Available online at www.sciencedirect.com
ScienceDirect
Computer Vision and Image Understanding 107 (2007) 97–107
Computer Vision and Image Understanding
www.elsevier.com/locate/cviu

Color text extraction with selective metric-based clustering ☆

Céline Mancas-Thillou *, Bernard Gosselin

Faculté Polytechnique de Mons, Boulevard Dolez, 31 7000 Mons, Belgium

Received 6 April 2006; accepted 20 November 2006
Available online 19 January 2007
Communicated by Rastislav Lukac

Abstract

Natural scene images usually contain varying colors which make segmentation more difficult. Without any a priori knowledge of degradations and based on physical light reflectance, we propose a selective metric-based clustering to extract textual information in real-world images. The proposed method uses several metrics to merge similar color together for an efficient text-driven segmentation in the RGB color space. However, color information by itself is not sufficient to solve all natural scene issues; hence we complement it with intensity and spatial information obtained using Log–Gabor filters, thus enabling the processing of character segmentation into individual components to increase final recognition rates. Hence, our selective metric-based clustering is integrated into a dynamic method suitable for text extraction and character segmentation. Quantitative results on a public database are presented to assess the efficiency and the complementarity of metrics, together with the importance of a dynamic system for natural scene text extraction. Finally running time is detailed to show the usability of our method.

Keywords: Clustering; Cosine-based similarity; Diffuse and specular surfaces; Natural scene; Text understanding

1. What are natural scene text images?

Camera-based images or sequences are mostly taken in natural scenes (NS). It gives users the freedom to capture anything under any conditions. Nevertheless, to understand challenges of this paper, it is needed to make a distinction between *camera-based document analysis* and *natural scene text understanding*. The first category focuses more on perspective correction, unwarping, low resolution text recognition, and layout analysis while the second one deals with text detection, text extraction from background, character segmentation and recognition. Moreover, issues of natural scene text are close to the ones of scene text in videos, video graphics or text from WWW pages embedded in logos or complex backgrounds. In the subsequent sections, we shall discuss challenges and solutions for natural scene images. Fig. 1 on right shows visual issues we take into account.

Definition of natural scene text images: Still images or video frames of a scene with no a priori knowledge of environment, lighting, objects supporting text, acquisition parameters and finally text itself. They could easily be viewed as text in real-world conditions without any constraints. The quality of such images usually varies depending on the following:

- the environment: complex or clear backgrounds;
- the lighting: glossy or diffuse reflection, shadows and highlights;
- the objects: matte or shiny and curved or not curved surfaces;
- the acquisition parameters: low-resolution, blur, observation angle or sensor noise;
- the text: different font styles and sizes, artistic display.

☆ This work is part of the Sypole project and is funded by Ministère de la Région wallonne in Belgium.
* Corresponding author.
E-mail address: celine.thillou@fpms.ac.be (C. Mancas-Thillou).
URL: http://www.tcts.fpms.ac.be/~thillou (C. Mancas-Thillou).

1077-3142/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2006.11.010

Fig. 1. Difference between camera-based document (left) and natural scene text (right).

This listing shows the wide range of possible degradations and emphasizes how much versatility is required by natural scene text analysis systems to handle most of the situations.

Early research on natural scene text analysis included fixed camera systems with license plate reading for parking lot tracking, speed camera or road sign recognition, either for driver assisted systems or for information display on windshields. Recently, new challenges and applications appeared with the advent of digital cameras and camera phones and their increasing popularity. From this evolution, interesting applications came out such as text recognition and translation on a personal digital assistant (PDA) for foreigners, where the usefulness is specially for languages with unknown character sets. Another challenge, which is the aim of this work, is to introduce reading systems on an embedded device for visually impaired people. The accessibility of written information in daily life for blind consumers gives them more autonomy in a society driven by textual information.

Several steps are required for natural scene image analysis and we will focus on the text extraction part, which is the segmentation of text from background, followed by character segmentation and recognition steps. The preliminary text detection step will only be briefly mentioned, being out of focus of this paper. Hence, previous works in text extraction will first be described in Section 1. We shall then detail physical sources of color variations in an image in Section 3. It will enable to understand how to circumvent them in Section 4. In Section 5, we will present our fully automatic text extraction algorithm using selective metric-based clustering. Our system exploits the subsequent steps of character segmentation and recognition to dynamically improve text extraction in order to increase recognition rates. Section 6 will display results of the system by pointing out the dynamic property of our algorithm. Finally, we will conclude this paper in Section 7.

2. State-of-the-Art in natural scene text extraction

Most papers dealt with text detection in order to locate text in images but performed poorly in character recognition due to the missing step of text extraction. Hence, recently, this latter part appeared necessary for an efficient natural scene text analysis system. Gatos et al. [1] first extracted text with the luminance component only, then binarized the gray-level image and the inverted image with an adaptive thresholding algorithm, and finally, they chose the optimum between both binarizations by estimating the background surface. Color information was not exploited but the adaptive binarization handled uneven lighting. The issue of complex backgrounds was not detailed. Other non-color-based techniques for natural scene text extraction could be found in the very interesting survey of Jung et al. [2].

Existing color text extraction systems usually operate in one or more color spaces. Hase et al. [3] used the *CIE* $L^*a^*b^*$ color space and a histogram analysis to define the number of colors by frequency occurrences. Experiments were done on magazines and cover pages with complex backgrounds. As a scanner-based acquisition was taken into account, shadows and highlights were not supported. For WWW images, Karatzas and Antonacopoulos [4] segmented text with a split-and-merge strategy based on the *Hue-Lightness-Saturation* (*HLS*) color space. Characters of words were then merged by the alignment property of text lines in several orientations. Hence non-aligned text and complex backgrounds were supported but no details were given for uneven lighting. For caption text in video sequences, Du et al. [5] applied entropy-based thresholding on each of the *R*, *G*, and *B* channels. Based on a between-class/within-class variance criterion, the three subimages were partially merged to recompose the final binary image. With all degradations needed to be handled in natural scene images, this algorithm performed poorly on our data set.

