2009 10th International Conference on Document Analysis and Recognition

Text Segmentation in Colour Posters from the Spanish Civil War Era

Antonio Clavelli and Dimosthenis Karatzas
Computer Vision Centre, UAB, Barcelona, Spain
{aclavelli, dimos}@cvc.uab.es

Abstract

The extraction of textual content from colour documents of a graphical nature is a complicated task. The text can be rendered in any colour, size and orientation while the existence of complex background graphics with repetitive patterns can make its localization and segmentation extremely difficult. Here, we propose a new method for extracting textual content from such colour images that makes no assumption as to the size of the characters, their orientation or colour, while it is tolerant to characters that do not follow a straight baseline. We evaluate this method on a collection of documents with historical connotations: the Posters from the Spanish Civil War.

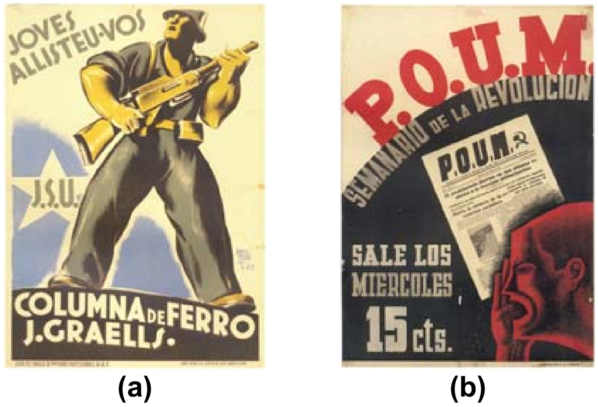
1. Introduction

Among the many collections of historical documents that are routinely the focus of mass-digitisation efforts, there are some of significant value that are typically overlooked due to their increased difficulty as far as it concerns the automatic extraction of textual content. The presence of colour and the overlapping of text and graphical regions are usual reasons for neglecting such documents.

One such collection is the Catalan Posters from the Spanish Civil War kept by the National Archive of Catalonia in Spain. The collection comprises about 10000 posters (1200 already scanned) with political, ideological and propagandistic content produced in the first half of the 20th century (see Figure 1).

The automatic extraction and interpretation of the semantic content in these images has a considerable scholarly value. Moreover, this collection is a good example of numerous other poster collections produced over the last century that share the same characteristics. Techniques devised for these images should therefore be generally applicable.

Technically speaking the extraction of the textual content from these images is a complicated process for a number of reasons. The text can be rendered in any style, colour, size and orientation, different fonts are often mixed on the same (curved) text line, characters are individually produced by the artist (instances of the same character slightly differ), while complex graphical backgrounds give rise to an excessive amount of noise components during segmentation.



(a) (b)

Figure 1. Spanish Civil War era posters.

In this paper we propose a method for extracting textual content from colour documents of a graphical nature that makes no assumption as to the size, orientation or colour of the characters, while it is tolerant to characters that do not follow a straight line.

In Section 2 an overview of existing work on the topic is presented. Section 3 details our method for text extraction, while Section 4 presents results over a set of images from the Spanish Civil War poster collection. Section 5 concludes this paper.

2. Literature Review

Contrary to classical document image analysis, where the problem of segmentation is equivalent to identifying regions of text (layout analysis) on a black and white image, in the case of colour images the problem is pixel-level segmentation, namely to produce such a black and white image on which character recognition can then take place.

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.32

The segmentation of text in colour images has received increasing attention over the last decade. Ad-hoc solutions for various types of colour text containers have been reported including colour paper documents [1-3], text in Web images [4], real scenes [5-7], video sequences [8] etc. Generally the reported methods have limited applicability outside their restricted focus due to the special characteristics of each image type.

In bottom-up segmentation schemes, results are generally represented as connected components, which have to be subsequently classified as belonging to the textual content or not. Neither colour segmentation nor component classification is a trivial problem.

Retornaza and Marcotegui [7] construct connected components using contrast information in a mathematical morphology framework. Component classification is performed on the basis of 27 features while a subsequent merging process is used to group them into horizontal text lines.

Jung et al. [6] use a stroke filter to perform local region analysis. Stroke-like structures are identified in all possible orientations sizes and positions. The hypothesis here is that characters have uniform thickness.

A typical problem with the classification of the resulting connected components is the sheer number and variability of components corresponding to parts of the image background. It is frequently the case that the ratio between foreground and background components in an image is between $1{:}10^2$ and $1{:}10^3$ depending on the complexity of the graphical content. Therefore, attempts to classify individual components yield limited results unless large numbers of features are used [7]. Generally, identifying groups of similar components is instead used to extract whole text lines. Even at this level problems might arise, as complex graphical regions generally give rise to patterns that resemble text.

A number of explicit or implicit assumptions are usually made when working with complex colour images. In most of the cases, text is assumed to have a constant colour [1-3, 5-8]. Also more often than not, text is supposed to be arranged on straight text lines, and quite frequently only horizontal straight lines are looked for [1, 5-8].

