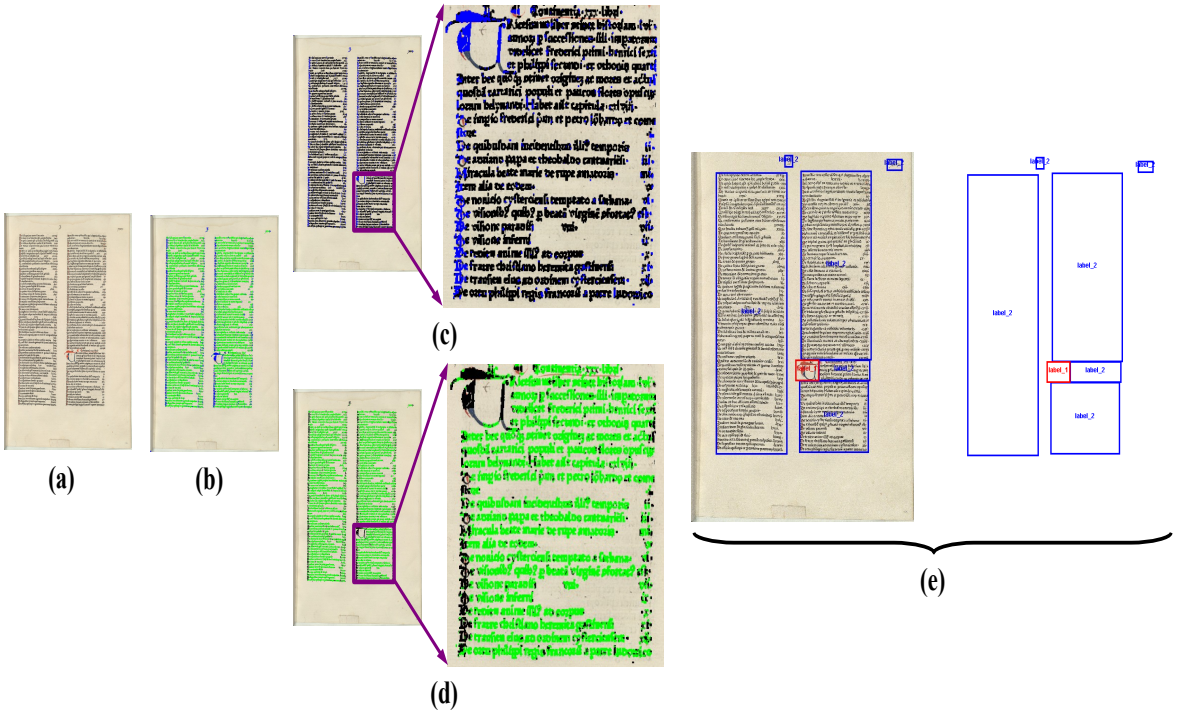
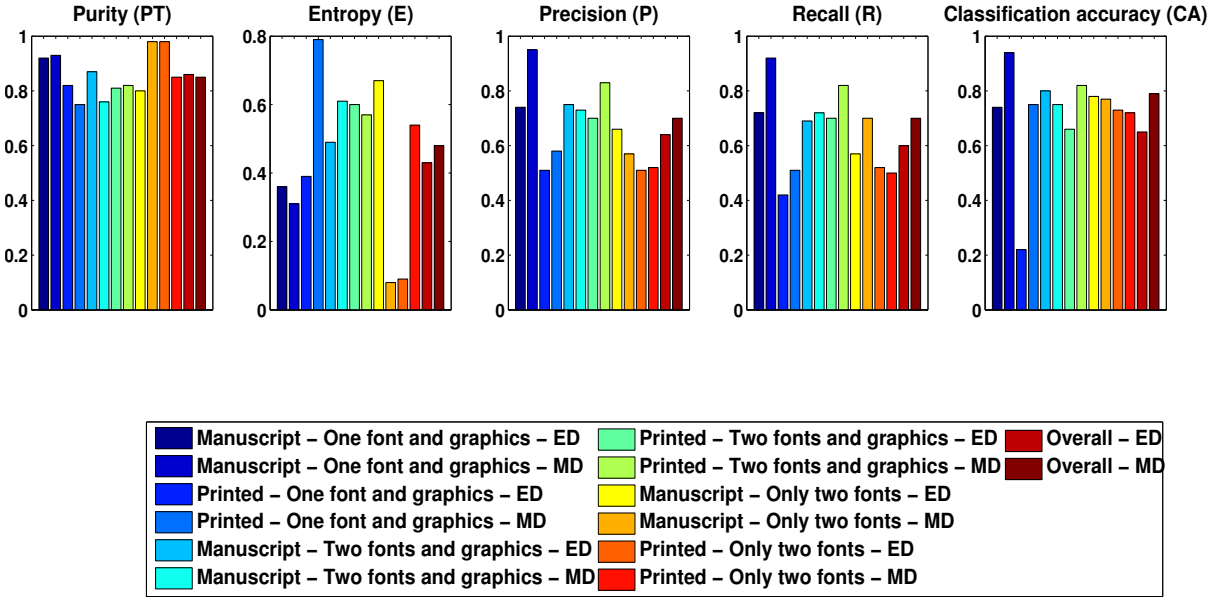


Fig. 19: Example of a segmentation result: **(a)** original grayscale image, **(b)** final result of clustering, **(c)** cluster representing the graphics, **(d)** cluster representing the text, and **(e)** ground truth.