Liu et al. [6] experimented a mixture model of Gaussians with parameters tuned by the expectation–maximization algorithm. Only two distinctive colors were assumed (text or non-text) and they fed their algorithm with *RGB* color data. Results were given only for the text detection part and the efficiency of text extraction was not assessed. Similarly, Gao et al. [7] used the same algorithm but determined the number of Gaussian mixtures by taking advantage of the text layout syntax. This approach made character recognition easier using properties of character components. Quantification of results was mainly done on Chinese text. Text properties such as height and width, spacing consistency, character-like aspect ratio and periodicity of vertical projection were also exploited in Crandall et al. [8] for extraction of caption texts, but without color information.

Color segmentation methods include clustering algorithms and recently, such techniques performed well on color text extraction. Wang et al. [9] used dimensionality reduction and graph theoretical clustering to segment text and to define the number of clusters dynamically. To circumvent a too high number of clusters, merging techniques were used with binary texture analysis (run-length histogram and spatial-size distribution) on each candidate image, followed by a linear discriminant analysis. Garcia

and Apostolidis [10] performed text extraction with a 4-means clustering on already detected text areas as in our text extraction algorithm. They obtained best non-quantified results in the *Hue-Saturation-Value* (*HSV*) color space. Thillou and Gosselin [11] segmented color text with a 3-means clustering algorithm in the *RGB* color space, where discrimination on clean and complex backgrounds was previously done to merge clusters more efficiently. Ashida et al. [12] chose the fuzzy C-means algorithm in the *CIE* $L^*u^*v^*$ color space. Clusters were then split, merged or discarded depending on the standard deviation within a cluster and a predefined threshold value.

This brief overview of literature highlights the fact that no color space fits particularly well with natural scene images. Hence, to focus on versatility, we shall aim at enhancing the complementarity between different metrics in the same color space along with intensity information.

3. Physical approach: light, camera and object

Illumination can vary drastically depending on the surrounding environment and these changes induce varying perceived colors. One of the human mechanisms for color constancy is chromatic adaptation, based on a chromatic behavior. In color segmentation, research attempts to reproduce the same effect for computers, which means merging similar colors independently on viewing conditions and environment. Hence, the triplet—consisting of light, camera, and object—must be considered to evaluate all possible degradations and color variations.

3.1. Light

As stated in [13], natural light has a diffuse behavior in which rays do not have a privileged orientation. Hence only a diffuse illumination source is considered in this paper, which means punctual source diffused equally in all directions. Although not included in our model, directional lights such as flashes or spots may be included in shiny models of objects.

3.2. Camera

The camera sensor is the particular light observer and viewing angle of the camera can induce different color perception of an object under diffuse lighting. Nevertheless, still camera-based images are considered here and the viewing angle is constant throughout an image. The impact of an observer has great importance on object classification or content-based retrieval from a database where viewing conditions may change from one image to another, which is not the case in text extraction.

3.3. Object

The dichromatic reflection model, introduced by Shafer [14], states that light is reflected on inhomogeneous dielectric materials in diffuse and specular reflection. Light I_r reflected from the surface of a colored object is a function of pixel location x and wavelength λ

$$I_r = \text{diffuse reflection} + \text{specular reflection} \quad (1)$$

$$I_r = \alpha(x)S(\lambda)E(\lambda) + \beta(x)E(\lambda) \quad (2)$$

where $E(\lambda)$ is the spectral power distribution of a light source, $S(\lambda)$ is the spectral-surface reflectance of an object, $\alpha(x)$ is the shading factor and $\beta(x)$ is a coefficient for the specular reflection term.

3.3.1. Diffuse reflection

Matte surfaces or Lambertian reflectors are considered in this case under the assumption of a white light source. The distribution of exiting light can be described by Lambert's cosine law, which states that the reflected light I_r appears equally bright regardless of viewing conditions. Light perceived I_p by the camera or the observer, which is equal to I_r, is the product of intensity of the light source I_s by the cosine of the angle θ_i between I_s and the normal direction $\overrightarrow{N}$ to the surface (Eq. (3)), perturbed by the shading factor, $\alpha(x)$. Hence, as the θ_i increases, the amount of light decreases.

$$I_p = I_s \times \cos(\theta_i) \times \alpha(x) \quad (3)$$

Gevers [15] concluded that a uniform colored surface which is curved returns different intensity values to the camera. Fig. 2 displays the Lambert's cosine law for different locations of a curved surface. This case is quite frequent in NS images.

3.3.2. Specular reflection

This case refers to shiny objects, presenting a globally symmetric reflection to the normal direction $\overrightarrow{N}$, hence with reflected intensity I_r depending on the viewing conditions. Specular reflection is an ideal case of glossy reflection. Phong's model [16] describes the geometry of image formation for computer generated images and eases our understanding of color variations in an image. The camera's viewing angle is fixed in NS text extraction but color varies with the surface orientation, leading to highlights. Fig. 3 shows the orientation of the exiting surface reflection I_r. The perceived intensity I_p is a function of the angle θ_j between I_p and I_r, described by Eq. (4).

$$I_p = I_s \times \cos^n(\theta_j) \times \beta(x) \quad (4)$$

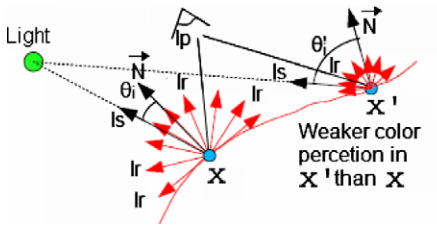


Fig. 2. Difference between diffuse reflection in different locations on a curved surface.

