Dedicated texture based tools for characterisation of old books

N. Journet[1] V. Eglin [2] J.Y Ramel[3] R. Mullot[1]

[1]L3I, 17042 La Rochelle Cedex 1 - FRANCE njournet@univ-lr.fr

[2]LIRIS INSA de Lyon, Villeurbanne cedex - FRANCE veronique.eglin@insa-lyon.fr

[3]LI-RFAI, 64 Avenue Jean Portalis 37200 TOURS - FRANCE Jean-yves.ramel@univ-tours.fr

Abstract

This paper lies on the field of ancient patrimonial books valorization: it precisely relates to the development of suitable assistance tools for humanists and historians to help them to retrieve information in great corpus of digitized documents. This paper presents a part of this ambitious project and deals with the presentation of a pixel classification method for ancient typewritten documents. The presented approach lies on a multiresolution maps construction and analysis. For 5 resolutions we construct 5 different characterisation maps. All the maps are based on texture information (correlation of pixels orientations, grey level pixel density…). After the merging of these 25 maps, each pixel of the original image is described by a vector which allows the computing of a hierarchical classification. In order to avoid issues linked to the binarization process, all or maps are computed on grey level images.

The system has been tested on a CESR database of ancient printed books of the Renaissance. The classification results are evaluated through several visual classification illustrations.

1. Introduction

All along the digitalisation campaigns, historians have accumulated many European ancient typewritten documents. The “digital” conservation of old books is not the only interest, the access and the diffusion of this knowledge looks quite important too. The number of virtual libraries [1,2,4] is the proof of the growing interest from historians and scientists for those campaigns.

We are currently working in collaboration with the “Centre d’Etudes Supérieures de la Renaissance” in France, who wants to construct a Humanistic Virtual Library and diffuse it through the Web. The problem is that, for the moment, the images of ancient documents are available only in bitmap format. That means that any kinds of information retrieval with ASCII key-words based queries, remain impossible. In this context, we need to develop more powerful indexation tools (based on dedicated OCR, specific layout extraction, document images and graphics retrieval system…) which would allow most interesting accesses to old collections.

This paper deals with one of this aspect of indexation: the layout extraction. Extracting the layout means to be able to segment the image document and determine the label of each segmented part. The main difficulty of layout extraction comes from the particularity of old documents corpus. Books come from different countries (France, Germany, Italy, Holland, etc…) and different centuries (from the middle of the 15th to the end of the 17th). In 3 centuries a lot of things have changed in printed techniques and editing rules. The resulting wide corpus is composed with books which are visually different the one from the others and for which it is difficult to set stable ontologies and editorial rules.

In this context, we present a new method which takes into account those specificities. In the two first parts of the paper, we will present the characteristics of old books collections and some layout extraction techniques used in contemporary books and we will talk about there adaptability to old documents. In the next two following parts we will detail our method based on the extraction of document texture features which take into account the specificity of our corpus (heterogeneity of data, variability of information densities, and variability of page layouts). Lastly, we evaluate the accuracy of our method based on a hierarchical classification.

2. Noticeable ancient printed books characteristics

Characteristics of old documents come from the origin of the book itself (century, technical process…) but also from techniques used in the digitalization and restoration processes.

The old documents digitalization is arduous and must take into account a lot of external factors which contribute to their degradation and their non ideal conservation. From one part, we can notice that the paper support has often suffered from intensive uses all along different historical periods. It is really recent to take so much care of books and even sometimes to forgive their access. Books degradation is visible in different ways: little holes, ink spots, apparition of the verso due to the ink acidity… So as to overcome those damages, digitalization programs have recently been taking into account. Dedicated old documents scanners and restoration processes have been created in order to propose viable solutions ([3], [5]). Typically, they allow resolving problems linked to lake of lighting, distortion, skew, and elimination of ink dots.

For page layouts, the technical and historical constraints impose particular presentations. The variability of page layouts is due to either technical inaccuracies or liberties taken by the printer. There are no exact rules, but most of time the body text part covers the quite totality of page area and generally with some additive marginal notes. The page can also contain graphical parts of various sizes and some ornament patterns. In the text, we can find known structures like the titles and the subtitles, the paragraphs, the page numbers, and other more particular structures like the catchwords. The styles used can alternate, with normal style, justified or aligned on the left. Another characteristic of old printed books comes from weak separations between blocks of text (notes in the margins and body text for example). Lastly, we can notice that on some documents, layout rules are not always respected. For example, an illustration can overflow into the margins (see figure 1).

This study enables us to draw up a list of characteristics concerning the visual document layout and the voluntary typographic dispositional effects which have been produced by the printer. The collect of those visual features is essential to develop a well adapted assistance tools for old documents layout analysis. Here are some of them: complex page layout (several columns with irregular sizes), specific and unknown fonts, frequent use of ornaments (borders, ornamental letters), low spaces between the lines (contacts between characters), non constant spaces between characters and words, superposition of information layers (noise, handwritten notes...).

Figure 1: Examples of typical documents of the Renaissance corpus

3. A brief survey of classical segmentation methods

3.1.

The page layout extraction and characterization is a research field which has been fully investigated those last ten years. It can be approached according to different points of view: the physical level which deals with the separation of different kinds of informative regions (basically the separation text/graphic) and the logical level which concerns the interpretation of the previous homogeneous cutting. In this work, we are interesting in the first segmentation level. In this section, we propose to make a synthesis of the most common and used techniques of page segmentation and to present there drawbacks and inadequacy to the context of ancient printed heterogeneous documents layout retrieval.

3.2. Bottom-up, top-down and hybrid segmentation methods

These families of methods are very widespread in the literature. The first kind (bottom-up methods) works by agglomeration of pixels. Most of time those methods are *had-hoc* algorithms. For example, in [6,7,8] the authors set their algorithms on the search and the analysis of white spaces or on the connected components extraction. Others authors like Zhang and Manzini in [9,10] use multiresolution approach to complete the bottom-up segmentation.