Table 5: Results of the classification accuracy metrics for the proposed framework performed with the Euclidean distance (ED) and the Mahalanobis distance (MD) in the pixel labeling task: purity (PT), entropy (E), precision (P), recall (R) and classification accuracy (CA). $\mu(.)$ and $\sigma(.)$ are the mean and standard deviation of $(.)$, respectively. The higher the values, the better the results (except E, where lower values are better). For documents containing two fonts and graphics (Figures 17(c) and 17(d)), all fonts in the text have the same label in the ground truth.

Euclidean distance	Document category	Document content	$\mu(PT)$	$\sigma(PT)$	$\mu(E)$	$\sigma(E)$	$\mu(P)$	$\sigma(P)$	$\mu(R)$	$\sigma(R)$	$\mu(CA)$	$\sigma(CA)$
	Manuscript	One font and graphics	0,92	0,02	0,36	0,11	0,74	0,38	0,72	0,38	0,74	0,39
		Two fonts and graphics	0,87	0,07	0,49	0,16	0,75	0,19	0,69	0,16	0,80	0,10
		Only two fonts	0,80	0,07	0,67	0,16	0,66	0,26	0,57	0,10	0,78	0,09
		Overall	**0,86**	**0,05**	**0,51**	**0,14**	**0,72**	**0,28**	**0,66**	**0,21**	**0,77**	**0,19**
	Printed	One font and graphics	0,82	0,24	0,39	0,53	0,51	0,07	0,42	0,06	0,22	0,27
		Two fonts and graphics	0,81	0,07	0,60	0,09	0,70	0,17	0,70	0,18	0,66	0,20
		Only two fonts	0,98	0,01	0,09	0,05	0,51	0,03	0,52	0,03	0,73	0,18
		Overall	**0,87**	**0,11**	**0,36**	**0,22**	**0,57**	**0,09**	**0,55**	**0,09**	**0,54**	**0,22**
	Overall		**0,86**	**0,08**	**0,43**	**0,18**	**0,64**	**0,18**	**0,60**	**0,15**	**0,65**	**0,20**

Mahalanobis distance	Document category	Document content	$\mu(PT)$	$\sigma(PT)$	$\mu(E)$	$\sigma(E)$	$\mu(P)$	$\sigma(P)$	$\mu(R)$	$\sigma(R)$	$\mu(CA)$	$\sigma(CA)$
	Manuscript	One font and graphics	0,93	0,01	0,31	0,06	0,95	0,02	0,92	0,02	0,94	0,01
		Two fonts and graphics	0,76	0,18	0,61	0,34	0,73	0,15	0,72	0,14	0,75	0,17
		Only two fonts	0,98	0,01	0,08	0,08	0,57	0,20	0,70	0,30	0,77	0,39
		Overall	**0,89**	**0,07**	**0,33**	**0,16**	**0,75**	**0,12**	**0,78**	**0,15**	**0,82**	**0,19**
	Printed	One font and graphics	0,75	0,05	0,79	0,07	0,58	0,24	0,51	0,02	0,75	0,06
		Two fonts and graphics	0,82	0,05	0,57	0,13	0,83	0,03	0,82	0,06	0,82	0,05
		Only two fonts	0,85	0,07	0,54	0,17	0,52	0,13	0,50	0,13	0,72	0,14
		Overall	**0,81**	**0,06**	**0,63**	**0,12**	**0,64**	**0,13**	**0,61**	**0,07**	**0,76**	**0,08**
	Overall		**0,85**	**0,06**	**0,48**	**0,14**	**0,70**	**0,12**	**0,70**	**0,13**	**0,79**	**0,13**



5.3 Discussion

We have demonstrated both qualitatively and quantitatively the effectiveness of the extracted texture features in the discrimination of the foreground layers of an ancient document image, particularly of text and graphics. Further work needs to be done to introduce other texture descriptors to discriminate between text in a variety of fonts and scales.

The techniques and parameters used in our framework, *i.e.* the clustering method, the standard non-parametric binarization method used to retrieve only pixels representing the information in the foreground, the sizes of the sliding windows for the multiscale approach, the number of selected pixels introduced as input in the estimation of the number of homogeneous regions regions, the distance used in the NNS technique, are selected based on work published in the literature and after performing several experiments to choose the best configuration of the

different techniques in the proposed framework. Moreover, a constructive compromise between computation time and pixel labeling quality is respected. Nevertheless, since the sliding window sizes are set at fixed pixel values and the scanning resolutions often varies considerably (*e.g.* from 72 dpi to 600 dpi), further work is needed to see how different dpi values would affect our framework.

Concerning the texture feature extraction task based on a multiscale analysis, a feature selection step is often required to select relevant features and remove redundant ones. For example, Tao *et al.* [162] introduced new manifold learning based subspace learning algorithm, called Discriminative Locality Alignment (DLA), to extract the discriminative information for similar handwritten Chinese character recognition. Later, they presented a novel feature selection based on the dimension reduction technique, called Sparse Discriminative Information Preservation (SDIP) for Chinese character font categorization, after applying the LBP operator [163]. Wei *et al.* [164] proposed a novel hybrid feature selection method for historical DIA by using adapted greedy forward selection and genetic selection in a cascading way. They concluded that the proposed feature selection method selected significantly less features and lower error rates (*i.e.* 7.97% of mean error rate was noted) were obtained than in the case of using all features. In addition, they noted that some texture features (*e.g.* gradient, Laplacian, and LBP) were frequently selected.

