16

Natural Scene Text Understanding

Céline Mancas-Thillou, Bernard Gosselin
Faculté Polytechnique de Mons
Belgium

1. Introduction

In a society driven by visual information and with the drastic expansion of low-priced cameras, vision techniques are more and more considered and text recognition is nowadays a fast changing field, which is included in a large spectrum, named text understanding.
Previously, text recognition was dealing with documents only; those which were acquired with flatbed, sheet-fed or mounted imaging devices. Recently, handheld scanners such as pen-scanners appeared to acquire small parts of text on a fairly planar surface such as that of a business card. Issues having an impact on image processing are limited to sensor noise, skewed documents and inherent degradations to the document itself. Based on this classical acquisition method, optical character recognition (OCR) systems have been designed for many years to reach a high level of recognition with constrained documents, meaning those falling into traditional layout, with relatively clean backgrounds such as regular letters, forms, faxes, checks and so on and with a sufficient resolution (at least 300 dots per inch (dpi)). With the recent explosion of handheld imaging devices (HIDs), i.e. digital cameras, standalone or embedded in cellular phones or personal digital assistants (PDAs), research on document image analysis entered a new era where breakthroughs are required: traditional document analysis systems fail against this new and promising acquisition mode and main differences and reasons of failures will be detailed in this section. Small, light, and handy, these devices enable the removal of all constraints and all objects, such as natural scenes (NS) in different situations in streets, at home or in planes may be now acquired! Moreover, recent studies [Kim, 2005] announced a decline in scanner sales while projecting that sales of HIDs will keep increasing over the next 10 years.

1.1. Challenge of natural scene text understanding

First of all, in order to understand challenges of this field, new imaging conditions and newly considered scenes need to be detailed. The new imaging conditions deal with:

- **Raw sensor image and sensor noise**: in low-priced HIDs, pixels of a raw sensor are interpolated to produce real colours, which can induce degradations. Demosaicing techniques, viewed more as complex interpolation techniques, are sometimes required. Moreover, sensor noise of an HID is usually higher than that of a scanner.
- **Viewing angle**: scene text and HIDs are not necessarily parallel creating perspective to correct.

Source: Vision Systems: Segmentation and Pattern Recognition, ISBN 987-3-902613-05-9,
Edited by: Goro Obinata and Ashish Dutta, pp.546, I-Tech, Vienna, Austria, June 2007

- **Blur**: during acquisition, some motion blur can appear or be created by a moving object. All other kinds of blur, such as wrong focus, may also degrade even more image quality.
- **Lighting**: in real images, real (uneven) lighting, shadowing, reflections onto objects, inter-reflections between objects may make colours vary drastically and decrease analysis performance.
- **Resolution and Aliasing**: from webcam to professional cameras, resolution range is large and images with low resolution must also be taken into account. Resolution may be below 50 dpi which causes commercial OCR to fail. It may lead to aliasing creating fringed artefacts in the image.

The newly considered scenes represent targets such as:

- **Outdoor/non-paper objects**: different materials cause different surface reflections leading to various degradations and creating inter-reflections between objects.
- **Scene text**: backgrounds are not necessarily clean and white, and more complex ones make text extraction from background difficult. Moreover scene text such as that seen in advertisements may include artistic fonts.
- **Non-planar objects**: text embedded in bottles or cans suffer from deformation.
- **Unknown layout**: there is no a priori information on structure of text to detect it efficiently.
- **Objects in distance**: distance between text and HIDs can vary, and character sizes may vary in a wide range, leading to a wide range of character sizes in a same scene.

Fig. 1. Samples of natural scene images.

The main challenge is to design a system as versatile as possible to handle all variability in daily life, meaning variable targets with unknown layout, scene text, several character fonts and sizes and variability in imaging conditions with uneven lighting, shadowing and aliasing. Our proposed solutions for each text understanding step must be context independent, meaning independent of scenes, colours, lighting and all various conditions. Hence we focus on methods which work reliably across the broadest possible range of NS images, such as displayed in Figure 1.

1.2. Numerous applications

As HIDs become more and more powerful, on-the-fly image processing becomes possible, opening up a new range of applications. Nevertheless, today's HIDs are easily connected to various networks and supplementary computing resources. Starting from sign recognition for foreigners for the 2008 Olympic Games in Beijing, automatic license plate recognition to driver assisted systems with text projection on windshields, various situations could be

handled. Interesting applications such as mobile phones operating as fax machines even led to strict sanctions in Japanese bookstores!
Visually impaired people are directly affected by such research [Thillou et al, 2005]. With an HID and sufficient resources, scene in daily life may be analyzed to give them access to text and, coupled with a text-to-speech algorithm, make them "read" book covers, banknotes, labels on office doors, medicine labels and so on. For the blind community, such devices are really expected.
Another promising application is the one of visual landmark-based robot navigation. Several kinds of robot navigation may be listed such as dead-reckoning, map-based navigation, positioning sensor-based navigation or landmark-based navigation, which can be divided into natural and artificial landmarks. Natural landmarks may be designed on purpose for indoor robot navigation, such as room numbers [Mata et al., 2001], displayed in Figure 2, but may also be part of real life such as natural scenes. Even if conditions of navigation are still constrained, natural landmark-based one is very promising and satisfying results already appeared. Hence either nameplates, information signs or any text embedded in images contain large quantities of useful semantic information. Text understanding may be useful in high level robot navigation, such as path planning or goal-driven navigation. Applications are very numerous and currently only limited by imagination. Scene text is an important feature to understand for all these applications.

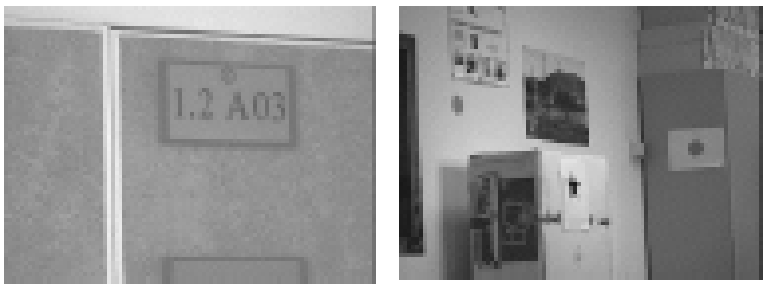

Fig. 2. Natural and artificial landmarks used in [Mata et al., 2001].

1.3. Overview of the chapter

How does one achieve the pre-cited applications? By using a text understanding system, which encompasses three main steps: text detection and localisation, text extraction from background, and text recognition.
Text detection and localisation find answers to the question: "Is there any text and where is it?". This part has been extensively studied during previous years. Text extraction from background is the field dealing mainly with uneven lighting and complex backgrounds. It is a paramount step to prepare data for OCR. Classical image segmentation such as separating sky from mountains does not need as much accuracy as text extraction, which is considered more as object-driven segmentation. Actually, text is a meaningful object which has to be extracted properly to be better recognised afterwards. Text recognition is the final step to convert character images into ASCII values to understand text and use it for particular applications.

Other NS text analysis steps such as warping, mosaicing or text tracking are also part of text understanding systems for different applications and for more details, the reader may refer to the overall state-of-the-art of Liang et al [Liang et al., 2005].
Particular focus is cast on the text extraction step: it is declared as the "most important factor for high performance" by In-Jung Kim [Kim, 2005]. Slightly studied since the inception of camera-based text analysis, text extraction suffers from imaging conditions. On the other hand, the text detection step will be only briefly mentioned in this chapter. S. Lucas, after the ICDAR (International Conference on Document Analysis and Recognition) 2005 text locating competition [ICDAR Competition, 2003], was able to conclude that "in text locating, [...] there has been a significant advance in performance [and] most easy-to-read (for humans) text is now well detected". He also mentioned that variations in illumination such as reflections cause significant problems for text understanding. Hence, considerations on uneven lighting and how to circumvent it for efficient text extraction are particularly highlighted as well.
Section 2 will describe background on text extraction and additional steps to achieve an efficient text understanding system such as character segmentation. Literature survey is also browsed along these lines. Section 3 will form the main body of the chapter with our selective metric clustering (SMC) algorithm for text extraction. The proposed solution is detailed with justifications of each step and several experiments including comparisons with other recent techniques to highlight the performance of the whole method. Section 4 will be devoted to segmentation of extracted text into individual units such as characters to improve recognition afterwards. Log-Gabor filters, well designed for NS images, are used here for the first time for character segmentation into individual components. Section 5 will describe home-made recognition used for natural scene characters with details to build an efficient training database. Finally, Section 6 will end this chapter with conclusions about text understanding for NS images and remaining issues.