The second class of top-down methods gathers approaches which include a precise model of the document. We usually talk about model-driven methods because the top-down segmentation algorithms are dedicated to specific documents by using a priori knowledge. In [11] for example, authors use a projection-profiled based algorithm to separate text blocks from image blocks in a Devanagari document. Nagy et Al in [12] have also developed a syntactic based system working with technical journals. They define a set of appropriate context free grammars, each defining rules to locate more and more structured entities till up to the logical objects. From the grammars, parsers are automatically obtained: they are then used to perform segmentation and labelling simultaneously. A set of alternative grammars can be used to allow different document structures to be extracted and checked.

By definition the hybrid methods use both bottom-up and top-down algorithms. A part of the analysis is done automatically (with few a priori knowledge) the rest is usually done or define by a user (specialist, simple user…). For example, authors in [13] have created a method that is based on interactions with end-user in order to create the best pattern (from a selection of characteristics) for the labelization process. In [14], the authors use a connected components analysis and a background analysis (done automatically) to segment their documents, then a classification of the extracted blocks can be achieved according to scenarios produced by the user.

3.3. Texture based segmentation methods

These methods are based on the simple observation that texts parts have regular and redundant texture patterns from background, pictures and drawings. By using classical texture based tools (usually used in natural image analysis) it's possible to set a list of relevant text based features which precisely describes the text with its typical presence of regular and ordered patterns (even with a punctual presence of noise, line skewness…).

The most famous texture segmentation methods are based on frequency analysis of the document images. They consider the document in a macroscopic point of view with its overall content. In [15,16,17] authors presents methods using wavelet based page segmentation. These papers describe how it's possible to separate text from graphics with an image orthonormal wavelet decomposition over different resolution.

In [18,19] the authors use Gabor filters. In [19], the paper describes how a bank of Gabor filters is constructed (4 orientations and 3 frequencies). This bank computes, for each pixels of the image, 12 energy measures which are compared to a threshold in order to decide if the pixel is a text or graphics pixel.

3.4. Evaluation of the existing methods on ancient books

The quality of segmentation or text/graphic separation methods has significantly increases those last ten years. But the current books have different standards of presentation from historical books. Consequently, the software which are devoted to contemporary documents recognition are often unsuitable to the processing of books of the Renaissance period. We have tested several classical methods on our corpus. The accuracy of segmentation results were linked to quantity and the quality of the threshold manually set. As we have seen in part 2, old documents are very different one from the others, thus we didn't succeed in creating an algorithm adapted to the entire corpus. The figure 2 illustrates 3 typical problems that can occur by using an algorithm based on connected component. The point 1 shows that the white space between the margin and the rest of the text is equal (and sometimes thinner) to the white space size between two words. So, it was impossible to separate them automatically. The point 2 illustrates a well known problem in old images analysis: the noise. As we can see some spot inks have the same characteristics than a letter. The point 3 deals with a current problem of connected component analysis: there are a lot of small components in bigger ones, and some can overlap others ones.

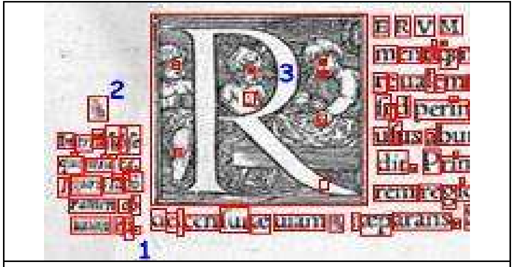


Figure 2: Typical problems of a connected component analysis

We also tried a classical document text/graphic separation method: the histogram projection analysis. Like the authors in [11], we project horizontally the pixels and identify the peaks which are significant of text lines textures. This technique gives good results when the document image is ideal: no skew, no noise…. The figure 3 illustrates a classical problem of text identification with this technique: if the text is skewed then the pixels projection doesn't create "peaks" and then it's impossible to identify text parts. It is frequent to meet situations where the alignment of the words is not guaranteed. The application of such projection based methods is very constraint and unsuited in most of case of text and drawing separation. But texture segmentation approaches also have their problems. As said in [20], one major problem of texture-based approaches is their high time complexity since different filters are needed to capture the desired local spatial frequency and orientation of the regions. Many masks are needed to extract local features and small masks do not allow detecting large-scales texture.

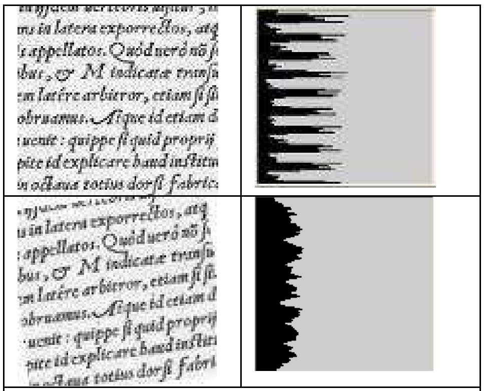


Fig 3. Different result of horizontal text projection. The second text is 8° skewed

Most of classical methods need a binarization step. The binarization of old document images is not as simple as it looks like. The growing number of articles which deals with dedicated old document binarization methods ([21,22]), testify of the importance and the complexity of this step. As shown in [23], commercial OCR and classical binarization algorithms (Sauvola, Niblack…) need new adaptive versions in order to be more robust to the noise and others old documents characteristics. The figure 4 illustrates usual problems encounter: either some thin parts can disappear or some spot ink can be consider as foreground information.