Concerning the overall results obtained with the proposed framework, 85%(*PPB*), 79%(*CA*), 70%(*P*) and 70%(*R*) are noted with a low processing time and memory complexity. The proposed framework has the advantage that it is performed in the absence of a hypothesis concerning the document layout (physical structure) or the typographical parameters of the document (logical structure). In this framework, the number of homogeneous regions is determined automatically using the CC technique on randomly selected pixels from ten book pages without taking into account the spatial attributes. This approach is based on book page analysis using the CC and NNS techniques to find similarities between the textural characteristics of their contents. In our framework, there is no post-processing of the segmented documents. The results will be improved if a new task is introduced for the use of spatial relationships among the selected pixels.

6 Conclusions and further work

This article proposes a generic framework for a texture-based pixel labeling framework of digitized historical book content with no hypothesis concerning the document layout or the typographical parameters of the document. The aim of this framework is to find homogeneous regions within the content of digitized historical books by extracting and analyzing texture features independently of the layout of the pages. It is therefore applicable to a large variety of books. Our framework is based on a feature vector that is composed of texture indices, all based on the autocorrelation function and the rose of directions. The texture features are extracted from the different areas of a page and at several resolutions. The robustness of the extracted features is used in a parameter-free unsupervised clustering method which is designed to determine the homogeneous regions (*i.e.* defined by similar autocorrelation indices). Moreover, the number of homogeneous regions does not need to be known in advance as it is determined automatically.

This framework has been evaluated on 316 pages of historical documents. We conclude that the autocorrelation features provide a good discrimination of the foreground layers of a document, particularly between text and graphics. The results show that the autocorrelation descriptors can distinguish textual from graphic regions of an analyzed document. 85% purity per block accuracy and 79% classification accuracy are obtained. However, it is possible to speculate that if we integrate several kinds of post-processing techniques, we will have better results than those reported in this article. It is to be noted that we do not assume knowledge about the font size, scanning resolution, column layout, orientation, *etc.* of the analyzed document.

The first aspect of future work will be to use the proposed framework on a larger database. This study is ongoing and will evaluate the framework more adequately, with more convincing experimental results. We will then study and combine statistical, geometric, model-based and other frequency texture features in order to refine the segmentation and ensure a distinction between different text fonts and various graphic types. Moreover, our results will be improved if we include topographical or spatial relationships and/or fuzzy clustering methods in our clustering approach. Furthermore, by integrating a new processing stage after pixel labeling, which consists of pixel grouping that takes into consideration the topographical relationships of pixels and their labels, e.g. some operators from mathematical morphology, the classification results should be improved. This will be studied in the near future.

There are several possibilities that stem from this work. In particular, this type of analysis can be used to design a computer-aided page categorization tool and provide a similarity measure for pages defined on the basis of a combination of a representation of homogenous regions and their topology.

Acknowledgements The support of this research by the ANR (French National Research Agency) under contract $ANR-10-CORD-0020$ is gratefully acknowledged. The authors would like also to thank Geneviève CRON of the BnF for providing access to the Gallica digital library.