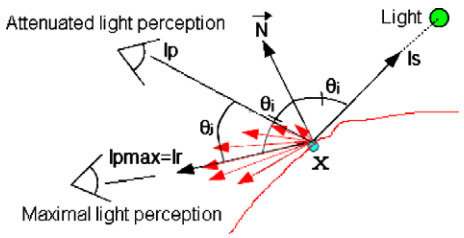


Fig. 3. Specular reflection defined by Phong's model [16].

where $\beta(x)$ defines the glossy coefficient for the point x and n is the diffusion coefficient around I_r, attenuating the perceived light when I_p is different from I_r.

Interreflection between objects is a generalisation of previous cases using a single light source because reflections onto objects are considered as another light source. Shadows are present due to obstacles (other objects) between light and object to be viewed. Light in the shadowed part results from other attenuated parts of incident light around.

All viewing conditions, matte or shiny surfaces, and diffuse illumination source induce that:

- Two identical (or different) colors in an object may be perceived identically (or differently) by the camera, which is the usual case with almost no degradations. For instance, this case occurs under diffuse light, matte and plane surfaces. The color magnitude is sufficient to merge (split) these two colors.
- Two different colors in an object may be perceived identically by the camera. This phenomenon is called illuminant metamerism where two colors match when viewed under one light source, but do not match when viewed under another, and vice versa. Color magnitude is not sufficient to separate them, thus color orientation should be used additionally.
- Two identical colors in an object may be perceived (slightly) differently due to a curved matte or shiny surface for example. This case is the main issue of object-driven segmentation where similar colors must be merged even with (slightly) different perceived colors by the camera. Color magnitudes are quite different and are useless to merge them. Hence one can benefit from color orientation by using a small angle between color vectors to group them together. For example, bright red and dark red have different magnitudes but similar color orientation in the *RGB* color space.

The color formation in a camera sensor does not handle all unknown sources of variations, present in natural scene images but emphasizes the importance of using both magnitude and orientation of color to support varying colors in a scene. For more information, the reader may refer to [15,17].

4. Computer-based approach: combination of several metrics

For traditional color segmentation algorithms, several color spaces are used for different applications as mentioned in Section 1. Nevertheless, Mancas-Thillou and Gosselin [18] previously explained why *RGB* color space handles variability of natural scenes better than most other spaces; it is general enough to support all degradations. To circumvent lighting effects, such as highlights or slightly varying colors in this work, we propose the use of two metrics to merge similar colors together in the *RGB* color space. For images with no degradations, the first metric which is the Euclidean distance D_{eucl} has proven its efficiency over all kinds of clustering distances while for more complex images, the second metric, which is a cosine-based similarity $S_{\cos}$, enables to cope with degradations. The Euclidean distance is defined by

$$D_{\text{eucl}}(x_i, x_j) = \sqrt{\sum_{k=1}^{dim} (x_{i,k} - x_{j,k})^2} \tag{5}$$

where *dim* is the dimension of the color space, i.e. $dim = 3$ for *RGB*.

The most known cosine-based similarity is defined by

$$S_{\cos_{\text{original}}}(x_i, x_j) = \frac{x_i . x_j}{\|x_i\| . \|x_j\|} \tag{6}$$

However, several cosine-based similarities have been designed and can be found in the tutorial work of Lukac and Plataniotis [19]. After different tests on natural scene images and especially on the database described in Section 6, we chose the following similarity $S_{\cos}$:

$$S_{\cos}(x_i, x_j) = 1 - \left(\frac{x_i . x_j}{\|x_i\| . \|x_j\|}\right)\left(1 - \frac{\|x_i\| - \|x_j\|}{\max(\|x_i\|, \|x_j\|)}\right) \tag{7}$$

This similarity presents a more compact support and performs better in natural scene text images.

Cosine-based similarity has been previously used for edge detection and color segmentation by Wesolkowski and Jernigan [20,21], color classification by Hild [22] and vector directional filtering by Lukac et al. [23], for example. In the case of text extraction, this metric enables to merge colors with large D_{eucl}, such as colors varying in intensity but not in hue. Cosine-based similarity presents several advantages such as:

Hue representation. Inside the *RGB* color space, a reliable and simple method to obtain hue information is through a cosine-based similarity [20].

Varying color characterization. Similar colors have parallel orientations even when degraded with uneven lighting or shiny material. In natural scene images (slight), variations are a frequent occurrence within the same object of same color due to all sources of variations described in Section 3. Color can gradually change and by definition, a cosine-based similarity can circumvent this issue.

Complementarity with the Euclidean distance. A cosine-based similarity, defined in the directional domain, calculates changes in color chromaticity whereas magnitude processing-like distance, such as the Euclidean one, calculates changes in the luminance information. Combining both magnitude and directional processing allows to