2. State-of-the-Art of Natural Scene Text Understanding

Text understanding systems include three main topics: text detection, text extraction and text recognition. We assume images input into our system have previously detected text if there is any in the image. A text extraction system usually assumes that text is the major input contributor, but also has to be robust against variations in the detected text's bounding box size. For a detailed survey on text localisation methods, usually grouped into region-based, edge-based, connected components-based and texture based, the reader may refer to the survey of Jung et al. [Jung et al., 2004]. Hence, this section first details state-of-the-art methods of text extraction and then discusses character segmentation to improve text extraction and consequently, text recognition.
Text extraction is a critical and essential step as it sets up the quality of the final recognition result. It aims at segmenting text from background, meaning isolated text pixels from those of background. A very efficient text extraction method could enable the use of commercial OCR without any other modifications. Due to the recent launch of the NS text understanding field, initial works focused on text detection and localisation and the first NS text extraction algorithms were computed on clean backgrounds in the gray-scale domain. Following that, more complex backgrounds were handled using colour information. Identical binarisation methods were at first used on each colour channel of a predefined colour space without real efficiency for complex backgrounds, and then more sophisticated

approaches using 3D colour information, such as clustering, were considered. The classification of text extraction methods is displayed in Figure 3 and will be detailed further.

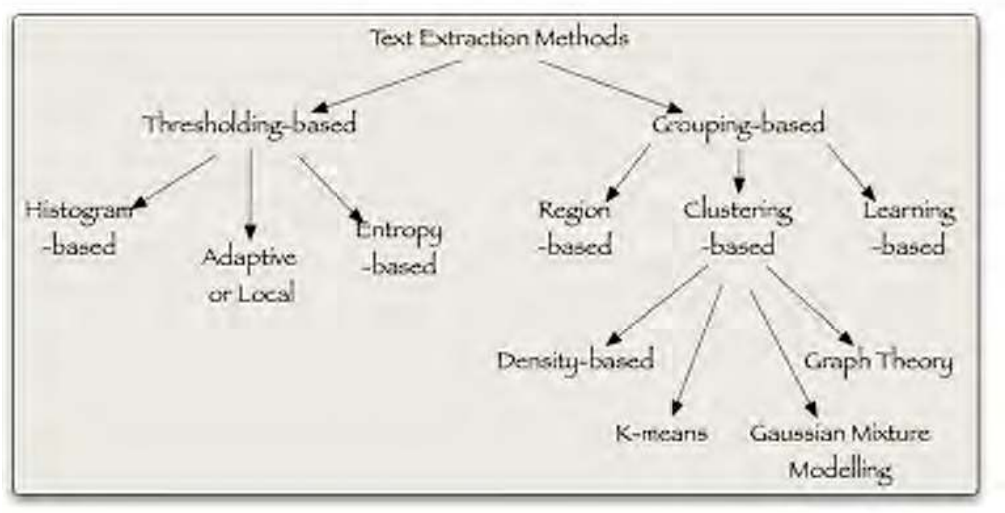

Fig. 3. Classification of text extraction methods.

- **Thresholding-based methods**

Thresholding-based methods, as the name implies, define a threshold globally (for the whole image) or locally (for some given regions) to separate text from background. **Histogram-based thresholding** is one of the most widely used techniques for monochrome image segmentation. Images are composed of several homogeneous regions with different pixel values; text is one of these regions. A histogram counts the number of each pixel value. Peaks (or modes) in histogram (meaning that several pixels have this same value) are considered as regions to segment. The threshold is chosen as the value corresponding to the valley between two peaks. The most referenced method is the one described by Otsu [Otsu, 1979], which minimises the weighted sum of within-class variances of the foreground and background pixels to get an optimum threshold as in [Thillou et al., 2005] for a visually impaired-driven application. Messelodi and Modena [Messelodi & Modena, 1992] chose two thresholds to strictly isolate the peak corresponding to text. These methods work well with low computational resources but are applied mostly on gray-scale images or colour channels independently. Moreover, they fail for images without any obvious peaks or with broad valleys which appear with complex backgrounds and slightly varying colours. **Adaptive or local binarisation techniques** define several thresholds for different image parts depending upon the local image characteristics. Several papers [Li & Doermann, 1999; Zandifar et al., 2005] for video text extraction used the Niblack's method [Niblack, 1986] where the threshold depends on local mean and standard deviation over a square window of size to define. An extension is the method of Sauvola and Pietikäinen [Sauvola & Pietikäinen, 2000] where the threshold is defined according to two parameters to define. Gllavata et al. [Gllavata et al., 2003] created their own local thresholding based on beginning and end of text lines. They assumed fairly horizontal text lines which is not necessarily the case for NS

images. Adaptive binarisations may handle more degradation (uneven lighting, varying colours) than global ones but suffer to be too parametric which is not versatile. Moreover, these techniques still consider gray-scale images only and were mainly used for video caption text or documents with clean backgrounds. **Entropy-based methods**, appropriately named, use the entropy of the gray levels distribution in a scene. Li and Doermann [Li & Doermann, 1999] minimised the cross-entropy between the input video gray-scale frame and the output binary image. The maximisation of the entropy in the thresholded image means that a maximum of information was transferred. Du et al. [Du et al., 2004] compared Otsu's binarisation and different entropy-based methods to assess that the joint relative entropy performs best on RGB channels independently for video caption text. Entropy-based techniques have been little referenced in NS context and applied only on gray-scale images or separate channels of a particular colour space.

Thresholding-based methods are lightweight enough to fit low-computational resources; that is why they are preferred for particular applications with clean backgrounds for their satisfying results on gray-scale images. Nevertheless, they are not the most suitable to handle complex backgrounds, varying colours, uneven lighting and so on.

- **Grouping-based methods**

The following methods group text pixels together according to certain criteria to extract text from background. Most popular techniques are clustering-based and are detailed further below. **Region-based approaches** include spatial-domain region growing, splitting and merging, and have been extensively used in general colour image segmentation with unknown content. These methods may be classified into two groups: top-down and bottom-up. The first one has been experienced in Kim et al. [Kim et al., 2005] by starting with the entire image and going towards smaller parts with differences between gray values exceeding a certain value. A merging process followed to refine results. In video captions, a bottom-up approach has been used by Lienhart and Wernicke [Lienhart & Wernicke, 2002]. Based on the assumption that the text contrasts well with its background, a seed around borders of text bounding box was chosen to be sure it belonged to background. With the Euclidean distance between RGB colours in a 4-neighborhood, background was extended if the distance remained below a particular value. In these two methods, a value was pre-defined and as all parametric methods, it is not versatile and cannot handle all degradations of NS images. Moreover region-based approaches are computationally quite expensive. However, they use spatial information which groups text pixels efficiently. **Learning-based approaches** have initially been designed to mimic humans by learning a training database to further recognise similar patterns. Text has interesting spatial properties and may be considered as a particular texture. Several classifiers are widely applied for pattern recognition and multi-layer perceptrons (MLP) and self-organising maps (SOM) are the most studied in text extraction. Neural networks, MLP or SOM, composed of linked neurons such as human brains, may model very general functions with any degree of non-linearity to separate pixels of text and non-text into two classes. In Hamza et al. [Hamza et al., 2005], a cascaded approach for colour historical documents with a SOM followed by an MLP was used in the training part while the trained MLP was used for testing alone. It overcame results of thresholding-based methods. Nevertheless, a training database is needed and with the wide range of NS images, this task is difficult to realise. Moreover it implies storage problems and labelling of the whole training database before being effective. **Clustering-based approaches** group colour pixels into several classes assuming that colours tend to

form clusters in the chosen colour space. They belong to unsupervised segmentation while learning-based approaches belong to supervised segmentation. Clustering-based algorithms are the most renowned and efficient methods for NS images. They are often considered as the multidimensional extension of thresholding methods. The most popular method is k-means but its generalisation, Gaussian Mixture Modelling (GMM), is more and more exploited.