References

1. J. André and M. A. Chabin, "Les documents anciens," *Document Numérique*, 1999.
2. F. LeBourgeois, E. Trinh, B. Allier, V. Eglin, and H. Emptoz, "Document images analysis solutions for digital libraries," in *International Workshop on Document Image Analysis for Libraries*. IEEE, 2004, pp. 2–24.
3. F. LeBourgeois and H. Emptoz, "DEBORA: Digital AccEss to BOoks of the RenAissance," *International Journal of Document Analysis and Recognition*, pp. 193–221, 2007.
4. M. Baechler, A. Fischer, N. Naji, R. Ingold, H. Bunke, and J. Savoy, "HisDoc: Historical document analysis, recognition, and retrieval," in *Digital Humanities - International Conference of the Alliance of Digital Humanities Organizations (ADHO)*, 2012.
5. J. M. Ogier and K. Tombre, "Madonne: Document image analysis techniques for cultural heritage documents," in *International Conference on Digital Cultural Heritage*, 2006.
6. T. M. Rath and R. Manmatha, "Word spotting for historical documents," *International Journal of Document Analysis and Recognition*, pp. 139–152, 2007.
7. H. S. Baird, "Digital libraries and document image analysis," in *International Conference on Document Analysis and Recognition*. IEEE, 2003, pp. 2–14.
8. J. M. Ogier, "Ancient document analysis: A set of new research problems," in *Colloque International Francophone sur l'Ecrit et le Document*, 2005.
9. M. Coustaty, R. Raveaux, and J. M. Ogier, "Historical document analysis: A review of French projects and open issues," in *European Signal Processing Conference*. EURASIP, 2011, pp. 1445–1449.
10. O. Okun and M. Pietikäinen, "A survey of texture-based methods for document layout analysis," in *Workshop on Texture Analysis in Machine Vision*. Springer-Verlag, 1999, pp. 137–148.
11. A. Piper, "Reading's refrain: From bibliography to topology," *Readings: Selected Essays from the English Institute*, pp. 373–399, 2013.
12. E. T. Nalisnick and H. S. Baird, "Extracting sentiment networks from Shakespeare's plays," in *International Conference on Document Analysis and Recognition*. IEEE, 2013, pp. 758–762.
13. G. Agam, G. Bal, G. Frieder, and O. Frieder, "Degraded document image enhancement," in *Document Recognition and Retrieval*. SPIE, 2007.
14. L. Likforman-Sulem, "Apport du traitement des images à la numérisation des documents anciens," *Document Numérique*, pp. 13–26, 2003.
15. J. André, H. Richy, L. Likforman-Sulem, and G. Ventabert, "Electronic representation and use of old documents (texts and images): About Philectre project experiments," *Document Numérique*, pp. 57–73, 1999.
16. L. Likforman-Sulem, A. Zahour, and B. Taconet, "Text line segmentation of historical documents: A survey," *International Journal of Document Analysis and Recognition*, pp. 123–138, 2007.
17. G. Nagy and S. Seth, "Hierarchical representation of optically scanned documents," in *International Conference on Pattern Recognition*. IEEE, 1984, pp. 347–349.
18. F. M. Wahl, K. Y. Wong, and R. G. Casey, "Block segmentation and text extraction in mixed text/image documents," *Computer Graphics and Image Processing*, pp. 375–390, 1982.
19. Y. P. Zhou and C. L. Tan, "Hough technique for bar charts detection and recognition in document images," in *International Conference on Image Processing*. IEEE, 2000, pp. 605–608.
20. A. Belaïd and N. Ouwayed, *Guide to OCR for Arabic scripts: Segmentation of ancient Arabic documents*. Springer, 2011.
21. N. Nikolaou, M. Makridis, B. Gatos, N. Stamatopoulos, and N. Papamarkos, "Segmentation of historical machine-printed documents using adaptive run-length smoothing and skeleton segmentation paths," *Image and Vision Computing*, pp. 590–604, 2010.
22. J. Serra, *Image analysis and mathematical morphology*. Academic Press, 1982.
23. I. Granado, M. Mengucci, and F. Muge, "Extraction de textes et de figures dans les livres anciens à l'aide de la morphologie mathématique," in *Colloque International Francophone sur l'Ecrit et le Document*, 2000.
24. F. Muge, I. Granado, M. Mengucci, P. Pina, V. Ramos, N. Sirakov, J. R. C. Pinto, A. Marcolino, M. Ramalho, P. Vieira, and A. M. d. Amaral, "Automatic feature extraction and recognition for digital access of books of the Renaissance," in *Research and Advanced Technology for Digital Libraries, Lecture Notes in Computer Science*. Springer-Verlag, 2000, pp. 1–13.
25. M. Mengucci and I. Granado, "Morphological segmentation of text and figures in renaissance books (XVI century)," *Mathematical Morphology and its Applications to Image and Signal Processing Computational Imaging and Vision*, pp. 397–404, 2002.
26. J. Y. Ramel, S. Leriche, M. L. Demonet, and S. Busson, "User-driven page layout analysis of historical printed books," *International Journal of Document Analysis and Recognition*, pp. 243–261, 2007.
27. A. Crasson and J. D. Fekete, "Structuration des manuscrits : du corpus à la région," in *Colloque International Francophone sur l'Ecrit et le Document*, 2004.
28. K. Kise, *Page segmentation techniques in document analysis*. Handbook of Document Image Processing and Recognition, Springer-Verlag, 2014.
29. B. Julesz, "Visual pattern discrimination," *Information Theory*, pp. 84–92, 1962.
30. N. Chen and D. Blostein, "A survey of document image classification: Problem statement, classifier architecture and performance evaluation," *International Journal of Document Analysis and Recognition*, pp. 1–16, 2007.
31. N. Journet, J. Ramel, R. Mullot, and V. Eglin, "Document image characterization using a multiresolution analysis of the texture: Application to old documents," *International Journal of Document Analysis and Recognition*, pp. 9–18, 2008.