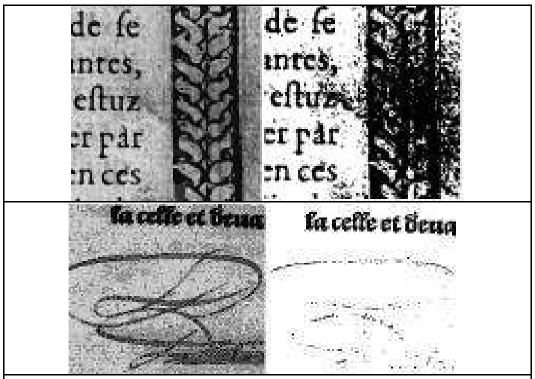


Figure 4: Binarization with Sauvola algorithm

4. Overview of our approach

The aim of our system is to realize a robust indexation that is adapted to large heterogeneous collections of ancient books. It must be adapted to a final user, non-specialist in document or image analysis. Thus, no threshold, no document model and no explicit structure have to be taken into account in the low-level image processing algorithms. Our main objective is to provide automatically a precise and complete characterization and first pre-segmentation of all the images of a book. Then, we can let the users extract the desired meta-data by using this representation and easy to use interfaces. The information retrieval must be realized without any binarization step which may cause problems in the context of ancient documents exploitation. Due to this high level of constraints, the indexation is based on both his own expertise and on the results of image analysis process. This process has been developed into two stages:

- Unsupervised automatic pixels classification using only texture features. This classification must take into account different book styles, different kind of digitalisation quality, different resolution format…
- Interactive process allowing indexation and semantic labelling of homogeneous areas. As [18], we think that external knowledge can be set by the expert end-users.

This paper mainly presents the first stage of the automatic process. The second one is still under study. That’s why in this paper, the interactive process is reduced to a simple operation: the user sets the number of classes textures he wants to see at the beginning of the indexation step.

After the study of documents characteristics and the evaluation of classical segmentation methods, we have decided to realize a robust assistance tool for large and heterogeneous collection of ancient books indexing. We have chosen 5 different and complementary texture features. Each of them is compute on grey level images in order to avoid binarization issues evoked in part 3.3. All the maps are link to a particularity of old documents visual layout and content. To do that, we have considered experts knowledge on old document structures and contents and we have analysed the psycho-visual environment of the reader to define robust and discriminant features.

Within those considerations, we have decided to develop a multiresolution approach. This choice has also been motivated by the fact that it’s very difficult to set the parameters of a texture based methods. The choices of masks sizes, and of bank filters shapes are closely dependant to the image resolution, to the font size, to the page layout symmetry, to the characters slant… Thus, for an image, we calculate 5 maps at 5 different resolutions. It gives 25 maps which are analysed and merged to feed a classification algorithm where the end-user only sets how many classes of textures he wants to have, see figure 5 for the overview.

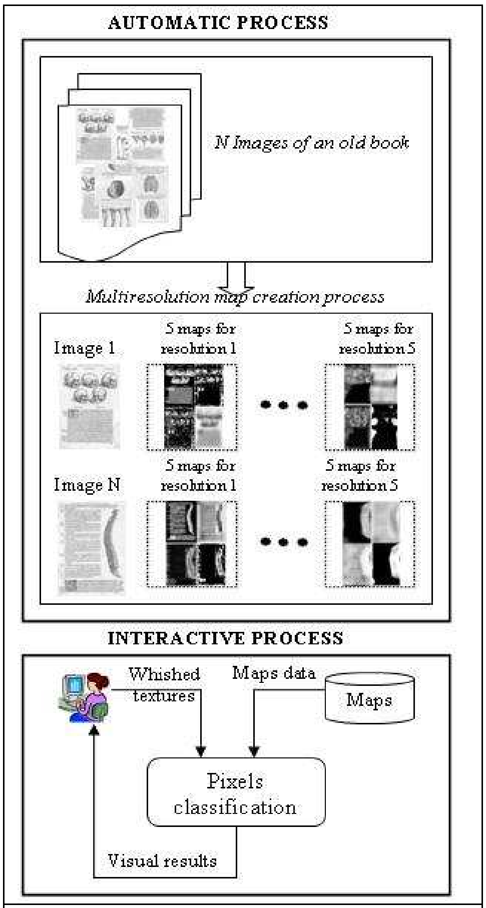


Figure 5: Overview of the approach

5. Principle of texture maps creation

Our 5 maps express the characteristics present in the old documents. A document image is a composition of 3 principal elements: the background, the text and the graphical parts. The problem is that each element is not clearly defined and then hard to identify. The background is not always uniform (presence of spot ink, different grey level…), in the same way the text lines are not always straight, font size can change from one paragraph to another, the drawings are not always rectangular and located in the same place. That's why we have created texture primitives which take into account all those aspects.

All the primitives are complementary; they have been chosen in order to characterize a maximum of visual features that are present in old documents. That's why we have decided to extract specifics characteristics like orientation based information (what is the most relevant orientation? How intense is it? Does the fluctuation of strokes orientation are important?), and like compactness which reveals local densities and the difference of used typographies. Especially here, in the case of texture based printed documents characterization, all information which are linked to the strokes frequency apparition (compactness on information) are extracted in order to obtain information about the size of the letters, the space between them, the weight, the height…

The maps can be separated into 2 family: the first ones are those linked to orientations information (principal orientation map, amplitude map, standard deviation map), the other ones are those linked to the frequencies information (background map and run length map).

Except for the background's map, all maps are constructed in the same way. A sliding window crosses the image from the top left corner to the right bottom corner. For each iteration, we calculate five different labels for the center pixel. In order to simplify issues linked to multiresolution methods, we have decided to simulate a multiresolution approach by changing the size of the sliding window instead of the size of the image. Thus, we start from a 16X16 pixel window (which corresponds to the first resolution analysis) and we compute the map, then we repeat the same map construction but this time with a 32x32 widow, and so on with a 64x64, 128x128 and a 256x256 window. The lonely (and well-known) problem with his choice of multiresolution simulation is that the border of the image is never labeled.

6. Orientation based maps

6.1. Principal orientation map

The orientation is one of the first low-level feature which is implicated in a preattentive exploratory task. It is a very influent and discriminative feature. Our interest in this map is to show different regions of the image that are visually distinct (essentially for textual regions and graphics). In our study, this feature is used to differentiate visually text parts which have different salient orientations (in particular for uppercase and lowercase texts separation). In a general manner, it allows to differentiate text from drawings, and one type of graphics from another.