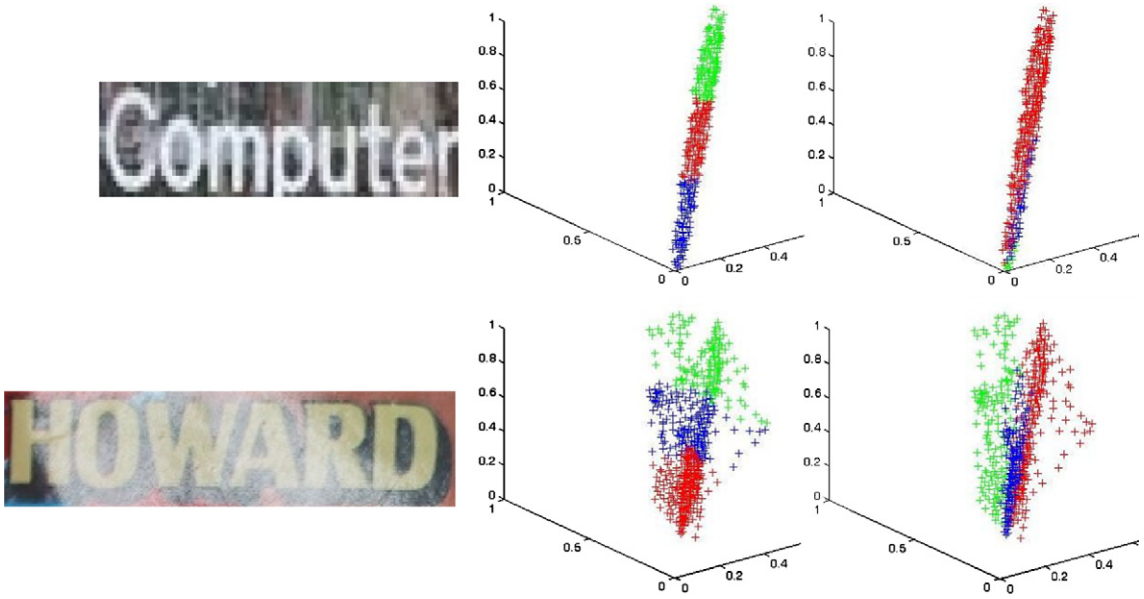


Fig. 4. (*R–G–B*) view of the clustering results done by D_{eucl} (middle) and by $S_{\cos}$ (right) on initial images (left). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

process a color image using both luminance and chrominance information, and potentially increases the performance. Additional information on this issue can be found in [24]. Moreover, in a clustering process as displayed in Fig. 4, we show the cluster definition using D_{eucl} and $S_{\cos}$ for some natural scene images. From the *RGB* color space, D_{eucl} separates pixels in the (*R–G–B*) view mostly in a horizontal way with groups presenting quite same volumes while $S_{\cos}$ does the same operation in a more vertical way with groups presenting different sizes. These observations are quite logical due to the definition of each distance but show an important complementarity depending on colors in the image.

5. The proposed algorithm: parallel use of clustering metrics and Log–Gabor filters

As shown in Fig. 5, various preprocessing steps [25] such as denoising and resolution enhancement are often used prior to the essential analysis step. Text detection and localization are usually the initial steps of scene text analysis. They determine if and where the text is present in the image. Issues include detection of text with varying sizes and fonts by merging characters of a word with similar properties. Several techniques, based on edges, colors or textures, are available [26]. Consecutive steps, possible image inversion, color text extraction and character segmentation, are detailed in the following subsections. Independent steps take premature decisions and lead to poor results due to all the variations of natural scene images. Hence, all these steps are dynamically related to increase robustness of each subpart, which is of utmost importance for natural scene images and to ease character recognition, which is the final step of scene text analysis.

5.1. Possible image inversion

In some images, text may appear bright on a dark background or inversely. General purpose of text extraction is segmentation of textual foreground. As described in Section 4, we use color magnitude and orientation in parallel, where the latter means exploitation of the angle between two color vectors relative to the origin of the *RGB* color space. This origin point corresponds to no illumination with $R = 0$, $G = 0$ and $B = 0$. Hence dark text on bright background is more appropriate as angles with colors close to the origin have wider dynamics. Cosine-based similarities add hue information inside RGB and similar to the

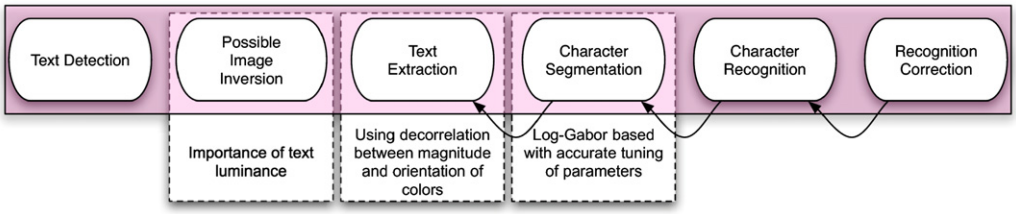


Fig. 5. Chart of natural scene text understanding systems.

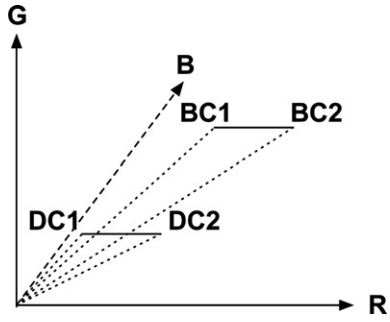


Fig. 6. Larger angle between dark colors DC1 and DC2 than between bright colors BC1 and BC2, even if $d(\mathrm{BC2},\mathrm{BC1})$ equals to $d(\mathrm{DC2},\mathrm{DC1})$ with d, the Euclidean distance.

drawbacks of hue, it becomes unstable for small angles. Fig. 6 shows the difference of angles for two colors of dark and bright text. The two sets of colors have the same Euclidean distance, either in dark or bright area. Hence our algorithm works better with dark text on bright background when degradations are present and color orientation needs to be used. Fig. 7 displays the difference of extraction results with and without inversion for bright text on dark backgrounds. The extraction process is the same one that is described in Section 5.2.

The choice of inversion is based on a quick global image thresholding such as the well-known Otsu method [27]. The maximum between black or white pixels on the image borders implies that text is brighter or darker than the background, assuming text is not mainly connected to borders. In most text detection algorithms, text is the main part of the textual areas but is not cut or linked with image borders.

5.2. Text extraction-by-segmentation

Our text extraction is performed in the RGB space using a clustering algorithm with two different metrics defined in Eqs. (5) and (7). Following this first step, a particular metric has to be chosen and we use spatial information to take the right decision regarding the best segmentation between both metrics. A smart way to combine color or gray-level variation with spatial information is by using Log–Gabor filters. Fig. 8 displays our text extraction-by-segmentation algorithm to get our final text cluster.