32. M. Mehri, P. Héroux, P. Gomez-Krämer, and R. Mullot, "A pixel labeling approach for historical digitized books," in *International Conference on Document Analysis and Recognition*. IEEE, 2013, pp. 817–821.
33. R. Cohen, A. Asi, K. Kedem, J. El-Sana, and I. Dinstein, "Robust text and drawing segmentation algorithm for historical documents," in *International Workshop on Historical Document Imaging and Processing*. ACM, 2013, pp. 110–117.
34. H. P. Lai, M. Visani, A. Boucher, and J. M. Ogier, "An experimental comparison of clustering methods for content-based indexing of large image databases," *Pattern Analysis and Applications*, pp. 345–366, 2012.
35. B. Allier, J. Duong, A. Gagneux, P. Mallet, and H. Emptoz, "Texture feature characterization for logical pre-labeling," in *International Conference on Document Analysis and Recognition*. IEEE, 2003, pp. 567–571.
36. A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *Pattern Analysis and Machine Intelligence*, pp. 4–37, 2000.
37. Y. Liua, S. Wub, and X. Zhoua, "Texture segmentation based on features in wavelet domain for image retrieval," pp. 2026–2034, 2003.
38. A. K. Jain, S. K. Bkattacharjee, and Y. Chen, "On texture in document images," in *Computer Vision and Pattern Recognition*. IEEE, 1992, pp. 677–680.
39. C. H. Chen, L. F. Pau, and P. Wang, *Texture analysis in the handbook of pattern recognition and computer vision*, 2nd ed. World Scientific, 1998.
40. M. Tuceryan and A. K. Jain, *Texture analysis*. The Handbook of Pattern Recognition and Computer Vision (2nd Edition), by C. H. Chen, L. F. Pau, P. S. P. Wang (eds.), World Scientific Publishing Co, 1998.
41. R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *Systems Man and Cybernetics*, pp. 610–621, 1973.
42. M. Tuceryan and A. K. Jain, "Texture segmentation using Voronoi polygons," *Pattern Analysis and Machine Intelligence*, pp. 211–216, 1990.
43. J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data," in *International Conference on Machine Learning*, 2001, pp. 282–289.
44. S. Nicolas, Y. Kessentini, T. Paquet, and L. Heutte, "Handwritten document segmentation using hidden Markov random fields," in *International Conference on Document Analysis and Recognition*. IEEE, 2005, pp. 212–216.
45. R. Chellappa and S. Chatterjee, "Classification of textures using Markov random field models," in *International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 1984, pp. 694–697.
46. R. Ferrell, S. Gleason, and K. Tobin, "Application of fractal encoding techniques for image segmentation," in *International Conference on Quality Control by Artificial Vision*. SPIE, 2003, pp. 69–77.
47. T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence*, pp. 971–987, 2002.
48. A. K. Jain and S. Bhattacharjee, "Text segmentation using Gabor filters for automatic document processing," *Machine Vision and Applications*, pp. 169–184, 1992.
49. C. Sabharwal and S. Subramanya, "Indexing image databases using wavelet and discrete Fourier transform," in *Symposium on Applied Computing*. ACM, 2001, pp. 434–439.
50. S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *Pattern Analysis and Machine Intelligence*, pp. 674–693, 1989.
51. M. Tuceryan, "Moment based texture segmentation," *Pattern Recognition Letters*, pp. 659–668, 1994.
52. S. Uttama, P. Loonis, M. Delalandre, and J. M. Ogier, "Segmentation and retrieval of ancient graphic documents," in *International Workshop on Graphics Recognition on Graphics Recognition (GREC): Ten Years Review and Future Perspectives*. Springer-Verlag, 2006, pp. 88–98.
53. M. Mehri, P. Gomez-Krämer, P. Héroux, and R. Mullot, "Old document image segmentation using the autocorrelation function and multiresolution analysis," in *Document Recognition and Retrieval*. SPIE, 2013.
54. R. M. Haralick, "Statistical and structural approaches to texture," *In Proceedings of the IEEE*, pp. 786–804, 1979.
55. M. Petrou and P. G. Sevilla, *Image processing: Dealing with texture*. John Wiley & Sons, 2006.
56. V. Eglin, S. Bres, and C. Rivero, "Hermite and Gabor transforms for noise reduction and handwriting classification in ancient manuscripts," *International Journal of Document Analysis and Recognition*, pp. 101–122, 2007.
57. A. Garz and R. Sablatnig, "Multi-scale texture-based text recognition in ancient manuscripts," in *International Conference on Virtual Systems and Multimedia*. IEEE, 2010, pp. 336–339.
58. C. Grana, D. Borghesani, and R. Cucchiara, "Automatic segmentation of digitalized historical manuscripts," *Multimedia Tools and Applications*, pp. 483–506, 2011.
59. A. Ouji, Y. Leydier, and F. LeBourgeois, "Chromatic / achromatic separation in noisy document images," in *International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 167–171.
60. S. Bres, "Contributions à la quantification des critères de transparence et d'anisotropie par une approche globale : Application au contrôle de qualité de matériaux composites," Ph.D. dissertation, Institut National des Sciences Appliquées de Lyon, Lyon, France, 1994.
61. M. Mehri, P. Gomez-Krämer, P. Héroux, A. Boucher, and R. Mullot, "Texture feature evaluation for segmentation of historical document images," in *International Workshop on Historical Document Imaging and Processing*. ACM, 2013, pp. 102–109.
62. ——, "A pixel labeling framework for comparing texture features: Application to digitized ancient books," in *International Conference on Pattern Recognition Applications and Methods*. SciTePress, 2014, pp. 553–560.
63. G. Peake and T. Tan, "Script and language identification from document images," in *Document Image Analysis*. IEEE, 1997, pp. 10–17.
64. A. Busch, W. W. Boles, and S. Sridharan, "Texture for script identification," *Pattern Analysis and Machine Intelligence*, pp. 1720–1732, 2005.
65. Y. Zhu, T. Tan, and Y. Wang, "Font recognition based on global texture analysis," *Pattern Analysis and Machine Intelligence*, pp. 1192–1200, 2001.
66. H. Ma and D. Doermann, "Gabor filter based multi-class classifier for scanned document images," in *International Conference on Document Analysis and Recognition*. IEEE, 2003, pp. 968–972.
67. A. K. Jain and Y. Zhong, "Page segmentation using texture analysis," *Pattern Recognition*, pp. 743–770, 1996.
68. T. Randen and J. H. Husøy, "Segmentation of text/image documents using texture approaches," 1994.
69. J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: the Fuzzy C-Means clustering algorithm," in *Computers & Geosciences*. Pergamon Press, 1984, pp. 191–203.