The estimation of relevant directions of the image is done by computing a directional rose that has been initially proposed by Bres [23]. The directional rose computation lies on the use of the autocorrelation function, which correlates the image with itself and highlights periodicities and orientations of texture. This function has been widely used in a context of texture characterization, [24]. Its definition for a bi-dimensional signal is the following:

$$C_{xx}(k,l) = \sum_{k'=-\infty}^{+\infty} \sum_{l'=-\infty}^{+\infty} x(k',l').x(k'+k,l'+l)$$

(Formula 1)

The autocorrelation result can be analyzed by the construction of a directional rose that reveals significant directions in the analyzed image. This rose gives with a great precision all privileged orientations of the image. The directional rose represents the sum R(θi) of different values $C_{xx}(i,j)$ (defined in formula 1) in a given θi direction. So, the directional rose corresponds to the polar diagram where each direction θi that is supported by the Di line, is represented by the sum R(θi). For all points (a,b) of the D_i line we have the following relation:

$$R(\theta_i) = \sum_{D_i} C_{xx}(a,b) \quad \text{(Formula 2)}$$

From this set of values, we only keep relative variations of all contributions of each direction. So, the relative sum R'(θi) is the following :

$$R'(\theta_i) = \frac{R(\theta_i) - R_{\min}}{R_{\max} - R_{\min}} \quad \text{(Formula 3)}$$

Figure 6 shows some examples of directional roses corresponding to 4 different images.

This feature has been derived into three complementary measures (so 3 different maps): the main orientation, the main directional intensity and the variance map. Algorithm A describes the construction of the first one which allows to construct the principal orientation map. This algorithm determines the direction which is the most frequent in the tested image (the maximum of the rose of direction).

So as to avoid problems that can appear during the comparisons of circular data composed vectors, we have decided to normalize each result from the set [270°,90°] to a limited set [0,90]. The figure 7 shows the resulting orientation maps.

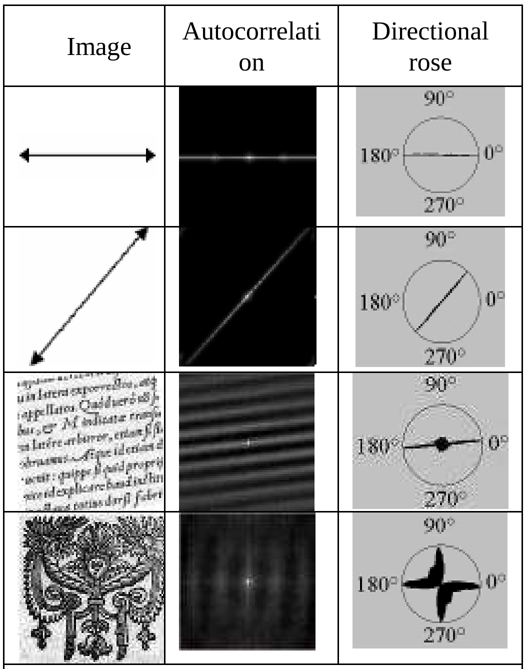
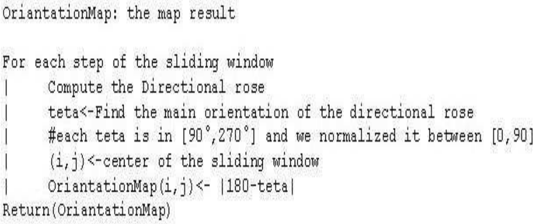


Figure 6 : Example of directional roses



```
OriantationMap: the map result

For each step of the sliding window
|    Compute the Directional rose
|    teta<-Find the main orientation of the directional rose
|    #each teta is in [90°,270°] and we normalized it between [0,90]
|    (i,j)<-center of the sliding window
|    OriantationMap(i,j)<- |180-teta|
Return(OriantationMap)
```

Algorithm A: construction of direction map

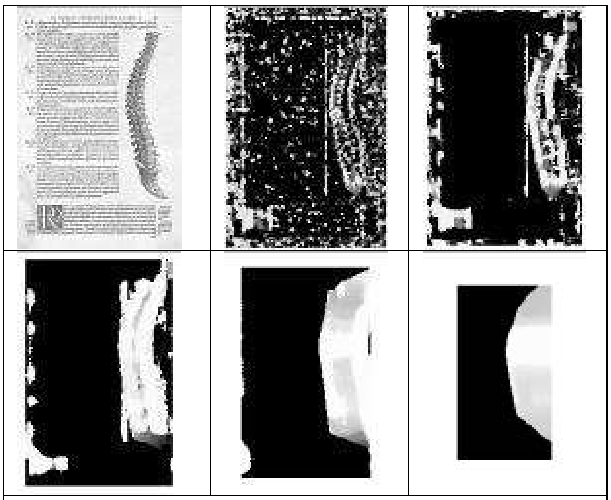
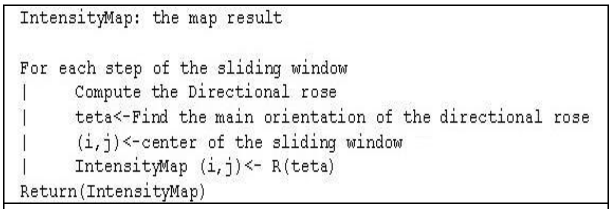


Figure 7: orientation maps for five sizes of window. The most a pixel is black, the most horizontal the relevant orientation is.

6.2. Amplitude map

The second feature that we extract (by applying algorithm B) from the rose of directions is the intensity response $R(\theta_i)$ (from formula 3); with θ_i the main relevant orientation. Base a map on $R(\theta_i)$ results is very interesting because it permits to differentiate zones that are visually very different. As a matter of fact we have experimentally notice that the intensity of the main orientation of the rose is rather different for graphics to text parts.



```
IntensityMap: the map result

For each step of the sliding window
|     Compute the Directional rose
|     teta<-Find the main orientation of the directional rose
|     (i,j)<-center of the sliding window
|     IntensityMap (i,j)<- R(teta)
Return(IntensityMap)
```

Algorithm B: construction of intensity map

The figure 8 provides examples of intensity maps.