5.2.1. Selective metric-based clustering

In order to segment similar colors together, we use an unsupervised segmentation algorithm with a fixed number of clusters. As areas are constrained, we use a 3-means clustering where two clusters belong to textual foreground and background, while the third one is a noisy cluster dedicated

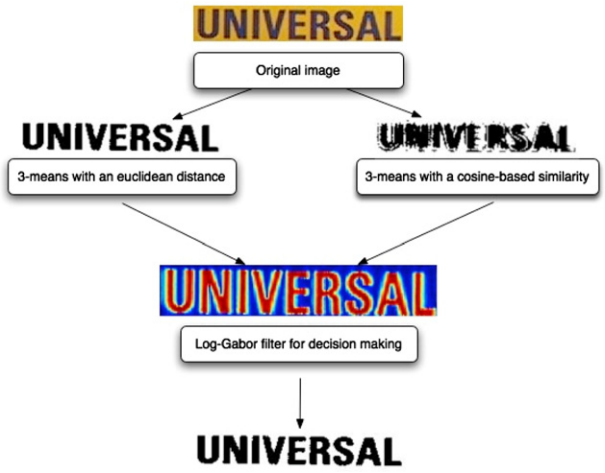


Fig. 8. Overview of the proposed algorithm combining color and spatial information. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

to either noise in complex images or edges of characters that are always slightly different, even in clear images. The background color is selected very easily and efficiently as being the color with the highest rate of occurrences on the image borders.

Next, we propose a new text validation measure M to find the most textual foreground cluster over the two remaining clusters. Based on properties of connected components of each cluster, spatial information is already added at this point to find the main textual cluster. The proposed validation measure, M, is based on the largest regularity of connected components of text compared to those of noise and background and is defined as follows:

$$M = \sum_{i}^{N} \left| \mathrm{area}_i - \frac{1}{N} \left(\sum_{i}^{N} \mathrm{area}_i \right) \right| \tag{8}$$

where N is the number of connected components and area_i refers to the area of component i. This measure enables to compute the variation in candidate areas. The main textual cluster is identified as the one having the smallest M. If the third unknown cluster belongs to text, both textual clusters need to be merged. A new computation of M is done considering the merging of both clusters. If M decreases, then the fusion is performed.

The 3-means clustering algorithm is performed twice with both metrics, D_{eucl} and $S_{\cos}$ described in Section 4. Fig. 9 displays examples where D_{eucl} performs better than $S_{\cos}$ (top), the inverse result (middle) and a last example (bottom) where both clustering distances perform quite similarly.

Fig. 7. Initial image (left), result of our algorithm without inversion (middle) and with inversion (right).

Fig. 9. Initial color images (left), extraction done by D_{eucl} (middle), extraction done by $S_{\cos}$ (right). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

Fig. 10. Results of Log–Gabor filters on the three examples of Fig. 9. Left: by using the mask of the extraction done by D_{eucl}, right: by using the mask of the extraction done by $S_{\cos}$.

The best result is chosen with the same tool that we use for character segmentation described more accurately in Section 5.3. Color information is a very consistent clue for natural scene images. However, the segmentation process we use does not exploit spatial information. It becomes necessary to handle it for text, which is a very particular object having interesting spatial properties. In our proposed algorithm, it enables the choice of the best segmentation among the two metrics inside the clustering algorithm.

In order to segment characters properly, we need to have spatial information to locate characters in the image as well as frequency information to use illumination variation to detect character edges. In this paper, we opt for Log–Gabor filters proposed by Field [28], because they have an extended tail in high frequencies as required for natural scene images.

Fig. 10 shows the result of the same three examples in Fig. 9 multiplied by the mask of each segmentation performed previously. Log–Gabor filters present globally high responses to characters. Hence, in order to choose efficiently which clustering distance is better to handle text extraction, we perform an average of pixel values inside each mask. The mask which has the highest average is chosen as the final segmentation.

5.3. Character segmentation-by-recognition

In order to segment characters into individual components properly, we need to perform simultaneous processing of spatial information to locate the character separation in the image and frequency information to use intensity variation for detecting these separations and to complement color information used in text extraction. At this stage, more accuracy is required and slight color variations between characters are better recovered with intensity differences. Gabor-based filters could be a choice to address this problem. However, Gabor filters present limitations: large bandwidth filters induce a significant continuous component, hence only a maximum bandwidth of 1 octave could be designed. Field [28] proposed an alternative function called Log–Gabor which lets us choose a larger bandwidth without producing a continuous component.

Log–Gabor filters in frequency domain can be defined in polar coordinates by $H(f,\theta) = H_f \times H_\theta$ with H_f being the radial component and H_θ, the angular one

$$H(f,\theta) = \exp\left\{\frac{-[\ln(f/f_0)]^2}{2[\ln(\sigma_f/f_0)]^2}\right\} \times \exp\left\{\frac{-(\theta-\theta_0)^2}{2\sigma_\theta^2}\right\} \qquad (9)$$

where f_0 is the central frequency, θ_0 is the filter direction, σ_f defines the radial bandwidth B in octaves with $B = 2\sqrt{2/\ln 2} * |\ln(\sigma_f/f_0)|$ and σ_θ defines the angular bandwidth $\Delta\Omega = 2\sigma_\theta\sqrt{2\ln 2}$.

As we are looking for vertical separation between characters, we only use two directions for the filter, the horizontal and the vertical one. Hence, for each directional filter, we use a fixed angular bandwidth of $\Delta\Omega = \pi/2$. Log–Gabor filters are not really strict with directions and defining only two directions enables handling of italic and/or misaligned characters. For highly misaligned characters, the number of directions can be simply increased to overcome this additional degradation.

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 their wavelength, hence it is quite 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 such as in Fig. 11, 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 'sharper' 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 Fig. 12.