70. F. Kovács, C. Legány, and A. Babos, "Cluster validity measurement techniques," in *International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases*. World Scientific and Engineering Academy and Society, 2006, pp. 388–393.
71. J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press, 1967, pp. 281–297.
72. L. Kaufman and P. J. Rousseeuw, *Finding groups in data: An introduction to cluster analysis*. John Wiley & Sons, 1990.
73. G. N. Lance and W. T. Williams, "A general theory of classificatory sorting strategies 1. Hierarchical systems," *The Computer Journal*, pp. 373–380, 1967.
74. M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *International Conference on Knowledge Discovery and Data Mining*. AAAI Press, 1996, pp. 226–231.
75. M. Ankerst, M. M. Breunig, H. P. Kriegel, and J. Sander, "OPTICS: Ordering Points To Identify the Clustering Structure," in *International Conference on Management of Data*. ACM Press, 1999, pp. 49–60.
76. G. J. McLachlan and T. Krishnan, *The EM algorithm and extensions*. John Wiley & Sons, 1997.
77. W. Wang, J. Yang, and R. Muntz, "STING: A statistical information grid approach to spatial data mining," in *International Conference on Very Large Data*. Morgan Kaufmann, 1997, pp. 186–195.
78. G. Sheikholeslami, S. Chatterjee, and A. Zhang, "WaveCluster: A multi-eesolution clustering approach for very large spatial databases," in *International Conference on Very Large Data*. Morgan Kaufmann, 1998, pp. 428–439.
79. E. Smigiel, A. Belaïd, and H. Hamza, "Self-organizing maps and ancient documents," in *International Workshop on Document Analysis Systems*. Springer-Verlag, 2004, pp. 125–134.
80. J. F. Rosenblatt, *Principles of neurodynamics*. Spartan Books, 1962.
81. R. Xu, "Survey of clustering algorithms," *Neural Networks*, pp. 645–678, 2005.
82. J. Cocquerez and S. Philipp, *Analyse d'images : filtrage et segmentation*. Masson, 1995.
83. R. Duda, P. Hart, and D. Stork, *Pattern classification*. 2nd Edition Wiley-Interscience, 2001.
84. M. Cord and P. Cunningham, *Machine learning techniques for multimedia case studies on organization and retrieval, series: cognitive technologies*. Springer-Verlag, 2008.
85. A. Cornuéjols and L. Miclet, *Apprentissage artificiel : concepts et algorithmes*. 2nd Edition Eyrolles, 2010.
86. N. Iam-on and S. Garrett, "LinkCluE: A Matlab package for link-based cluster ensembles," *Journal of Statistical Software*, pp. 1–36, 2010.
87. S. Ray and R. H. Turi, "Determination of number of clusters in k-means clustering and application in color image segmentation," in *International Conference on Advances in Pattern Recognition and Digital Techniques*. Narosa Publishing House, 1999, pp. 137–143.
88. H. A. Moesa, D. B. K.C., and T. Akutsu, "Efficient determination of cluster boundaries for analysis of gene expression profile data using hierarchical clustering and wavelet transform," *Genome Informatics*, pp. 132–141, 2005.
89. P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, pp. 53–65, 1987.
90. R. Lletía, M. C. Ortiza, L. A. Sarabiab, and M. S. Sánchez, "Selecting variables for k-means cluster analysis by using a genetic algorithm that optimises the silhouettes," in *Colloquim Chemiometricum Mediterraneum*. Elsevier Science, Analytica Chimica Acta, 2004, pp. 87–100.
91. StatSoft. (2010) Finding the right number of clusters in k-means and EM clustering: v-Fold Cross-Validation. Electronic Statistics Textbook. [Online]. Available: http://www.statsoft.com/textbook/cluster-analysis/
92. Q. Zhao, M. Xu, and P. Fränti, "Extending external validity measures for determining the number of clusters," in *International Conference on Intelligent Systems Design and Applications*. IEEE, 2011, pp. 931–936.
93. K. Kryszczuk and P. Hurley, "Estimation of the number of clusters using multiple clustering validity indices," in *International Conference on Multiple Classifier Systems*. Springer-Verlag, 2010, pp. 114–123.
94. N. Bolshakova and F. Azuaje, "Estimating the number of clusters in DNA microarray data," *Methods of information in medicine*, pp. 153–157, 2006.
95. J. Yu, D. Tao, J. Li, and J. Cheng, "Semantic preserving distance metric learning and applications," *Information Sciences, Multimedia Modeling*, pp. 674–686, 2014.
96. M. Wang, B. Ni, X. S. Hua, and T. S. Chua, "Assistive tagging: A survey of multimedia tagging with human-computer joint exploration," *Computing Surveys*, pp. 1–25, 2012.
97. J. Yu, R. Hong, M. Wang, and J. You, "Image clustering based on sparse patch alignment framework," *Pattern Recognition*, pp. 3512–3519, 2014.
98. J. Yu, Y. Rui, Y. Y. Tang, and D. Tao, "High-order distance-based multiview stochastic learning in image classification," *Transactions on Cybernetics*, pp. 1–12, 2014.
99. M. Cote and A. B. Albu, "Texture sparseness for pixel classification of business document images," *International Journal of Document Analysis and Recognition*, pp. 1–17, 2014.
100. M. Mehri, V. C. Kieu, M. Mhiri, P. Héroux, P. Gomez-Krämer, M. A. Mahjoub, and R. Mullot, "Robustness assessment of texture features for the segmentation of ancient documents," in *International Workshop on Document Analysis Systems*. IEEE, 2014, pp. 293–297.
101. N. Otsu, "A threshold selection method from gray-level histograms," *Systems, Man, and Cybernetics*, pp. 62–66, 1979.
102. L. Shijian and C. L. Tan, "Script and language identification in noisy and degraded document images," *Pattern Analysis and Machine Intelligence*, pp. 14–24, 2008.
103. J. He, Q. D. M. Do, A. C. Downton, and J. H. Kim, "A comparison of binarization methods for historical archive documents," in *International Conference on Document Analysis and Recognition*. IEEE, 2005, pp. 538–542.
104. A. G. Lasmar, A. Kricha, and N. E. B. Amara, "A segmentation text/background method for degraded ancient Arabic manuscript," in *International Conference on Information & Communication Technologies*. IEEE, 2006, pp. 1327–1331.
105. J. Li, J. Z. Wang, and G. Wiederhold, "Classification of textured and non-textured images using region segmentation," *Image Processing*, pp. 754–757, 2000.
106. L. Cinque, L. Lombardi, and G. Manzini, "A multiresolution approach for page segmentation," *Pattern Recognition Letters*, pp. 217–225, 1998.
107. C. Tan and P. Ng, "Text extraction using pyramid," *Pattern Recognition*, pp. 63–72, 1998.
108. C. Tan and Z. Zhang, "Text block segmentation using pyramid structure," in *Document Recognition and Retrieval*. SPIE, 2000, pp. 297–306.
109. A. Lemaitre, J. Camillerapp, and B.Coüasnon, "Multiresolution cooperation improves document structure recognition," *International Journal of Document Analysis and Recognition*, pp. 97–109, 2008.