1. **From density-based clustering to Mean-Shift:** Extension of histogram-based thresholding, density-based clustering is applied on colour images and needs the computation of a 3D histogram to handle colour dimensions. Adjacent colours are then merged towards the nearest highest peak. The algorithm terminates when the number of desired colours is obtained. It was used on coloured books and journal covers with relatively clean background and video scene text in Sobottka et al. [Sobottka et al., 1999]. Perroud et al. [Perroud et al., 2001] used a 4D-histogram with the RGB colour space and the channel of luminance. The Mean-Shift algorithm, first created by Fukunaga in 1975 and extended by Comaniciu [Comaniciu, 2000], seeks the "mode", point of highest density, of the 3D colour histogram. First it defines a window centered randomly at a point. The mean over the window is computed and the Mean-Shift is expressed according to the density estimate. This successful technique has not been tested on NS text, but more generally on colour segmentation.
2. **From graph theory to spectral clustering:** In graph theory concept, colour pixels are merged based on the minimum Euclidean distance (or another one) in a connected neighbourhood to form regions in the colour space. These merged pixels are represented by vertices in the graph and links between geometrically adjacent regions have weights that are proportional to the colour distance between the regions they connect. They describe a hierarchy to solve by graph theory such as in [Lopresti & Zhou, 2000; Wang et al., 2004]. It may be solved by finding a minimum of normalised cuts or more generally by spectral clustering. This latter method computes eigenvectors of the Laplacian matrix to have representation in the spectral space. The Laplacian matrix L is equal to $L = I - D^{-1/2} A D^{-1/2}$ where I is the identity matrix, D is the diagonal matrix whose diagonal elements are the sum of corresponding row of A, the affinity matrix, by then stacking the k eigenvectors in columns in a matrix which will be normalised, and fed to the k-means algorithm. k is the desired number of clusters. The main advantage of this technique is the invariance against varying colours.
3. **From k-means to GMM**: k-means is considered the most used technique in clustering. The procedure follows a simple approach to classify colour pixels in a defined colour space through a certain number of clusters (k) fixed a priori. The main idea is to define k centroids, one for each cluster and compute a defined distance between points and centroids. Iteratively, all pixels belong to a cluster whose centroid is the nearest one. Another way to deal with clustering issues is to use a model-based approach, also called probabilistic clustering. In practice, each cluster can be mathematically represented by a parametric distribution (assumed to be Gaussian). All colour pixels are therefore modelled by a finite mixture of these distributions and parameters are automatically computed with the Expectation-Maximisation (EM) algorithm or one of its variants.

There has been little experimentation done on text extraction using other clustering methods such as fuzzy c-means, which is the extension of k-means with a degree of belonging to a

cluster. As all methods can obviously not be cited in this thesis, the reader may refer to the survey of Berkhin [Berkhin, 2002].
Faced with multiple degradations and diversity of situations, text extraction alone is not sufficient to produce recognisable text for off-the-shelf OCR. Work on OCR itself may be done to improve results such as recognition of much degraded characters [Ojima et al., 2005] without any pre-processing. Nevertheless, since the main aim is to provide a solution having satisfying performance for several kinds of NS images, it is better to improve text quality beforehand, and only if necessary. Typical OCR fails against medium-quality extracted text having background portions, misalignment, too many adjoining characters such as text on a wavy tee-shirt where some characters are closer than others or totally connected. Hence to provide a very high quality extracted text, some post-processing is sometimes required and literature mainly counts rule-based methods and segmentation algorithms of characters into individual components.

- **Rule-based methods** are useful to remove spurious parts of non-textual extracted parts. Gatos et al. [Gatos et al., 2005] defined several thresholds and global variables such as the maximum and minimum number of expected characters in a text line along with the maximum and minimum number of lines in a paragraph, while Esaki et al. [Esaki et al., 2004] defined a number of rules about character sizes to remove certain parts after a global binarisation method. Text properties, such as geometry, alignment, colour, differentiating text from other objects may be used to improve text extraction algorithms. Nevertheless, strict rules with thresholds are not exploitable at all for NS images.
- Classical **character segmentation** for traditional typewritten characters fails for NS images as it assumes clean conditions and particular kinds of connectedness between characters such as the projection profile method implying vertical break lines [Luo et al., 2004]. An exhaustive survey on classical character segmentation into individual components may be found in [Casey & Lecolinet, 1996]. With the recent emergence of NS image analysis, most papers focus on text detection and localisation. When text extraction is considered, main tested images include either clean or complex backgrounds but almost without joined characters. Text on NS images such as road signs, advertisements, has to be large and easy to view with well-spaced characters. Nevertheless, more complex images may be considered with all text present in daily life such as labels on logos, brand names on clothes and so on. As previously mentioned, few papers proposed solutions. Among them, Karatzas and Antanacopoulos [Karatzas & Antanacopoulos, 2004] worked on WWW images with difficult text and suggested a region-based method to extract text followed by a fuzzy proximity measure to add topological properties of character strokes. Chen [Chen, 2003] obtained more individual components by considering text extraction with spatial information by using MRF-based text extraction. Thillou and Gosselin [Thillou & Gosselin, 2004] extracted text with a k-means clustering method and combined textual clusters by paying attention to pixels which connected individual components.

Part of our motivation is to build an efficient text understanding system with lightweight algorithms to fit within mobile devices' resources (such as PDAs) as they will be intensive future users of these systems.

3. Natural Scene Text Extraction

Text extraction is a challenging issue, made even more difficult in a NS context. Classical binarisation algorithms on gray-scale images showed their limitations to handle NS degradations. Colours have to be taken into account and we propose an algorithm that we call *Selective Metric Clustering (SMC)*. We perform a 3-means clustering algorithm using two metrics, the Euclidean distance D_{eucl} and an angle-based similarity S_{cos}, in order to mainly circumvent effects of varying colours, complex backgrounds and uneven lighting.
Several metrics, either distances or similarities, have been designed to be used in k-means in different fields requiring unsupervised classification, such as the Minkowski metric, generalisation of the traditional Euclidean distance, the Canberra distance or the normalised correlation for example. Several other measures exist and the reader is referred to [Plataniotis & Venetsanopoulos, 2000]. Angle-based similarities have been previously used for edge detection or colour segmentation by Wesolkowski [Wesolkowski, 1999] by exploiting the sine of the angle between colour vectors, for colour classification by Hild [Hild, 2004], and for vector directional filtering by Lukac et al. [Lukac et al., 2005].
To include hue information inside the RGB colour space, angle-based similarities may be considered as:

$$Hue = \begin{cases} \theta \text{ if } B < G \\ 2\pi - \theta \text{ otherwise} \end{cases} \qquad \theta = \arccos\left(\frac{1}{2} \frac{(R - G) + (R - B)}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right) \tag{1}$$

Hence, by keeping the same colour space and preventing computationally expensive conversions, hue information may be included with the use of angle-based similarities. Moreover, similar colours have parallel orientations even when degraded with uneven lighting or by shiny material. In natural scene images, (slight) variations are a frequent occurrence within the same object of same colour due to all sources of variations and angle-based similarity may deal with metamers to properly extract text. Finally, an angle-based similarity represents chromaticity difference information whereas the Euclidean distance computes the intensity difference information. Their combination enables one to perform intensity-dependent segmentation directly from the RGB image in areas of different colours, and the other to perform intensity-invariant segmentation in regions of similar but not identical colours.
Based on intensive tests [Mancas-Thillou, 2006], we chose an angle-based similarity S_{cos} equal to Equation 2.

$$Scos = 1 - \left(\frac{xy}{\|x\|.\|y\|} \right) \left(1 - \frac{\|x\| - \|y\|}{\max(\|x\|, \|y\|)} \right) \tag{2}$$

Additionally, intensity is paramount information to distinguish similar pixels of the same colour but different intensities and SMC includes a gray-scale image, thresholded with a traditional global binarisation to build a multi-hypothesis text extraction. Finally, as text is a meaningful object and as the chosen k-means clustering does not integrate spatial information, SMC opts for the proper text extraction by using clues of spatiality.
Figure 4 details steps of the SMC algorithm for text extraction and the following subsections detail each of these steps.

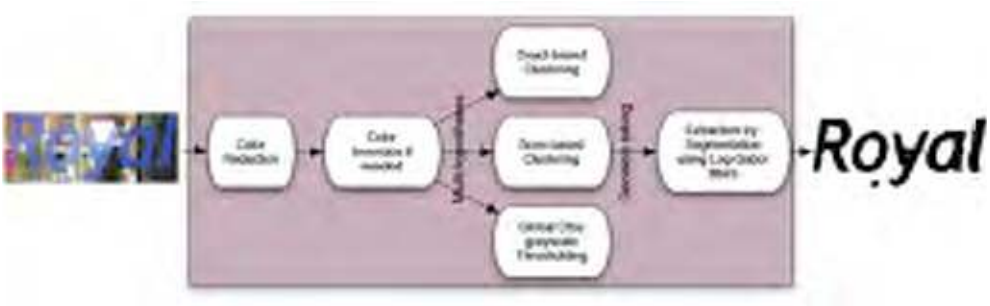

Fig. 4. Steps of the SMC algorithm.