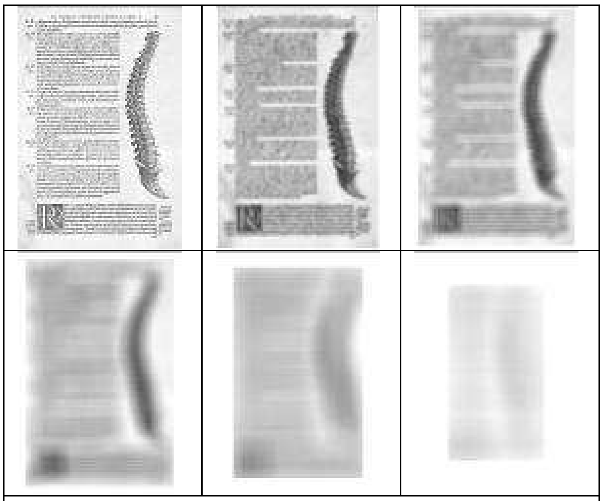
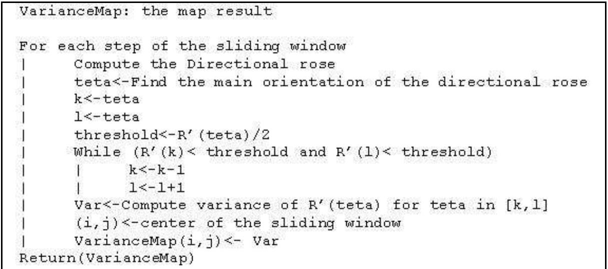


Figure 8: intensity maps for five sizes of window. The most a pixel is black, the most important the intensity is.

6.3. Standard deviation map

The last feature extracted from the rose of directions (by applying the algorithm C) is the variance map. By studying a lot of roses of directions, it reveals that the shape of the rose around the main intensity is quite discriminating. For example, the rose of a text part (fig 4 part 3) has a specific shape. Its looks like a thin “peak” for the horizontal orientation. It’s due to the fact the horizontality of the text is so important in relation to the others orientations. On the contrary the rose of a drawing (fig 4 part 4) have not only one important orientation and there is no thin “peak” around the main orientation but a wide one. So, by studying the shape of the rose it’s possible to extract interesting information.

With the algorithm C, we characterize the shape of the peak by studying the standard deviation around $R^{'}(\theta_i)$ (from formula 4); with θ_i the most relevant orientation.



```
VarianceMap: the map result

For each step of the sliding window
|     Compute the Directional rose
|     teta<-Find the main orientation of the directional rose
|     k<-teta
|     l<-teta
|     threshold<-R’(teta)/2
|     While (R’(k)< threshold and R’(l)< threshold)
|     |     k<-k-1
|     |     l<-l+1
|     Var<-Compute variance of R’(teta) for teta in [k,l]
|     (i,j)<-center of the sliding window
|     VarianceMap(i,j)<- Var
Return(VarianceMap)
```

Algorithm C: construction of variance map

The figure 9 illustrates examples of variance maps.

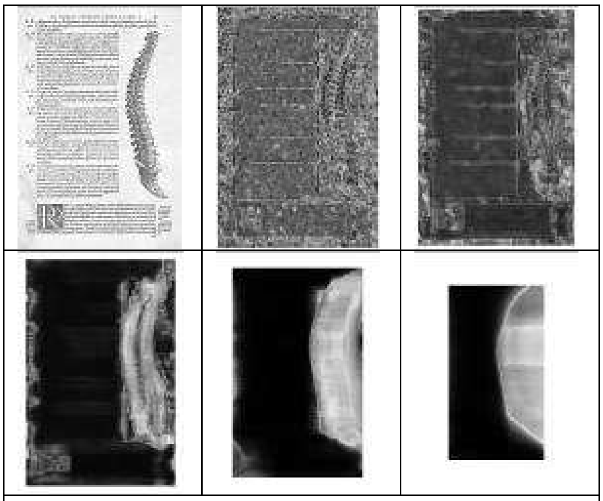


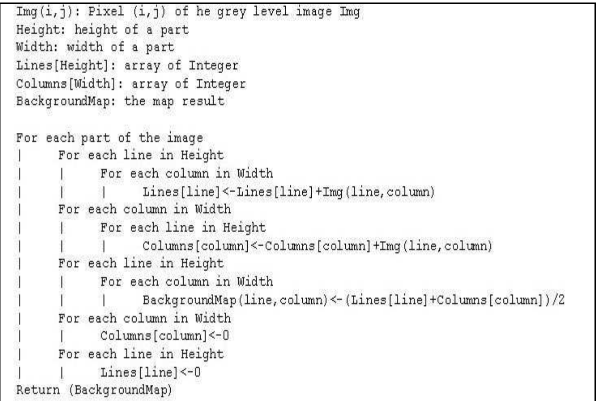
Figure 9: variance maps for five sizes of window. The most a pixel is black, the most important the variance is.

In this example, the text is visually revealed by directional roses which have significant horizontal picks and a noticeable global anisotropy for all other directions.

7. Frequencies maps

7.1. Map of background

The aim of this map is to characterize the transitions between foreground information and background information. By studying wide areas, it’s possible to characterize interest regions from non interesting regions. This map is the only one that is not computed at different resolutions. Here we use a 5 steps XY-cut algorithm. At step n, the image is recursively cut into 4 parts and so on... For a step n, we compute the algorithm D for each part.



```
Img(i,j): Pixel (i,j) of he grey level image Img
Height: height of a part
Width: width of a part
Lines[Height]: array of Integer
Columns[Width]: array of Integer
BackgroundMap: the map result

For each part of the image
|     For each line in Height
|     |     For each column in Width
|     |     |     Lines[line]<-Lines[line]+Img(line,column)
|     For each column in Width
|     |     For each line in Height
|     |     |     Columns[column]<-Columns[column]+Img(line,column)
|     For each line in Height
|     |     For each column in Width
|     |     |     BackgroundMap(line,column)<-(Lines[line]+Columns[column])/2
|     For each column in Width
|     |     Columns[column]<-0
|     For each line in Height
|     |     Lines[line]<-0
Return (BackgroundMap)
```

Algorithm D: Construction of background map

The figure 10 illustrates a result of 3 background maps computed from the same image. For a better visualisation, the result of the algorithm is normalized between 0 and 255.