110. H. Greenspan, "Multi-resolution image processing and learning for texture recognition and image enhancement," Ph.D. dissertation, California Institute of Technology, 1994.
111. S. Contassot-Vivier, G. L. Bosco, and N. C. Dao, "Multiresolution approach for image processing," in *Erasmus ICP-A-2007*, 1996.
112. A. Kricha and N. E. B. Amara, "Exploring textural analysis for historical documents characterization," *Journal of computing*, pp. 24–30, 2011.
113. D. J. Ketchen and C. L. Shook, "The application of cluster analysis in strategic management research: An analysis and critique," *Strategic Management Journal*, pp. 441–458, 1996.
114. T. Simpson, J. Armstrong, and A. Jarman, "Merged consensus clustering to assess and improve class discovery with microarray data," *Boston Medical Center Bioinformatics*, pp. 1471–1482, 2010.
115. S. Monti, P. Tamayo, J. Mesirov, and T. Golub, "Consensus Clustering: A resampling-based method for class discovery and visualization of gene expression microarray data," *Machine Learning*, pp. 91–118, 2003.
116. G. Nguyen, M. Coustaty, and J. M. Ogier, "Stroke feature extraction for lettrine indexing," in *International Conference on Image Processing Theory Tools and Applications*. IEEE, 2010, pp. 355–360.
117. J. Ward, "Hierarchical grouping to optimize an objective function," *Journal of the American Statistical Association*, pp. 236–244, 1963.
118. F. Lalys, C. Haegelen, M. Mehri, S. Drapier, M. Vérin, and P. Jannin, "Anatomo-clinical atlases correlate clinical data and electrode contact coordinates : Application to subthalamic deep brain stimulation," *Journal of Neuroscience*, pp. 297–307, 2013.
119. D. E. Knuth, *The art of computer programming, volume 3: (2nd ed.) sorting and searching*. Addison Wesley Longman Publishing Co, 1997.
120. P. Mahalanobis, "On the generalised distance in statistics," in *Proceedings of the National Institute of Sciences of India*. NISI, 1936, pp. 49–55.
121. D. Doermann, E. Zotkina, and H. Li, "GEDI - A Groundtruthing Environment for Document Images," in *International Workshop on Document Analysis Systems*. ACM, 2010.
122. F. Ge, S.Wang, and T. Liu, "New benchmark for image segmentation evaluation," *Journal of Electronic Imaging*, pp. 1–16, 2007.
123. H. Zhang, J. Fritts, and S. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," *Computer Vision and Image Understanding*, pp. 260–280, 2008.
124. S. Wontaek, M. Agrawal, and D. Doermann, "Performance Evaluation Tools for zone Segmentation and classification (PETS)," in *International Conference on Pattern Recognition*. IEEE, 2010, pp. 503–506.
125. E. Rendón, I. Abundez, A. Arizmendi, and E. M. Quiroz, "Internal versus external cluster validation indexes," *International Journal of Computers and Communications*, pp. 27–34, 2011.
126. E. Rendón, I. Abundez, C. Gutierrez, S. D. Zagal, A. Arizmendi, E. M. Quiroz, and H. E. Arzate, "A comparison of internal and external cluster validation indexes," in *Applications of Mathematics and Computer Engineering (AMERICAN-MATH/CEA 2011)*. World Scientific and Engineering Academy and Society (WSEAS), 2011, pp. 158–163.
127. A. Silva, "Metrics for evaluating performance in document analysis: Application to tables," *International Journal of Document Analysis and Recognition*, pp. 101–109, 2011.
128. J. R. Jensen, *Introductory digital image processing*. Prentice-Hall, Englewood Cliffs, NJ, 1986.
129. P. M. Mather, *Computer processing of remotely-sensed images: An introduction*. 2nd Edition John Wiley & Sons, 1999.
130. J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel, "Performance measures for information extraction," in *DARPA Broadcast News Workshop*. Morgan Kaufmann Publishers, Inc, 1999, pp. 249–252.
131. J. M. Wei, X. J. Yuan, Q. H. Hub, and S. Q. Wangc, "A novel measure for evaluating classifiers," *Expert Systems with Applications*, pp. 3799–3809, 2010.
132. D. M. W. Powers, "Evaluation: From precision, recall and F-factor to ROC, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, pp. 37–63, 2011.
133. B. Liu, *Web data mining: Exploring hyperlinks, contents, and usage data*. Springer-Verlag, 2011.
134. A. K. Santra and C. J. Christy, "Genetic algorithm and confusion matrix for document clustering," *International Journal of Computer Science*, pp. 322–328, 2012.
135. P. C. Saxena and K. Navaneetham, "The effect of cluster size, dimensionality, and number of clusters on recovery of true cluster structure through Chernoff-type faces," *Journal of the Royal Statistical Society, The Statistician*, pp. 415–425, 1991.
136. E. B. Fowlkes and C. L. Mallows, "A method for comparing two hierarchical clusterings," *Journal of the American Statistical Association*, pp. 553–569, 1983.
137. Y. Zhao and G. Karypis, "Criterion functions for document clustering: Experiments and analysis," Department of Computer Science, University of Minnesota, Tech. Rep. Technical report TR 0140, 2001.
138. W. J. Krzanowski and Y. T. Lai, "A criterion for determining the number of groups in a data set using sum-of-squares clustering," *International Biometric Society, JSTOR*, pp. 23–34, 1988.
139. J. A. Hartigan, *Clustering algorithms*. John Wiley & Sons, 1975.
140. R. B. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics*, pp. 1–27, 1974.
141. W. S. Sarle, "The cubic clustering criterion," SAS Institute, Tech. Rep. SAS Technical Report A-108: The Cubic Clustering Criterion, 1983.
142. A. J. Scott and M. J. Symons, "Clustering methods based on likelihood ratio criteria," *Biometrics*, pp. 387–397, 1971.
143. F. H. Marriott, "Practical problems in a method of cluster analysis," *Biometrics*, pp. 501–514, 1971.
144. G. W. Milligan and M. Cooper, "An examination of procedures for determining the number of clusters in a data set," *Psychometrika*, pp. 159–179, 1985.
145. H. P. Friedman and J. Rubin, "On some invariant criteria for grouping data," *Journal of the American Statistical Association*, pp. 1159–1178, 1967.
146. J. Rubin, "Optimal classification into groups: An approach for solving the taxonomy problem," *Journal of Theoretical Biology*, pp. 103–144, 1967.
147. L. J. Hubert and J. R. Levin, "A general statistical framework for assessing categorical clustering in free recall," *Psychological Bulletin*, pp. 1072–1080, 1976.
148. D. L. Davies and D. W. Bouldin, "A cluster separation measure," *Pattern Analysis and Machine Intelligence*, pp. 224–227, 1979.
149. D. A. Ratkowsky and G. N. Lance, "A criterion for determining the number of groups in a classification," *Australian Computer Journal*, pp. 115–117, 1978.