As degradations and conditions of frequency estimation are quite unexpected, we choose the bandwidth filter in a dynamic way using recognition results. As objects to be

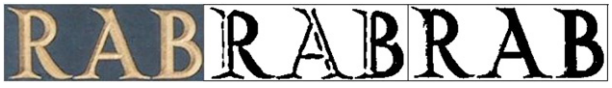


Fig. 11. From left to right: original image, segmentation with misestimated thickness, segmentation with the same thickness corrected by a larger bandwidth.

Fig. 12. From left to right: original image, binary version, segmentation by large bandwidth with still connected characters and segmentation by narrow bandwidth with individual characters.

segmented have a particular meaning, we can use segmentation-by-recognition to choose the convenient bandwidth. Based on several natural scene images, we set the initial and final values for the bandwidth estimation. From around 2 octaves to around 8 octaves, which makes σ_f/f_0 vary with a step of 0.1, we process six filters and provide the result to an OCR engine. To choose the bandwidth for filters, we use a home-made OCR algorithm composed of a multi-layer perceptron with geometrical features to recognize characters [29]. Fig. 13 shows two examples with varying bandwidths and results from recognition, which permit us to take the right decision for bandwidth estimation. Recognition rates for each character (or assumed character) are averaged and the maximum score enables to estimate the bandwidth. The first example is an image with little contrast between characters and background, where the best result is obtained with a recognition rate of 0.90. The second image presents a misaligned and slanted text where better results are obtained with a larger bandwidth at a recognition rate of 0.86. This estimation needs six straightforward filters with only one frequency which enables the use of Log–Gabor filters for character segmentation in an embedded context.

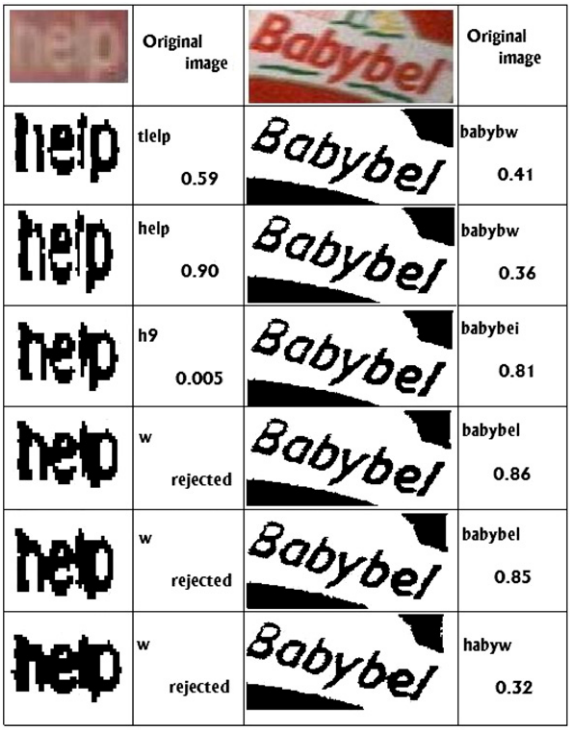


Fig. 13. First and third columns: character segmentation with bandwidth varying from 2 (on top) to 8 octaves (on bottom), 2nd and 4th columns: OCR results with average recognition rate.

Fig. 14. Some examples of natural scene images either taken by a visually impaired people or from WWW pages.

6. Results and influence of feedback mechanisms

Like in some color segmentation algorithms, testing natural scenes text understanding system with synthetic images is not relevant, by definition. Hence, we use a publicly available database from the Robust Reading Competition of ICDAR 2003 [30]. Images contain different kinds of degradations with uneven lighting, curved or/and shiny surfaces, complex backgrounds, different text fonts and sizes, different resolutions and so on. The database *Sample of Words* of this competition is very representative of natural scene images and includes 171 words, with some words even not readable by humans. Some examples from this database are shown in Fig. 1 on right and Figs. 4,6,8,11,12, and 15.

In order to be independent of the databases, we tested our algorithm on other sets of images such as pictures taken by visually impaired people in the framework of the Sypole project[1] or from WWW pages, as displayed in Fig. 14. Text detection was done by A. Chen's algorithm[2] [26]. Tests have been done on 500 images acquired by blind people and 150 images from the Internet. Results were quite similar to the *Sample of Words* database.

Several tests have been done to evaluate the efficiency of the complete algorithm. First of all, the use of several clustering distances is compared to a single distance-based clustering using only D_{eucl}, which works in most cases. The improvement in the number of better-extracted words is of 6.3%, showing the efficiency of the simultaneous use of both clustering distances. Performance of word extraction is measured by a home-made OCR algorithm [29], dedicated to natural scene images. A word is considered to be better extracted if the Levenshtein distance d_L, between the ground truth t and the recognized word r, is smaller. The Levenshtein distance [31] 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

[1] http://tcts.fpms.ac.be/projects/sypole/index.php.

[2] This text location system is publicly available. Issued from the Robust Reading Competition of ICDAR 2003, it is proved to be the best web-deployed one.

weights for each operation are employed in our computation.

To highlight effectiveness of our proposition, we compare our selective-metric based clustering (SMC) with two competing and recent algorithms in natural scene text extraction. Comparisons are firstly done with the algorithm of Gatos et al. [1] which uses background surface thresholding with an adaptive binarization, followed by an image upsampling to improve quality. Note that text extraction is performed on the whole image, without any preceding text detection. Non-text components are then removed with several rules based on text properties. The second comparison is performed with the algorithm of Garcia and Apostolidis [10], which uses Euclidean-based color clustering in the *HSV* color space. Manual combination of clusters is performed to compare only quality of text extraction with this latter algorithm.