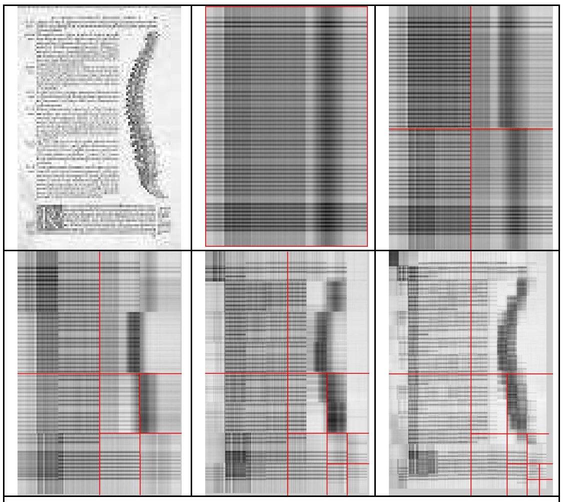


Figure 10: Background maps for step n 1, 2, 3, 4 and 5.

7.2. Run length map

This map lies on the compactness of information. That’s why we studying the run length of the grey level pixels of the image. This feature permits to express differences between different kind of letters (big ones from little ones) and also different kind of drawings (the ones composed with a lot of strokes from the ones composed with few strokes). This map is different from the background map since we express micros information (information extracted from little parts) in density map. So, this last map is constructed by applying the algorithm E. It lies on the study of Run Length pixels sizes. The figure 11 illustrates how we calculate a run length pixel size (Rs).

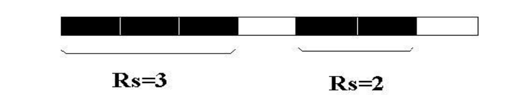
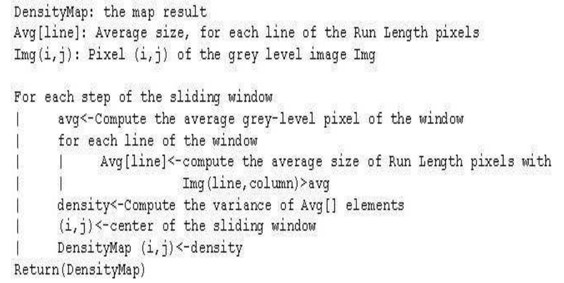


Figure 11: Example of a run length black pixels size computation



```
DensityMap: the map result
Avg[line]: Average size, for each line of the Run Length pixels
Img(i,j): Pixel (i,j) of the grey level image Img

For each step of the sliding window
|     avg<-Compute the average grey-level pixel of the window
|     for each line of the window
|     |     Avg[line]<-compute the average size of Run Length pixels with
|     |                 Img(line,column)>avg
|     density<-Compute the variance of Avg[] elements
|     (i,j)<-center of the sliding window
|     DensityMap (i,j)<-density
Return(DensityMap)
```

Algorithm E: construction of density map

The figure 12 illustrates examples of density maps.

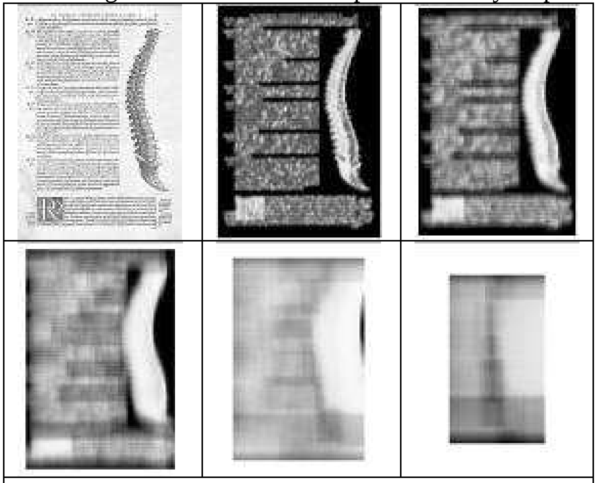


Figure 12: Density maps for five sizes of window. The most a pixel is black, the less important the density is.

8. Application to indexation

After the construction of all the texture maps, we have for each pixel of the original image 25 features. We compute the Clara classification algorithm where each pixel corresponds to an observation (Clustering LARe Applications). Clara is fully described in [25]. The CLARA classification approach is built in statistical analysis packages. It draws *multiple samples* of the data set, applies a p*artitioning around representative objects (also called Medoids and PAM method by authors)* on each sample, and gives the best clustering as the output. The *PAM method* starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering. Compared to other partitioning methods, it can deal with much larger datasets. This is achieved by considering sub-datasets of fixed size such that the time and storage requirements become linear in n rather than quadratic. This choice is motivated by the fact that all our images are in high resolution. Thus the number of pixels is too important to use another classification algorithm. Next sections present qualitative results of the Clara classification and underline the ability of our system to segment digitalized pages with a possible tuneable number of classes which is at the basis of a separation in coarse or fine regions.

8.1. Qualitative scheme of the inner page data classification

Figure 13 illustrates visual results for a three and a six texture classes. separation.

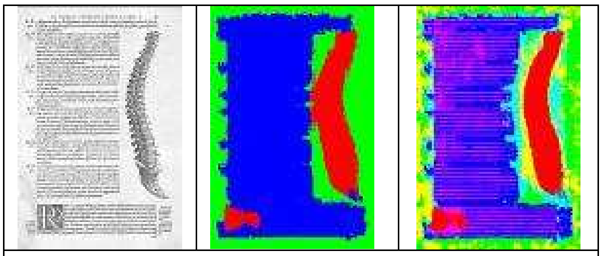


Figure 13: classification with 3 and 6 classes

As we have said before, we can't guess what each user wants. That's why we prefer to let him decide what is looking for. Figure 13 illustrates this principle. For a 3 class classification, we obtain the background, the texts parts and the drawings. For a 6 class classification, the space-lines appear (in pink).

The influence of the ideal number of classes on classification results is hard to establish because it is strongly correlated with the page content. Generally by increasing the number of class, more details appear. Most of time, the "drawing class" is separated in two classes: "borders drawings" and "middle of drawings". In regard to the background, it can be separated in two classes: background near foreground classes and background far away from it (fig 14). In the same way text is separated in different classes (body, notes in margin, titles…). The main problem still remains with the difficulty to set automatically the number of classes.