150. G. H. Ball and D. J. Hall, "ISODATA, a novel method of data analysis and pattern classification," Menlo Park: Stanford Research Institute, Tech. Rep. AD0699616, 1965.
151. G. W. Milligan, "An examination of the effect of six types of error perturbation on fifteen clustering algorithms," *Psychometrika*, pp. 325–342, 1980.
152. T. Frey and H. V. Groenewoud, "A cluster analysis of the d-squared matrix of white spruce stands in saskatchewan based on the maximum-minimum principle," *Journal of Ecology*, pp. 873–886, 1972.
153. J. O. McClain and V. R. Rao, "CLUSTISZ: A program to test for the quality of clustering of a set of objects," *Journal of Marketing Research*, pp. 456–460, 1975.
154. J. Dunn, "Well separated clusters and optimal fuzzy partitions," *Journal of Cybernetics*, pp. 95–104, 1974.
155. M. Halkidi, M. Vazirgiannis, and I. Batistakis, "Quality scheme assessment in the clustering process," in *Principles and Practice of Knowledge in databases*. Springer-Verlag, 2000, pp. 265–276.
156. M. Halkidi, I. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *Journal of Intelligent Information Systems*, pp. 107–145, 2001.
157. E. Deza and M. M. Deza, *Encyclopedia of distances*. Springer-Verlag, 2013.
158. W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, pp. 846–850, 1971.
159. L. Hubert and P. Arabic, "Comparing partitions," *Journal of Classification*, pp. 193–218, 1985.
160. A. Kraskov, H. Stögbauer, R. G. Andrzejak, and P. Grassberger, "Hierarchical clustering based on mutual information," in *Quantitative Methods (q-bio.QM)*. CoRR q-bio.QM/0311039, 2003, pp. 193–218.
161. N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance," *Journal of Machine Learning Research*, pp. 2837–2854, 2010.
162. D. Tao, L. Liang, L. Jin, and Y. Gao, "Similar handwritten Chinese character recognition by kernel discriminative locality alignment," *Pattern Recognition Letters*, pp. 186–194, 2014.
163. D. Tao, L. Jin, S. Zhang, Z. Yang, and Y. Wang, "Sparse discriminative information preservation for Chinese character font categorization," *Neurocomputing*, pp. 159–167, 2014.
164. H. Wei, K. Chen, R. Ingold, and M. Liwicki, "Hybrid feature selection for historical document layout analysis," in *International Conference on Frontiers in Handwriting Recognition*. IEEE, 2014, pp. 87–92.