In order to perform the same evaluation as in Gatos et al. [1], we compute the Levenshtein distance on the same images (from the public ICDAR 2003 database, as well), as displayed in Table 1, with the same commercial OCR. Improvement from Gatos et al. [1] may be observed with a decrease in error rate of around 43%.

For comparison with the algorithm of Garcia and Apostolidis [10], Precision and Recall are defined enabling evaluation of text extraction quality

$$\text{Precision} = \frac{\text{Correctly extracted characters}}{\text{Total of extracted characters}} \tag{10}$$

$$\text{Recall} = \frac{\text{Correctly extracted characters}}{\text{Total number of characters}} \tag{11}$$

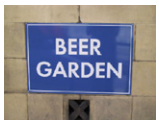
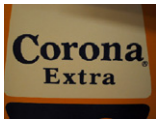
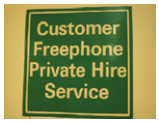
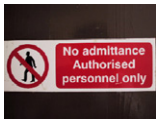
Precision measures the quality of extraction while Recall measures the quantity of high quality extraction. "Correctly extracted characters" means characters which are extracted without noise or missing parts of the character. As no ground truth is available, visual inspection is performed and results are given in Table 2. Tests have been based on the public *Sample of Words* database. Results assess a global improvement with Precision increasing from 0.64 to 0.93 and Recall from 0.56 to 0.91.

The combination of color, intensity and spatial information has to be assessed to measure the impact of decision taken by the whole algorithm. In order to decide which segmentation is correct, visual judgement is employed for segmentations presenting different results, as one image over the two results is hardly readable. For cases where both segmentations give similar results, we use the same homemade OCR, after separation into individual characters. Some examples of the results produced using our text extraction algorithm are shown in Fig. 15.

In order to assess the use of spatial information to choose between the two distances, the silhouette Sil [32]—which can be seen as a measure of how well clusters are separated—is calculated as follows:

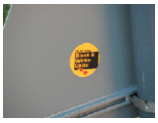
$$\text{Sil} = \frac{\min(\text{mean}_{\text{between}}(i,k)) - \text{mean}_{\text{within}}(i)}{\max(\text{mean}_{\text{within}}(i), \min(\text{mean}_{\text{between}}(i,k)))} \tag{12}$$

Table 1
Comparison of OCR results between our proposed method and Gatos et al's one [1]

NS images	OCR alone	Gatos et al. [1]	Our SMC method
	21	0	0
	25	18	6
	5	4	0
	2	2	1
	3	3	4
	1	1	0
	0	0	0
	2	1	1
	0	0	0
	2	1	1
	2	2	0
	32	3	4

(*continued on next page*)

Table 1 (*continued*)

NS images	OCR alone	Gatos et al. [1]	Our SMC method
	2	0	0
	39	18	19
	10	1	1
	10	10	0
	6	3	0
	38	16	10
Total	201	83	47

Evaluation is based on Levenshtein distance from the ground truth.

Table 2
Precision and Recall measures for text extraction evaluation between Garcia and Apostolidis's algorithm [10] (G and A) and our proposed SMC method

Results	G and A's [10]	SMC
Precision	0.64	0.93
Recall	0.56	0.91

Fig. 15. Examples of text segmentation by our SMC method. First and second set: initial image (top) and extracted text (bottom).

Fig. 16. Error example of our selective metric-based clustering: initial color image (left) and result (right). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this paper.)

where $\text{mean}_{\text{within}}(i)$ is the average distance from the ith point to other points in the same cluster and $\text{mean}_{\text{between}}(i,k)$ is the average distance from the ith point to points in another cluster k. The average distance is defined either by D_{eucl} or $S_{\cos}$.

It is appropriate to think that best text extraction results present the best separation between clusters. However, the silhouette method performs well in 77.7% images while our proposed method using spatial information in 93.2%. Hence, an improvement of 19.9% may be observed.

Due to the explosion of camera phones and digital cameras resulting in huge amount of images to process for text extraction, the algorithm needs to be relatively fast in order to provide satisfying results for a frequent use. The text extraction algorithm runs in 0.61 s on average for each image of the *Sample of Words* database on a PC with a micro-processor Pentium M-1.7 GHz. The average resolution of these samples is 1600×1200 pixels. Codes of text extraction were developed in C but could still be optimized.

Our selective metric-based clustering uses mainly color information for text extraction and our system fails for natural scene images having embossed characters. In this case, foreground and background have the same color presenting partial shadows around characters due to the relief but not enough to separate textual foreground from background in a discriminative way as displayed in Fig. 16. Gray-level information with the simultaneous use of a priori information on characters could be a solution to handle these cases. Moreover, the recognition step may compensate this erroneous segmentation.

7. Conclusion

Natural scene text extraction and understanding represent new challenges based on the explosion of digital still cameras or camera phones in the market. Several types of degradations, such as uneven illumination or shiny materials, induce color variations. Traditional algorithms fail to handle this variability and the use of several color spaces or a dedicated one is a solution for a particular application only. In a general context which requires versatility, we propose a selective metric-based clustering using the Euclidean distance and a cosine-based similarity. Both metrics are complementary and their combination improves text extraction results. Spatial information is exploited to choose the right metric depending on images and based on Log–Gabor filtering. Particularly well defined for natural scene images, Log–Gabor filters

emphasize separation between characters and simultaneously choose which metric is the best and segment characters into individual components. Finally, some parameters of the filters are automatically tuned based on recognition results.

By using a public database, we compared our results with a single metric clustering and two competing algorithms and showed the impact of feedback mechanisms in natural scene images. A limitation of our selective metric-based clustering is for embossed text with similar colors for foreground and background. In our opinion, it may be solved by gray-level information and a priori information on text.

Acknowledgments

The authors wish to sincerely thank researchers in color vision and physical-based segmentation who kindly gave time and sent papers to make their field less opaque and more particularly Prof. M. Hild from Osaka University, Japan. Moreover, the authors thank the reviewers who contributed to the improvement of this paper.