3.1. Utilisation of a multi-hypothesis text extraction

After a colour reduction and colour inversion to always get dark text on bright background, SMC performs two clustering algorithms on the initial image with both metrics, D_{eucl} and S_{cos}. Moreover, to alleviate effects of achromatic images and improve results of text extraction, we add intensity information with the thresholded gray-scale image. For pure achromatic images (meaning R=G=B), S_{cos} cannot build 3 clusters efficiently as all pixels are on the same diagonal in the RGB cube. The same phenomenon appears for non-pure achromatic images where it is rather difficult to separate colours efficiently. This drawback is also true in hue-based colour spaces where hue is even not defined! We obtain also three possible text extraction results of both metrics and the binarised gray-scale image.

K-means clustering applied on NS colour images with two metrics forms 3 clusters for each one and one cluster is obviously a part of the background, another one is a part of the text and the third one is either text or background. For sharper results and hence better character recognition, it may be interesting to combine both textual clusters. First of all, the background colour is selected very easily and efficiently as being the colour with the biggest rate of occurrences on the image edges. Next, we propose a new text validation measure R to find the most textual foreground cluster over the two remaining clusters. Based on properties of connected components of clusters, spatial information is already added at this point to find the main textual cluster. R is based on the largest regularity of connected components of text compared to those of noise and background and is defined in Equation3.

$$R = \sum_{i}^{N} \left| \text{area}(i) - \frac{1}{N} \left(\sum_{i}^{N} \text{area}(i) \right) \right| \tag{3}$$

where N is the number of connected components and area(i) refers to the area of component i. This measure enables the computation of the variation in candidate areas. The main textual cluster is identified as the one having the smallest R. If the third unknown cluster belongs to text, both textual clusters need to be merged. A new computation of R is performed considering the merging of both clusters. If R decreases, the fusion is processed. This method enables the merging of text of different colours in the same word for instance as regularity becomes better.

With this multi-hypothesis text extraction, we may handle a very large range of NS images. The use of S_{cos} is preponderant, as illustrated in Figure 5 with some complex NS images which can not be better handled in a k-means framework. Some comparisons were done

with the Euclidean distance and by increasing the number of clusters or with other colour spaces [Mancas-Thillou, 2006]. Angle-based similarities can extract text of very challenging NS images without additional effort and by keeping versatility for other NS images.

Fig. 5. Extraction results using SMC in a RGB-based k-means framework.

3.2. Extraction-by-segmentation

After computation of k-means with two different metrics, the choice between the three text extraction methods has to be done. A multi-hypothesis method has been shown by Chen [Chen, 2003] by varying the number of clusters in a GMM-based clustering and choosing the right segmentation with the final step of recognition. One drawback to this method is to keep several segmentations to process during subsequent steps and to increase the number of text areas to recognise. Moreover, recognition is logically an efficient step to choose the right segmentation, but in complex NS images, character segmentation or even denoising steps must be added, and no decision could be done before the final step of recognition; otherwise, recognition results may be erroneously considered bad. In SMC, we choose to intermingle consecutive steps to avoid this disadvantage and to add as much information as possible.

Colour information is a very consistent clue for NS images. However the segmentation process, previously described in this section, does not make use of spatial information, which is quite necessary for object-driven segmentation and specifically text extraction. In order to extract characters properly, we exploit the same tool for character segmentation, detailed in depth in Section 4. We need to have spatial information to locate characters in the image, as well as needing the frequency information to use illumination variation to detect character edges. Hence, log-Gabor filters proposed by Field [Field, 1987] are chosen for decision making, because they particularly fit well to NS images.

One important parameter for log-Gabor filters is the filter frequency. As we used them to enhance characters in a gray-scale image, we choose a frequency equal to the inverse of the rough thickness of characters, determined by the number of pixels of the extracted result and its skeleton. A simple ratio between these two latter values is computed and the inverse is the frequency of log-Gabor filters. Results of log-Gabor filters present globally high responses to characters with this set frequency. Hence in order to efficiently choose the best extracted text result, we perform an average of pixel values. The segmentation having the highest average is chosen as the final segmentation.

3.3. SMC evaluation and results

Table 1 details results for the three hypotheses (two clustering and global binarisation) on the public database ICDAR 2003 [ICDAR Competition, 2003], which includes 2268 natural scene words. Results are expressed in terms of Precision, Recall and F-scores defined in Equation 4. F-score is the weighted harmonic average of Precision and Recall in order to more easily compare results.

$$\text{Precision} = \frac{\text{Correctly extracted characters}}{\text{Total extracted characters}},\ \text{Recall} = \frac{\text{Correctly extracted characters}}{\text{Total number of characters}} \quad (4)$$
$$\text{F-score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Recall} + \text{Precision}}$$

Extraction	Precision	Recall	F-score
D_{eucl}	0.90	0.88	0.89
S_{cos}	0.93	0.36	0.52
Binarised gray-scale image	0.88	0.76	0.82

Table 1. Precision, Recall and F-score measures of text extraction performed by the three extraction hypotheses.

To add more arguments to complementarities between these three extracted results, D_{eucl} performs better in 5% images, while S_{cos} in 12% and the global thresholding in 9%. There is a larger overlap between D_{eucl} and the global thresholding which performs quite equally in 69% images.

To choose the right text extraction, we opt for log-Gabor filters by adding spatial information. In [Mancas-Thillou & Gosselin, 2006], we compared the performance of this method with the Silhouette technique, a measure of how well clusters are separated, to choose between the two metrics only. It can be logical to think that best text extraction results present the best separation between clusters. However, it is not always true because Silhouette performs well in 77.7% images and our proposed method using spatial information performs well in 93.2%, yielding an improvement of 19.9%.

A few works deal with NS text extraction and we compare SMC, firstly, with solutions of Wolf et al. [Wolf et al., 2002] which designed an extended method of Sauvola and Pietikäinen [Sauvola & Pietikäinen, 2000] to extract text from NS images or videos, and then, with solutions of Garcia and Apostolidis [Garcia & Apostolidis, 2000] which used a k-means clustering in the HSV space with the Euclidean distance only. Combination of clusters in this last method has not been implemented and a perfect combination is assumed while our method is tested including our combination method. Results are presented in terms of Precision, Recall and F-score in Table 2.

Methods	Precision	Recall	F-score
Wolf et al.	0.35	0.19	0.25
Garcia and Apostolidis	0.66	0.57	0.61
SMC	0.95	0.91	0.93

Table 2. Comparison of Precision, Recall and F-score measures between Wolf's method, Garcia and Apostolidis's method and our SMC method.

The combination of two metrics in a clustering framework and a global thresholding has proven its efficiency compared to two recent and competing algorithms. Finally, due to the explosion of use of camera phones or digital cameras and huge amount of images to process for text extraction, the algorithm needs to be relatively fast in order to provide satisfying results for frequent use. Our text extraction algorithm runs in 0.61 seconds on average for our databases on a PC with a Pentium M-1.7 GHz micro-processor. The source code for text extraction was developed in C language but could be optimised further.

4. Unit-based Segmentation

This section deals with segmentation of text areas into specific units, such as lines, words and characters. In commercial OCR systems, this process is usually included and is quite successful except for severely degraded characters, strongly broken or tightly connected ones where recognition rates drastically drop. Incorrect segmentations due to perspective, for example, may even lead to no recognition at all. Usually, NS text, handled in literature, is well separated due to their reading goal. However, complex NS images with low resolution, perspective or wavy surfaces present challenges and unit-based segmentation has recently become a point-of-interest to circumvent recognition errors. Hence, we describe a fast and simple line and word segmentation method and an innovative and robust character segmentation method using log-Gabor filters.

4.1. Line segmentation

NS images may present several words but usually only a few lines if we cite street names or book titles. Nevertheless, colourful magazine headlines or abstracts on book covers or even camera-based documents such as restaurant menus may have several lines. Line segmentation are usually not considered as difficult for NS images but present interesting challenges for skewed text areas; as such we present very fast and intuitive algorithms.

Segmentation into lines is an old topic and the two main and successful methods are either the vertical projection profile or the Hough transform. The first one is a histogram of the number of text pixels accumulated along text lines and projected vertically. The projection profile has maximum-height peaks for text and valleys for inter-line spacing. It is quite sensitive to noise and skewed lines. The second method maps each point in the original (x,y) plane to all points in the (r, θ) Hough plane of possible lines through (x,y) plane with slope θ and distance from origin r. This method performs well on skewed text and may also simultaneously deskew it with the knowledge of θ value but it is on the other side computationally quite expensive.