The main interest of our method is a total free consideration of the document context. We do not require any a priori knowledge on the content appearance: especially here it is not necessary to know if the text is skewed, if there is noise, or if images of a same collection must have the same size. In this study, we are only processing grey-level images without binarization to conduct the classification process.

.The figure 14 illustrates a classification results for texts with different orientations. According to the number of whished classes, it's possible to separate text from background or each texts part.

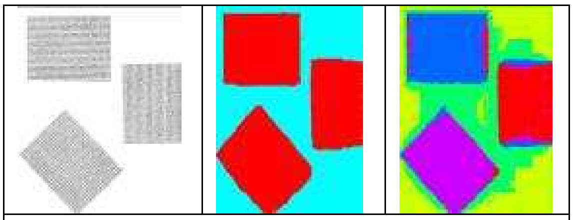
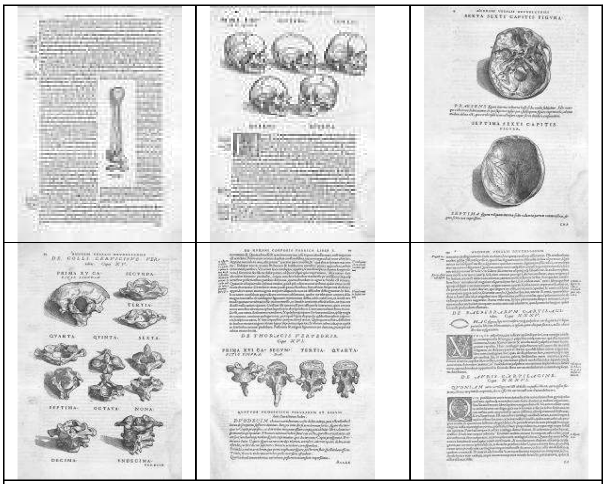


Figure 14: classification results for 2 classes and 5 classes

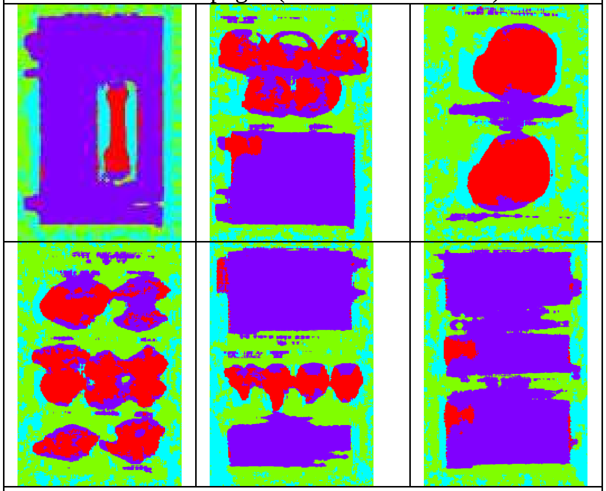
In most cases, we have visually notice that for a 3 class classification problem we have quite good results. The main problem comes from non constant transitions that exist between blocks and from very huge characters sizes in some titles. Except those two special cases, our method gives quite good results and allows us to realize our purpose: a relevant classification of pixels into pre-labelled regions that also highlights the document visual content with robust extractors (with same results whatever are the typography, the fonts, the background noise and the resolution).

8.2. Qualitative scheme of the entire book classification

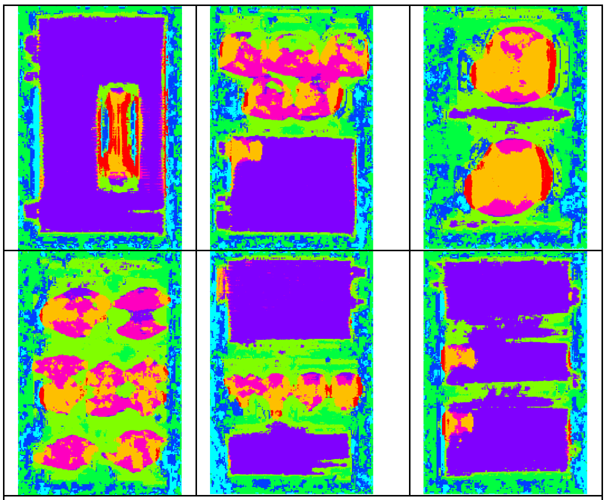
The pixel classification takes a real sense if we compute it on an entire book. In this way, it's possible for a user to set is own indexation. This choice permits, for example, to set the granularity for the research of text only by changing the number of classes, user cans have either all the text parts in the same class or notes in margin and body text in two separated classes. Those results could be the start for a navigation tool that would permit to navigate from one class to another, all over the book. Thus, just by clicking on the image it's possible to help users to find similar zones in other pages. This classification can also be seen as a learning step for a document image retrieval tool. By setting the number of texture classes he wants to see, the user indicates what is important to be taking into account for the comparison of layouts. The direct analysis of colour distribution for a layout based comparison is one of the next tracks we have under development. The figure 15 shows results of a classification computed on 6 images of the same book. So, if for two pages the colour is the same that is to mean that it's the same class.



Tested pages (from the same book)



Results of a 4 classes classification (simulation of text/drawings separation)



Results of a 8 classes classification

Figure 15: examples of classifications for several pages of a book

8.3. Exploitation and future improvements

The classification results have been validated in its *three* clusters version by using the only main orientation features, [26]. Those results have allowed to classify pages layouts according to the distribution and quantity of information related to *background, text* and *graphics*. The introduction of complementary perceptual new features is the first step to a more complete and detailed segmentation scheme. It will be used for a fine content based indexation. To realize this objective, we must improve 3 main points.

First of all: the computing time. All the algorithms presented in this article (except classification step) are time consuming because they are linked to the size of the images. For the moment the feature maps are generated offline. The classification process is the only one to be done in real-time.