References

[1] B. Gatos, I. Pratikakis, K. Kepene, S.J. Perantonis, Text detection in indoor/outdoor scene images, in: Proc. First Workshop of Camera-based Document Analysis and Recognition, 2005, pp. 127–132.
[2] K. Jung, K.I. Kim, A.K. Jain, Text information extraction in images and video: a survey, Pattern Recognit. 37 (2004) 977–997.
[3] H. Hase, T. Shinokawa, M. Yoneda, C.Y. Suen, Character string extraction from color documents, Pattern Recognit. 34 (2001) 1349–1365.
[4] D. Karatzas, A. Antonacopoulos, Text extraction from web images based on a split-and-merge segmentation method using colour perception, in: Proc. Int. Conf. Pattern Recognition, 2004, pp. 634–637.
[5] Y. Du, C. Chang, P. Thouin, Unsupervised approach to color video thresholding, in: Proc. SPIE Optical Imaging, vol. 43, n.2, 2004, pp. 282–289.
[6] Y. Liu, S. Goto, T. Ikenaga, A robust algorithm for text detection in color images, in: Proc. Int. Conf. Document Analysis and Recognition, 2005, pp. 399–403.
[7] J. Gao, J. Yang, Y. Zhang, A. Waibel, Text detection and translation from natural scenes, Carnegie Mellon University Tech. Report CMU-CS-01-139, 2001.
[8] D. Crandall, S. Antani, R. Kasturi, Extraction of special effects caption text events from digital video, Int. J. Doc. Anal. Recognit. 5 (2003) 138–157.
[9] B. Wang, X.-F. Li, F. Liu, F.-Q. Hu, Color text image binarization based on binary texture analysis, in: Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing, 2004, pp. 585–588.
[10] C. Garcia, X. Apostolidis, Text detection and segmentation in complex color images, in: Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing, 2000, pp. 2326–2330.
[11] C. Thillou, B. Gosselin, Color binarization for complex camera-based images, in: Proc. Electronic Imaging Conf. Int. Soc. Opt. Imaging, 2005, pp. 301–308.
[12] S. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, R. Young, K. Ashida, H. Nagai, M. Okamoto, H. Yamamoto, H. Miyao, Y. Zu, W. Ou, C. Wolf, J.-M. Jolion, L. Todoran, M. Worring, X. Lin, ICDAR 2003 robust reading competitions: entries, results and future directions, Int. J. Doc. Anal. Recognit. 7 (2005) 105–122.
[13] G. Sharma, Digital Color Imaging Handbook, CRC Press LLC, Boca Raton, 2003.
[14] S.A. Shafer, Using Color to Separate Reflection Components, TR-136, Computer Sciences Dept., University of Rochester, New York, 1984.
[15] T. Gevers, Color in Image Databases, Isis Report, University of Amsterdam, The Netherlands, 2000.
[16] B.T. Phong, Illumination for computer generated pictures, Communications of the ACM, vol. 18, n.6, 1975, pp. 311–317.
[17] W. Skarbek, A. Koschan, Colour image segmentation - a survey-, Technischer Bericht 94-32, TU Berlin, 1994.
[18] C. Mancas-Thillou, B. Gosselin, Color text extraction from camera-based images-the impact of the choice of the clustering distance-, in: Proc. Int. Conf. Document Analysis Recognition, 2005, pp. 312–316.
[19] R. Lukac, K.N. Plataniotis, A taxonomy of color image filtering and enhancement solutions, Adv. Imaging Electron Phys. 140 (2006) 187–264.
[20] S. Wesolkowski, E. Jernigan, Color edge detection in rgb using jointly Euclidean distance and vector angle, Vision Interface, Canada, 1999, pp. 9–16.
[21] S. Wesolkowski, Shading and highlight invariant color segmentation, EAC Tech. Research Report, University of Waterloo, Canada, 2000.
[22] M. Hild, Color similarity measures for efficient color classification, J. Imaging Sci. Tech. (2004) 529–547.
[23] R. Lukac, B. Smolka, K. Martin, K.N. Plataniotis, A.N. Venetsanopoulos, Vector filtering for color imaging, IEEE Signal Process. Mag. 22 (2005) 74–86.
[24] R. Lukac, K.N. Plataniotis, B. Smolka, A.N. Venetsanopoulos, Generalized selection weighted vector filters, Eurasip J. Appl. Signal Process. 12 (2004) 1870–1885.
[25] C. Mancas-Thillou, M. Mirmehdi, Super-resolution text using the Teager filter, in: Proc. Camera-based Document Analysis and Recognition, 2005, pp. 10–16.
[26] S. Lucas, C. Jaimez Gonzales, Web-based deployment of text locating algorithms, in: Proc. Camera-based Document Analysis and Recognition, 2005, pp. 101–107.
[27] N. Otsu, A thresholding selection method from gray-level histogram, IEEE Trans. Syst. Man Cybern. 8 (1978) 62–66.
[28] D.J. Field, Relations between the statistics of natural images and the response properties of cortical cells, J. Opt. Soc. Am. (1987) 2379–2394.
[29] C. Thillou, S. Ferreira, B. Gosselin, An embedded application for degraded text recognition, Eurasip J. Appl. Signal Process. Spec. Issue Adv. Intell. Vis. Syst. Methods Appl. 13 (2005) 2127–2135.
[30] Robust reading competition of ICDAR, <http://algoval.essex.ac.uk/icdar>, 2003.
[31] V.I. Levenshtein, Binary codes capable of correcting deletions, insertions and reversals, Soviet Physics Doklady, vol. 10, n.8, 1966, pp. 707–710.
[32] L. Kaufman, P.J. Rousseeuw, Finding groups in data: an introduction to cluster analysis, Wiley Editions, New York, 1990.