Connected components coming from our text extraction step to perform the deviation measure R are already computed with general properties, such as height of characters h_{char}. On the bounding box of the text area, we define the approximate number of lines N_l by:

$$N_l = \text{floor}(1 + \frac{h_{text} - \mu(h_{char})/2}{\mu(h_{char}) * 3/2}) \quad (5)$$

where h_{text} is the height of the text area, $\mu(h_{char})$ is the average of h_{char} on all characters and floor(x) is the largest integral value less or equal to x. All y-coordinates of character centroids are then clustered with the k-means algorithm, k being equal to N_l segmentation.
For strongly skewed lines, a fast deskewing is required based on the height of the text bounding box. The first text pixel of the first row of the tightest bounding box is detected and if its position is before the middle of the image width, the skew angle is negative; otherwise it is positive. A first rotation of 1° is computed in the determined direction. If the bounding box is shorter in height than the previous one, successive rotations are performed until the bounding box becomes higher meaning that the skew angle was larger than 1°.

4.2. Word segmentation

Word segmentation, contrarily to line segmentation useful for better character recognition, is a crucial step for text understanding after recognition, such as by speech synthesis. A natural linguistic parser is always part of a text-to-speech algorithm and it is important to identify words for a proper pronunciation as explained in the example:
Ex: in French, the phonetic transcription can be different, depending on word segmentations:

« les tas » → [l e t a] and « lestas » → [l ε s t a]

In Latin alphabets, the inter-words distance D_{IW} is larger than the one of inter-characters D_{IC}. We compute word segmentation by identifying word separations by all distances superior to std(D_{IC}) + mean(D_{IC}) with std(.) and mean(.), respectively standard deviation and mean of inter-character distances in a given line. This step occurs after the refined character segmentation in order to have more correct calculations based on characters and spaces between characters.
For this step, we use a simple statistic method. Some errors may occur when a few words are present with distances between words varying due to different fonts or perspective. Nevertheless, this algorithm is robust when run against text areas presenting only one word, which is quite frequent in NS images or after text detection algorithms, which usually oversegment lines. Finally, this rule basically bends to oversegmentation more than subsegmentation, which may be more easily handled by our recognition and correction we proposed in [Mancas-Thillou, 2006].

4.3. Character segmentation using log-Gabor Filters

The first character segmentation algorithms, developed for typewritten characters, appeared more than forty years ago to separate each character individually, in order to subsequently feed into OCR. Later, these techniques have been extended to segmentation of cursive writing for handwritten text. Main techniques for typewritten characters are categorised into three groups. *Image-based methods* are mainly issued from projection analysis or the

"Caliper" distance, which is the distance between the uppermost and bottommost pixels in each column meaning that smallest distances are tentative segmentation places, as experienced in camera-based document processing [Thillou et al, 2005]. These methods imply vertical separation only, which is not convenient at all for strongly joined characters or skewed and italic ones where parts of a character infringe on the space occupied by the next one. *Recognition-based methods* use a sliding window of variable width to provide sequences of hypothetical segmentation locations which are confirmed or refuted by character recognition. These techniques also give only vertical separations and need robust OCR to reject or accept all possible segmentations, which are quite numerous, even for a single word! *Hybrid methods* mainly encompass oversegmentation methods. A word is dissected into its smallest possible components and recognition is based on these units to individually recompose the characters one at a time. They are particularly well suited for joined and broken characters and segmentation results are not only vertical as based on small components. Nevertheless, oversegmentation techniques need a dedicated recogniser based on unit features.

NS images need robust character segmentation since not all aforementioned methods are suitable, and off-the-shelf OCR using them lead to too many recognition errors. A gap between complex NS images and character recognition has to be filled to extend applications and use of NS images in daily life. A NS character segmenter is needed to increase NS character recognition and has to be robust against already individual characters, broken and joined ones and against unknown fonts, italic characters or with perspective. A very innovative solution, using log-Gabor filters and the recognition step that follows in a hybrid method, is fundamentally different from existing ones, and is presented after focusing on properties of these filters

- **Why are log-Gabor filters appropriate for NS character segmentation?**

Character segmentation in NS images obviously needs text properties and gray-level information to complement the colour information exploited in text extraction. Hence simultaneous spatial and directional information (for character separation location) and frequency information (gray-level variation to detect cuts) are required. Gabor filters are a traditional choice to address this issue: they are cosine-like filters having a given direction and modulated by a Gaussian window. They have been extensively used to characterise texture, and more specifically in our context, to detect and localise text into an image. In this aim, Gabor filters are quite time consuming because several directions and frequencies must be used to handle the variability in character sizes and orientations. Moreover, Gabor filters present limitations: large bandwidth filters induce a significant continuous component and only a maximum bandwidth of 1 octave could be designed. Field [Field, 1987] proposed an alternative function called log-Gabor which lets us choose a larger bandwidth without producing a continuous component. Moreover, he suggested that natural images are better coded by filters that have a Gaussian transfer function on a logarithmic frequency scale, by showing that their spectrum statistically falls off at approximately 1/f, which corresponds well to where the log-Gabor filter spectrum falls off on a linear scale. Figure 6 displays the shape of log-Gabor functions at the same frequency but with bandwidth varying from 2 to 8 octaves. Log-Gabor functions have the same appearance as Gabor functions for bandwidths less than one octave. The possibility of sharpening the filters is highlighted.

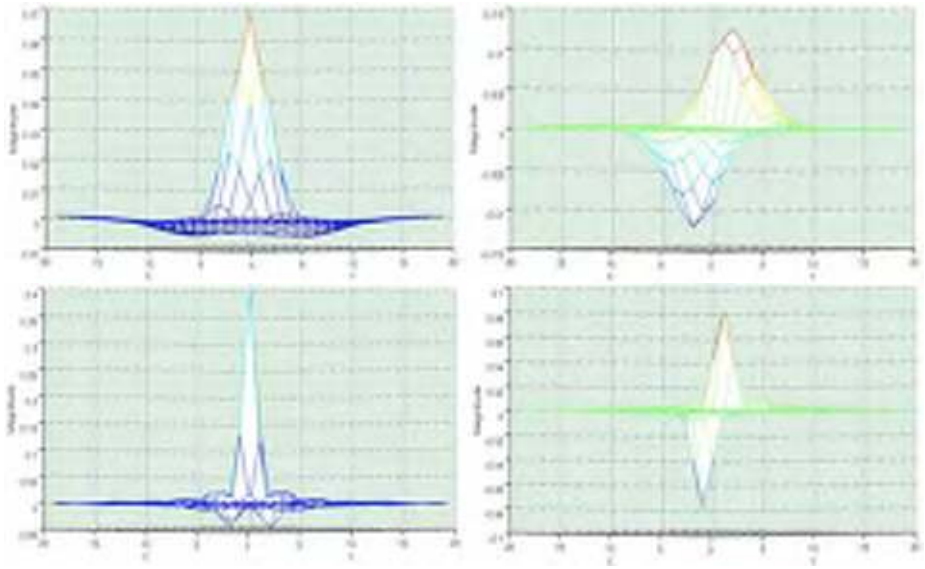

Fig. 6. From top to bottom: even (left) and odd (right) log-Gabor filters with a bandwidth of 2 octaves and even (left) and odd (right) log-Gabor filters with a bandwidth of 8 octaves. In the spatial domain, the possibility of sharpening the filters is highlighted.

Log-Gabor filters in the frequency domain can be defined in polar coordinates by $H(f, \theta) = H_f * H_\theta$ where H_f is the radial component and H_θ, the angular one:

$$H(f,\theta) = \exp\left\{\frac{-\left[\ln(f / f_0)\right]^2}{2\left[\ln(\sigma_f / f_0)\right]^2}\right\} * \exp\left\{\frac{-(\theta-\theta_0)^2}{2\sigma_\theta^2}\right\} \quad (6)$$

where f_0 is the central frequency, θ_0 is the filter direction, σ_f is the standard deviation of the radial components of the Gaussian describing the filter and is used to define the radial bandwidth and σ_θ is the standard deviation of the angular part of the Gaussian and enables the definition of the angular bandwidth. As we are looking for vertical separation between characters, we only use two directions for the filter: the horizontal and the vertical ones. Hence, for each directional filter, we have a fixed angular bandwidth of $\pi/2$, which determines σ_θ. Log-Gabor filters are not really strict with directions and defining only two directions enables the handling of italic and/or misaligned characters. For highly misaligned characters, the number of directions could be increased to handle this additional degradation, but it is important to mention that the angular bandwidth will become narrower and hence more selective.