Secondly: the selection of features for the classification process is another important part to improve. For the moment we take into account all the features, but it will be more interesting to keep only the most "significant" ones to avoid misclassification pixel results. It will be also interesting to complete the texture maps with some information like the pixels positions or the pixels neighborhood characteristics.

Thirdly: Create interesting user applications is the last important objective of our work. As a matter of fact, help a user to navigate more easily in a book or in a collection of books is our main goal. For example it will be great if after a click on a drawing the user could be taking away to the next one. Another interesting idea sets on the identification of the titles in order to automatically create synopsis of a book. Create a document image retrieval program is also one of our future interest. It will be interesting to be able to answer to this kind of questions: “give me all the pages that look like to the one I give you”.

Currently, we have first results on information retrieval with partial image samples queries (a manual image selection realized by the expert user). The output of this process is a list of pages selections whose content is similar to the initial selection one. The tests currently relate to the definition of measures of similarities to compare the initial selection to all variable sized answers which are contained in the corpus pages.

9. Conclusion

In this paper we present an efficient method for pixel classification adapted to ancient printed documents. The originality of our approach lies on the development of a new extractor and analysis tool which is dedicated to ancient document images without any parameters, thresholds, models, or structure information. The result of a document analysis is employed as features for a relevant pixels classification. Current results are very promising. It needs still some more investigation to achieve a complete indexation system for non specialist users.

Acknowledgements: These studies had the support of the CESR-CNRS (Tours – France) which provides images databases, [2]. We also want to acknowledge C. Saint-Jean for his software support in data analysis.

10. References

[1] Bibliothèque Gallica. http://gallica.bnf.fr/

[2] Bibliothèque humaniste du CESR de l’université de Tours. http://www.**cesr**.univ-tours.fr

[3] Digibook, http://www.i2s-bookscanner.com/fr/default.asp

[4]Institut en Recherche et Histoire de textes, http://www.irht.cnrs.fr

[5] TRINH, E., De la numérisation à la consultation de documents anciens. Thèse de doctorat en Informatique. Insa de Lyon. 175p., 2003.

[6]HADJAR, K., INGOLD, R. *Arabic Newspaper Page Segmentation*, in ICDAR, pp. 895-900, 2003.

[7] BREUEL, M;T. Two Geometric Algorithms for Layout Analysis, Xerox Palo Alto Research Center, in Document Analysis System, pp. 214-222, 2002.

[8] LIE, J., HU, J., WU, L. Page segmentation of chinese newspaper, in Pattern recognition, 2695-2704, 2002.

[9] L. CINQUE, L. LOMBARDI, G. MANZINI, A multiresolution approach for page segmentation, Pattern Recognition Letters 19(2): 217-225 (1998)

[10] TAN, C.L., ZHANG, Z. Text block segmentation using pyramid structure, SPIE Document Recognition and retrieval, vol.8, pp. 297-306, 2001.

[11] KHEDEKAR, S.,RAMANAPRASAD, V., SETLUR S., GOVINDARAJU, V. Text - Image Separation in Devanagari Documents, in proc. On ICDAR, vol.2, pp. 1265-1269, 2003.

[12] G. NAGY, S. SETH, M. KRISHNAMOORTHY, AND M. VISWANATHAN. *Syntactic Segmentation and Labeling of Digitized Pages from Technical Journals.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15(7): p737-747, 1993.

[13] ROBADEY, L. Une méthode de reconnaissance structurelle de documents complexes basée sur des patterns bidimensionnels. PhD Université de Fribourg, 195p. 2001.

[14] J.Y. RAMEL, S. LERICHE, M.L. DEMONET, S. BUSSON, User-driven Page Layout Analysis of Historical Printed Books, Rapport Interne, LI Tours, 2005.

[15] P.GUPTA, N.VOHRA, S.CHAUDHURY, S.DUTT, Wavelet Based Page Segmentation, Proc. of the ICVGIP, pp.51-56, 2000.

[16] JIA LI, ROBERT M. GRAY, Context-Based Multiscale Classification of Document Images Using Wavelet Coefficient Distributions, *IEEE Trans. Image Processing*, vol. 9, pp. 1604-1616, Sept. 2000.

[17]K.ETEMAD, D.DOERMANN, R.CHELLAPPA, *Multiscale Segmentation of Unstructured Document Pages Using Soft Decision Integration,* IEEE Trans. on Pattern Analysis and Machine Intelligence, v.19 n.1, p92-96, 1997.

[18] A.K. JAIN, S.BHATTACHARJEE. Text segmentation using Gabor filters for automatic document processing, Machine Vision and Applications,Vol.5,169 – 184. 1992.

[19] P.Basa P.S. Sabari, R.Nishikanta, P. A G Ramakrishnan, Gabor filters for Document analysis in Indian Bilingual Documents, In *Proceedings International Conference on Intelligent Sensing and Information Processing.*, pages pp. 123-126, 2004.

[20] LEE, S.W., RYU, D.S., Parameter-Free Geometric Document Layout Analysis, IEEE Tran.on Pattern Analysis and Machine Intel., Vol. 23, No. 11, p1240-1256, 2001.

[21] H. HAMZA, E. SMIGIEL, AND A. BELAID, Neural Based Binarization Techniques, In proc. Of ICDAR, pp.317-321, 2005.

[22] J.HE, Q.DO, A.DOWNTON, J.H.KIM, A Comparison of Binarization Methods for Historical Archive document, 8th International Conference on Document Analysis and recognition, pp.538-542. 2005.

[23] S BRES, Contributions à la quantification des critères de transparence et d’anisotropie par une approche globale. PhD Thesis, 1994.

[24] W.K. PRATT, Digital Image Processing, 2nd edition *New- York : Wiley*, 1991, p230.

[25] KAUFMAN, L.,ROUSSEEUW, P. J. Finding Groups in Data,, Wiley Series in Probability and Mathematical Statistics,John Wiley and Sons Inc., 340 p.,1990.

[26]N JOURNET, V. EGLIN, R.J-Y RAMEL, R. MULLOT, Text/Graphic labelling of Ancient Printed Documents. Dans International conference on document analysis and recognition, ICDAR, Séoul (Corée). pp. 1010-1014, 2005.