Only two parameters remain to be defined, f_0 and σ_f, which are used to compute the radial bandwidth. The central frequency f_0 is used to handle gray level variations to detect separation between characters. The spatial extent of characters is their thickness that we consider as the wavelength of "characters", hence it is logical to get a central frequency close to the inverse of the thickness of characters to get those variations. However, the measurement of character thickness may not be very accurate depending on the presence of

degradations. In order to handle all kinds of degradations, we compensate for inaccurate thickness estimation with the second parameter σ_f. If the thickness of characters is not consistent inside a character, some character parts can be removed permanently. In this case, by increasing the bandwidth, we can support the variability in the thickness of characters with a "larger" filter. Moreover, sometimes with very degraded or close characters, the thickness is very difficult to estimate and the filter must be very sharp to get each small variation in the gray level values such as in Figure 7, with a complex NS image.

Fig. 7. Impact of varying log-Gabor bandwidth for character segmentation. Original image (top left), binary version (top right), segmentation with large bandwidth (bottom left), segmentation with narrow bandwidth (bottom right).

As degradations and conditions of frequency estimation are quite unexpected, we chose the bandwidth filter in a dynamic way using recognition results. In the following part, we detail our method and how each parameter is estimated.

- **Character segmentation-by-recognition**

Based on the binarisation of the detected area, which is available with the proposed SMC algorithm, the character segmentation may now be performed on gray-level images. To define frequency, a classical way is to use a "wavelet-like" method. This means trying out several frequencies to get a good result for one of them. This method is time consuming due to several convolutions with multiple frequency filters and the number of computations rose to the power of two with the second parameter. Text embedded in natural scene images presents a quite consistent wavelength, which is very different from the background. For our filter, we decided to use a wavelength related to the average of the character thicknesses. This is computed by using the ratio between the number of pixels of the first mask obtained by the SMC method and its skeleton.

Due to the large variation in NS character fonts and sizes, the bandwidth has to be chosen dynamically. As objects to be segmented are text, we can use segmentation-by-recognition to choose the convenient bandwidth. We fix the initial and final values for the bandwidth estimation. From approximately 2 octaves to approximately 8 octaves, which makes σ_f/f_0 vary with a step of 0.1 (from 0.1 to 0.6), we process six filters and provide the result to an OCR engine.

The result is composed of the vertical filter only as the character separation is mainly vertical. Moreover, in the output, only the phase of the filter will be exploited. As the text and background information have different wavelengths, the phase contains much more information than magnitude, as displayed in Figure 8. Moreover, local variation issued from

the initial separation between characters induces a phase difference. The latter one contains the gray-level information while the phase shows a local map which makes a good separation between the background and the textual information; this intermediate result is then multiplied by the first mask from text extraction to remove possible noise around characters as displayed in Figure 9.

Fig. 8. Log-Gabor filtering results for each filter property. From left to right: phase of the horizontal filter, phase of the vertical filter, magnitude of the vertical filter and absolute phase of the vertical filter.

As shown in Figure 9 after filter convolution, characters have mainly low intensities and higher background intensities. In order to remove spurious parts between characters and to remain parameter-free, we use a global Otsu thresholding [Otsu, 1979], which automatically chooses the threshold to minimise the intra-class variance of the thresholded black and white pixels. With the use of the absolute phase of the vertical filter, only one threshold needs to be determined. After this step, we get a result, such as the one shown at the bottom of Figure 9, to choose the bandwidth for filters.

Fig. 9. Phase of the vertical filter multiplied by the mask issued from the text extraction (left) and result after global thresholding (middle). Improvement is obvious from the binary version (right).

We use a home-made OCR algorithm composed of a multi-layer perceptron with geometrical features to recognise characters, which is trained by a separate data set and is used to assess how well characters are segmented. Detailed explanations about this in-house OCR are provided in Section 5. After applying log-Gabor filters, connected components (mostly characters) are given as inputs to OCR. Recognition rates for each character or assumed character are averaged and the maximum score enables the choice of the bandwidth. This estimation needs six straightforward filters with only one frequency which enables the use of log-Gabor filters for character segmentation in a low-resource context.

Some examples are given in Figure 10 to appreciate performance of this proposed character segmentation based on log-Gabor filters. From top, the third example is composed of severely joined characters and the result after segmentation is very satisfying. Between 'i' and 'n' of the word 'smokin', the connection is still present but the recognition is now successful even with off-the-shelf OCR including traditional segmentation. The last example illustrates an original image with characters of two different major colours (yellow and white) and a yellow and blue background. Based on our combination of clusters, the 'M' of the word 'Memorex' has been reconstituted but simultaneously with some parts of background. Nevertheless, the yellow background information has a different intensity and frequency than the 'x' character, leading to a successful segmentation.

Even if in NS images, broken characters are rare due to the relatively large thickness of characters whose aim is to be read, it may be useful to have solutions for handling them. To recompose parts of a single character, we proposed in [Mancas-Thillou et al., 2005] an

algorithm using log-Gabor filters as well. It enables the correction of already broken characters (particular fonts or text extraction errors) and new broken characters due to recognition failures. The bandwidth is fixed and the frequency estimation is refined by an iterative log-Gabor convolution.

Fig. 10. Some character segmentation examples. From left to right: original image, SMC-based binary version and result after character segmentation.

The convolution of text extraction results with log-Gabor filters has several goals: to choose the better extracted text, to segment characters into individual parts and also to fuse broken characters by validating or not previous outputs. Log-Gabor filters give a large set of applications in NS images with a large modularity and very satisfying results as detailed in the following subsection.

4.4. Character segmentation evaluation

In Table 3, comparisons are done between the behavior of an efficient commercial OCR (ABBYY FineReader 8.0 Professional Edition Try&Buy) against initial images without any processing, after the SMC-based text extraction without character segmentation, after a classical "Caliper" distance-based segmentation and after the log-Gabor-based segmentation -by-recognition to show the efficiency and necessity of this latter method to improve recognition results. Error rates are computed using the Levenshtein distance between the ground truth and the resulting text. The Levenshtein distance between two strings is given by the minimum number of operations needed to transform one string into the other, where an operation is an insertion, deletion, or substitution of a single character. Equal weights for each operation are employed in our computation. Error rates are then computed by dividing with the number of characters. By using the Levenshtein distance, some error rates for a word may be superior to 1, but it is useful to penalise broken characters. Tests have been computed on 10% of the database due to the impossible automatic processing with a commercial OCR. To compute log-Gabor filtering, we use the Kovesi' toolbox [Kovesi, 2006] in Matlab. The home-made OCR, which is useful to choose the right bandwidth, has been extended in C language from a version of Gosselin [Gosselin, 1996]. The "Caliper" distance and evaluation measures have been developed in Matlab.

Error rates	Colour images	SMC-based images	"SMC-based+ Caliper" images	"SMC-based +Log-Gabor" images
ICDAR2003 database	71%	40%	43%	19%

Table 3. Usefulness of character segmentation in natural scene images stated from recognition error rates with a commercial OCR.

For the ICDAR2003 database, "Caliper"-based segmentation even gives worse results than without segmentation. It is mainly due to the number of broken characters which increases. Log-Gabor segmentation drastically decreases error rates.
In this proposed character segmentation, the bandwidth is estimated with the recognition step and we compute the efficiency rate of this decision. Some erroneous choices could be made due to our majority vote on the whole text and the decision is correctly taken in 98.1% of images. Errors are mostly avoided with this character segmentation-by-recognition as each decision is checked with other steps dynamically. Main errors are either due to the OCR engine with much degraded characters or to the presence of thin characters. As log-Gabor filters exploit intensity information to accurately segment characters into individual components, if characters are too thin, they will be easy to break in several pieces of characters, leading to erroneous recognition.
Some deeper comparisons [Mancas-Thillou, 2006] have been done with a recent method from Gatos et al. [Gatos et al., 2005], who used the same public database. Their text extraction is based on a gray-scale adaptive thresholding and they proposed to recombine characters components based on several rules to avoid too many joined characters. We use the same evaluation method being the Levenshtein distance. Improvement may be observed with an error rate decreasing of around 43%.

5. Natural Scene Character Recognition

From text extraction to unit-based character segmentation, the main goal was to improve extracted text in order to finally increase recognition rates. Hence, in this section, the objective is to provide high-quality extracted text in order to exploit off-the-shelf OCR. Nevertheless, NS character recognition, faced with the very large diversity of images without any a priori information, needs suitable conditions to work properly, such as a huge and representative training database.

5.1. Considerations on character recognition

To focus on NS character recognition, main recent papers deal with gray-level characters to handle degradations and low resolution of acquisition. The idea is therefore to build efficient recognisers against some issues without improving characters beforehand. For WWW images, Zhou et al. [Zhou et al., 1997], first extracted characters by colour clustering and then converted the characters' colours into gray-scale. The main colour receives the value of 255 and the other ones are set to differences from the representative colour. The character shape is then treated as a 3D surface and a polynomial surface fitting method (Legendre polynomial basis) is used as feature extractor and a basic character-to-class Euclidean distance is used to recognise characters. For NS text, Zhang et al. [Zhang et al., 2002] exploited also gray-scale images after intensity normalisation with Gabor-based

features in the context of Chinese sign recognition. They performed feature selection with a linear discriminate analysis to build a space as discriminate as possible. Finally the classification is solved with kNN.
To perform segmentation-by-recognition in Section 4, we use an extended version of classifier from Gosselin [Gosselin, 1996], based on geometrical features and a multi-layer perceptron (MLP). In order to recognise many variations of the same character, features need to be robust against noise, distortions, style variation, translation, rotation or shear. Invariants are features which have approximately the same value for samples of the same character, deformed or not. To be as invariant as possible, our input-characters are normalised into an N*N size with N=16. However, not all variations among characters such as noise or degradations can be modelled by invariants, and the database used to train the neural network must have different variations of a same character.
In our experiments, we use a feature extraction based on contour profiles. The feature vector is based on the edges of characters and a probe is sent in each direction (horizontal, vertical and diagonal) and to get the information of holes like in the 'B' character, some interior probes are sent from the center. Moreover, another feature is added: the ratio between original height and original width in order to very easily discriminate an 'i' from an 'm'.
Experimentally, in order to lead to high recognition rates, we complete this feature set with Tchebychev moments, which are orthogonal moments. Moment functions of a 2D image are used as descriptors of shape. They are invariant with respect to scale, translation and rotation. According to [Mukundan et al., 2001], we use Tchebychev moments of order 2 for their robustness to noise.
No feature selection is defined and the feature set is a vector of 63 values provided to an MLP with one hidden layer of 120 neurons and an output layer of size 36 for each Latin letter and digit. Due to few training samples for capital letters, uppercase and lowercase letters were initially grouped into the same class. Nevertheless, with the algorithm of increasing database described in the next paragraph, an output layer of 62 neurons may be considered efficiently. The total number of training samples is 40614 divided into 80% for training only and 20% for cross-validation purpose in order to avoid overtraining.

5.2. Zoom on training database: how to build a relevant and general one?

Traditional database increasers are based on geometrical deformations such as affine transformations or on the reproduction of a degradation model such as [Sun et al., 2004] to mimic low resolution. In NS images, the very large diversity must be handled and character extraction of a huge data set is awkward and difficult to achieve. Hence, we increase the NS database with the image analogies of Hertzmann et al. [Hertzmann et al., 2001], with the particular algorithm of texture-by-numbers. Given a pair of images A and A', with A' being the binarised version of A, the textured image in our algorithm, and B' the black and white image to transfer texture, the texture-by-numbers algorithm applies texture of A into B' to create B. Binary versions are composed of pixels having values of 0 or 1; texture of A corresponding to areas of 0 of A' will be transferred to areas of 0 of B' and similarly for 1. Multiscale representations through Gaussian pyramids are computed for A, A' and B' and at each level, statistics for every pixel in the target pair (B, B') are compared to every pixel in the source pair (A, A') and the best match is found.
One sample used to increase the training database is displayed in Figure 11, which also schematises the concept of image analogies.

Fig. 11. Principe of image analogies in the context of database increase: A represents the textured and segmented character, A' its binary version. From a binary version of an 'm' in B', the texture is transferred onto B, similar to the analogy between A and A'.

The entire process of increasing database is firstly based on character extraction from a given data set, using SMC algorithm of Section 3. Characters are hence binarised and normalised. Deformations on character thickness, slant, rotation, and perspective are then performed and the texture-by-numbers is applied on each binary image. A huge and new data set is hence built. To provide standardised characters, all newly-textured characters are then binarised always using our SMC algorithm, leading to realistic degradations of NS images, which enables to increase the database as naturally as possible. Based on the finite steps of variation for each of the pre-cited parameters, for one extracted character and one given texture, 33480 samples may be created. Hence, the power of increasing database of this method is very large (almost infinite depending on the parameter variation and the number of textures). Some tests have been done on recognition and rates are slightly increased. Extensive studies are needed to know if the increase is due to the enlarging database and/or the representativeness of the database with texture transfer. Nevertheless, this technique enables the growing of a database in a fast and reliable way.
Finally, character recognition alone is hardly error-free and linguistic information needs to be added to correct errors for which we build a light and modular solution. For this purpose, we intermingle steps of recognition and correction in order not to consider OCR as a "black box".

6. Conclusion and Future Works

This last section aims at concluding this chapter by summing up main steps in the first part to highlight important points according to us to realize an efficient and versatile NS text understanding and the second parts emphasizes interesting work prolongations in other image processing fields and the focus to give in next years.
Our SMC algorithm has been proposed based on a multi-hypothesis text extraction by selecting either the right clustering metric or the dual information between colour and illumination, using log-Gabor filters. Several points have been detailed such as the superiority of metrics over colour spaces in a clustering framework inside a general NS context. Angle-based similarities have overcome any other colour spaces to handle complex NS images, meaning mainly images with complex backgrounds and uneven lighting. Moreover, complementarities between the Euclidean distance and angle-based similarities in a k-means method to handle a very large set of NS images have also been described. Spatial and luminance information have been added to choose the best text extraction to provide to recognition. To circumvent NS challenges, text extraction was

intermingled with the subsequent step of character segmentation and very encouraging results have been shown in terms of Precision, Recall and F-score, comparison with other state-of-the-art algorithms, and while keeping a reasonable computation time.
Our selective metric-based clustering is aimed at being versatile and results we have provided show that it is. Nevertheless, SMC mainly uses colour information and one drawback of our system is for natural scene images having embossed characters. In this case, the foreground and background have the same colour imparting partial shadows around characters due to the relief but not enough to discriminately separate the textual foreground from the background as displayed in Figure 12. Gray-level information with the simultaneous use of a priori information on shadows and character properties could be a solution to handle these cases. Nevertheless, it may be relevant to note that a robust OCR may also give satisfying results without any modifications of our algorithm.

Fig. 12. Error example of our selective metric-based clustering: initial colour embossed image on left and the SMC result on right.

In a second step, we propose NS character segmentation-by-recognition based on log-Gabor filters whose some parameters are defined dynamically. This algorithm fulfils initial requirements and gives interesting results under various aspects:

- No assumption on characters fonts, sizes or skew is done
- Characters are segmented with not only vertical separations but cuts following the character profile, leading to increased recognition rates
- Touching and broken characters are handled
- The algorithm is made more robust by using additional information with the consecutive step of character recognition
- Satisfying results in terms of recognition rates and Levenshtein distance.

To conclude, log-Gabor filters are very modular and efficient tools to segment NS characters into individual and understandable components.
Among future works of each step detailed in the previous paragraphs, one of the main prolongation works will be to extend some of these solutions for extraction of other objects in natural scene images to show once again versatility of these methods. Obviously, character segmentation is a dedicated step of text analysis. Nevertheless, our combination of colour, intensity and spatial information or handling of low resolution frames may lead to interesting results for other applications.

About the global system and if resources are available, the small amount of errors at each step may be decreased by keeping information until recognition. These additional hypotheses will be handled through another step of information fusion.
Due to the great expansion of electronic goods and their ever increasing performance, readers may wonder if these chapter topics will not be obsolete in a few years. In some recently launched smartphones in Asia with 3.2 Megapixels cameras and rudimentary embedded OCR or with expansion to 8 Megapixels of consumer-grade digital cameras, text extraction part handling complex backgrounds and uneven lighting will be necessary for a long time: professional expensive cameras have still problems with illumination by nature and complex backgrounds, especially in advertisements. Such issues will not disappear anytime! Unit-based segmentation may be removed by other computationally very demanding methods but character recognition is mandatory to understand text. Hopefully, text understanding steps will be automatically embedded into handheld imaging devices soon for exciting and useful applications in daily life!

7. References

Berkhin, P. (2002). Survey of clustering data mining techniques, *Tech. report*, Accrue Software

Casey, R.G. & Lecolinet, E. (1996). A survey of methods and strategies in character segmentation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 18, No. 7, pp. 690-706

Chen, D. (2003). *Text detection and recognition in images and video sequences*, PhD thesis, Ecole Polytechnique Fédérale de Lausanne

Comaniciu, D. (2000). *Nonparametric robust methods for computer vision,* PhD thesis, Rutgers University

Du, Y.; Chang, C-I. & Thouin, P.D. (2004). Unsupervised approach to colour video thresholding, *Optical Engineering*, Vol. 43, No. 2, pp. 282-289

Esaki, N.; Bulacu, M. & Shomaker, L. (2004). Text detection from natural scene images: towards a system for visually impaired persons, *Proceedings of Int. Conf. Pattern Recognition*, pp. 683-686

Field, D.J. (1987). Relations between the statistics of natural images and the response properties of cortical cells, *Jour. Opt. Soc. Amer. A*, Vol. 4, No. 12, pp. 2379-2394

Garcia, C. & Apostolidis, X. (2000). Text detection and segmentation in complex colour images, *Proceedings of Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 2326-2330

Gatos, B.; Pratikakis, I. & Perantonis, S.J. (2005). Towards text recognition in natural scene images, *Proceedings of Int. Conf. Automation and Technology*, pp. 354-359

Gllavata, J.; Ewerth, R. & Freisleben B. (2003). Finding text in images via local thresholding, *Proceedings of IEEE Symposium on Signal Processing and Information Technology*, pp. 539-542

Gosselin, B. (1996). Application de réseaux de neurones artificiels à la reconnaissance automatique de caractères manuscrits, PhD thesis, Faculté Polytechnique de Mons

Hamza, H.; Smigiel, E. & Belaid, A. (2005). Neural based binarisation techniques, *Proceedings of Int. Conf Document Analysis and Recognition*, pp. 317-321

Hertzmann, A.; Jacobs, C.E.; Oliver, N.; Curless, B. & Salesin, D.H. (2001). Image analogies, *Proceedings of ACM SIGGRAPH, Int. Conf. On Computer Graphics and Interactive Techniques*

Hild, M. (2004). Colour similarity measures for efficient colour classification, *Jour. of Imaging Science and Technology*, Vol. 15, No. 6, pp. 529-547

ICDAR Competition (2003). http://algoval.essex.ac.uk/icdar

Jung, K.; Kim, K.I. & Jain, A.K. (2004). Text information extraction in images and video: a survey, *Pattern Recognition*, Vol. 37, No. 5, pp. 977-997

Karatzas, D. & Antonacopoulos, A. (2004). Text extraction from web images based on a split-and-merge segmentation method using colour perception, *Proceedings of Int. Conf. Pattern Recognition*, Vol. 2, pp. 634-637

Kim, I.J. (2005). Keynote presentation of camera-based document analysis and recognition, http://www.m.cs.osakafu-u.ac.jp/cbdar

Kim, J.; Park, S. & Kim, S. (2005). Text locating from natural scene images using image intensities, *Proceedings of Int. Conf Document Analysis and Recognition*, pp. 655-659

Kovesi, P.D. (2006). MATLAB and Octave functions for computer vision and image processing, School of Computer Science & Software Engineering, The University of Western Australia, http://www.csse.uwa.edu.au/~pk/research/matlabfns/

Li, H. & Doermann D. (1999). Text enhancement in digital video using multiple frame integration, *Proceedings of ACM Int. Conf. on Multimedia*, pp. 19-22

Liang, J.; Doermann, D. & Li, H. (2003). Camera-based analysis of text and documents: a survey, *Int. Journal on Document Analysis and Recognition*, Vol. 7, No. 2-3, pp. 84-104

Lienhart, R. & Wernicke, A. (2002). Localising and segmenting text in images, videos and web Pages, *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 12, No. 4, pp. 256-268

Lopresti, D. & Zhou, J. (2000). Locating and recognising text in WWW images, *Information Retrieval*, Vol. 2, pp. 177-206

Lukac, R.; Smolka, B.; Martin, K.; Plataniotis, K.N. & Venetsanopoulos, A.N. (2005). Vector filtering for color imaging, IEEE Signal Processing, Special Issue on Color Image Processing, Vol. 22, No. 1, pp. 74-86

Luo, X.-P.; Li, J. & Zhen, L.-X. (2004). Design and implementation of a card reader based on build-in camera, *Proceedings of Int. Conf. Pattern Recognition*, pp. 417-420

Mancas-Thillou, C. (2006). *Natural scene text understanding*, PhD thesis, Faculté Polytechnique de Mons, Belgium

Mancas-Thillou, C. & Gosselin, B. (2006). Spatial and color spaces combination for natural scene text extraction, *Proceedings of Int. Conf. Image Processing*

Mancas-Thillou, C.; Mancas, M. & Gosselin, B. (2005). Camera-based degraded character segmentation into individual components, *Proceedings of Int. Conf Document Analysis and Recognition,* pp. 755-759

Mata, M.; Armingol, J.M.; Escalera, A. & Salichs, M.A. (2001). A visual landmark recognition system for topologic navigation of mobile robots, *Proceedings of Int. Conf. on Robotics and Automation*, pp. 1124-1129

Messelodi, S. & Modena, C.M. (1992). Automatic identification and skew estimation of text lines in real scene images, *Pattern Recognition*, Vol. 32, No. 5, pp. 791-810

Mukundan, R.; Ong, S.H. & Lee, P.A. (2001). Discrete vs. continuous orthogonal moments in image analysis, *Proceedings of Int. Conf. On Imaging Systems, Science and Technology*, pp. 23-29

Niblack, W. (1986). *An introduction to image processing*, Prentice-Hall, pp. 115-116

Ojima, Y.; Kirigaya, S. & Wakahara, T. (2005). Determining optimal filters for binarisation of degraded gray-scale characters using genetic algorithms, *Proceedings of Int. Conf Document Analysis and Recognition*, pp. 555-559

Otsu, N. (1979). A threshold selection method from gray level histograms, *IEEE Trans. System, Man and Cybernetics*, Vol. 9, No. 1, pp. 62-66, 1979

Perroud, T.; Sobottka, K.; Bunke, H. & Hall, L. (2001). Text extraction from colour documents - clustering approaches in three and four dimensions -, *Proceedings of Int. Conf Document Analysis and Recognition*, pp. 937-941

Plataniotis, K.N. & Venetsanopoulos, A.N. (2000). *Colour image processing and applications*, Springer Verlag

Sauvola, J. & Pietikainen, M. (2000). Adaptive document image binarisation, *Pattern Recognition*, Vol. 33, pp. 225-236

Sobottka, K.; Bunke, H. & Kronenberg, H. (1999). Identification of text on coloured book and journal covers, *Proceedings of Int. Conf Document Analysis and Recognition*, pp. 57-62

Sun, J.; Hotta, Y. & Katsuyama, Y. (2004). Low resolution character recognition by dual eigenspace and synthetic degraded patterns, *Proceedings of ACM Hardcopy Document Processing Workshop*, pp. 15-22

Thillou, C. & Gosselin, B. (2004). Segmentation-based binarisation for color degraded images, *Proceedings of Int. Conf. on Computer Vision and Graphics*

Thillou, C.; Ferreira, S. & Gosselin, B. (2005). An embedded application for degraded text recognition, *Eurasip Jour. on Applied Signal Processing, Special Issue on Advances in Intelligent Vision Systems: methods and applications*, Vol. 13, pp. 2127-2135

Wang, B.; Li, X.-F.; Liu, F. & Hu, F.-Q. (2004). Colour text image binarisation based on binary texture analysis, *Proceedings of Int. Conf. Acoustics, Speech and Signal Processing*, pp. 585-588

Wesolkowski, S. (1999). Colour Image Edge Detection and Segmentation: a Comparison of the Vector Angle and the Euclidean Distance Colour Similarity Measures, Master thesis, University of Waterloo

Wolf, C.; Jolion, J. & Chassaing, F. (2002). Text localisation, enhancement and binarisation in multimedia documents, *Proceedings of Int. Conf. on Pattern Recognition*, pp. 1040-1057

Zandifar, A.; Duraiswami, R. & Davis, L.S. (2005). A video-based framework for the analysis of presentations/posters, *Int. Journal on Document Analysis and Recognition*, Vol. 7, No. 2-3, pp. 178-187

Zhang, J.; Chen, X.; Hanneman, A.; Yang, J. & Waibel, A. (2002). A robust approach for recognition of text embedded in natural scenes, *Proceedings of Int. Conf. on Pattern Recognition*

Zhou, J.; Lopresti, D. & Lei, Z. (1997). OCR for world wide web images, *Proceedings of SPIE on Document Recognition V*, Vol. 3027, pp. 58-66