3. Text Extraction Method

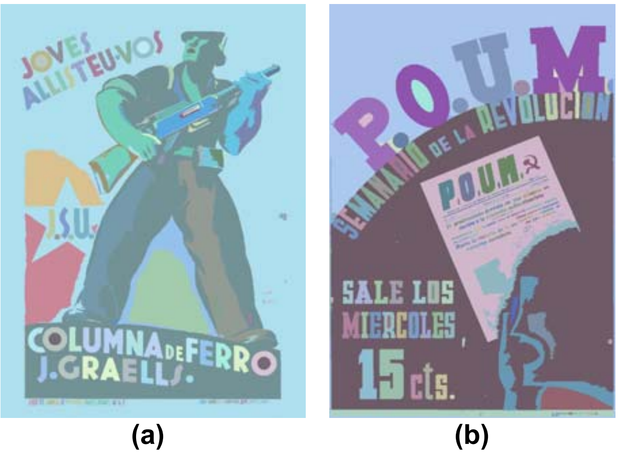
The text extraction method follows three steps: colour segmentation, individual connected component filtering and connected component grouping into text-lines. These are explained in detail in the next sections.

3.1. Colour Segmentation

In the specific document collection we are examining here characters are rendered in uniform colour. This is an advantage as it ensures a mostly error-free segmentation. Due to this a relatively simple colour segmentation method has been employed here.

The segmentation process used here is a one-pass algorithm which creates 8-connected components based on colour similarity. The algorithm processes the image in a left-to-right, top-to-bottom fashion, and for each pixel calculates its colour similarity with the four of its neighbours that are already assigned to a connected component. The pixel then is either assigned to one of the existing components if their colour difference is below a set threshold, or it is used as the seed for a new component. Additional checks are performed in case a pixel is similar to more than one existing component, in which case components might be merged. The algorithm is further described in [9].

Colour similarity is assessed in the RGB colour space. A better choice would be a perceptually uniform colour space such as CIELab or CIELuv, but since text here is rendered in a uniform colour working with RGB results in a considerable faster algorithm without seriously affecting performance. The colour threshold was set to 75 after experimentation (see section 4). The result of the segmentation process is a set of connected components that comprise pixels of similar colour. An example is shown in Figure 2.



(a) (b)

Figure 2. Segmentation result on the images of Figure 1.

3.2. Component Filtering

In order to reduce the number of components that correspond to areas of the background, an initial size filtering is performed based on the diagonal of the bounding box of the components. Components that have a diagonal of less than five pixels are deemed too small to represent characters and are discarded. Similarly, components with a diagonal bigger than half the minimum dimension of the image are deemed too big and are discarded.

Contrary to existing methods, we do not intend to classify single components as text or non-text, but instead extract complete groups of components as text lines. This filtering step is therefore very conservative and only intended to discard components that are extremely small or too big to be characters. Nevertheless it reduces a lot the complexity of the following steps.

3.3. Text Line Identification

The remaining components correspond to characters as well as to parts of the background. The question posed at this point is how to select groups of components that correspond to the characters of a text line without making any a-priori strong assumption. The single assumption that we will have to make is that the characters that belong to the same word are similar. However, defining in what way they are similar is not a trivial issue.

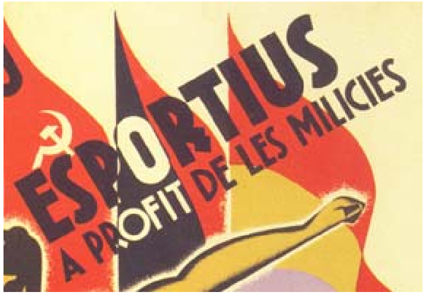


Figure 3. Example of text comprising different coloured characters.

Although the colour of individual characters is constant, this is not always the case when whole words or text lines are examined. For example there are words that comprise letters of different colour (see Figure 3). As a result, colour similarity is not used here as a feature for text line identification.

Similarity is instead defined purely on spatial and topological characteristics. Specifically, characters of the same word are required to have similar height and thickness, and to be placed at uniform intervals along the text line.

As will become clear next, the notion of height is dynamic, as it is always defined based on the direction of the text line being assessed. In addition, a text line is not assumed to be a straight line, instead it dynamically follows the character components.

3.3.1. Seed Components Selection. Our starting hypothesis is that every connected component can potentially correspond to a character. If this is the case, then there has to exist a second component also corresponding to a character in close proximity to the first one that shares similar height and thickness characteristics. Each such pair of components is used as a seed for a potential text line. For the purpose of this work, thickness is defined as the ratio of the area to the perimeter of a component. This is not a precise definition but it serves well for our purposes.

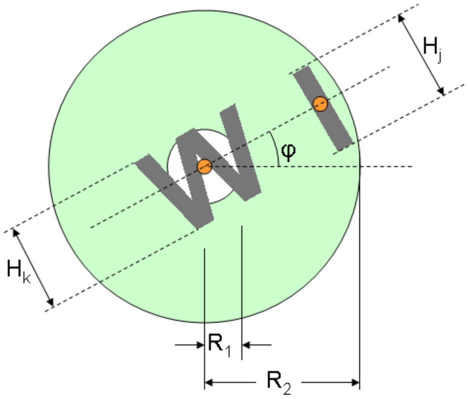


Figure 4. Search space and definition of height.

A donut shaped area with $R_1 = 0.2 \cdot D$ and $R_2 = 2 \cdot D$ (where D is the diagonal of the original component) around the centre of the bounding box of the original component is searched, and components that satisfy the following equations are identified.

$$\frac{T_k}{1.5} \leq T_j \leq 1.5 \cdot T_k \tag{1}$$

$$\frac{H_k}{1.5} \leq H_j \leq 1.5 \cdot H_k \tag{2}$$

Where T and H are the thickness and height of the component, k is the reference component, and j is the component being examined. The height of a component is defined as the projection of the component on the orientation of the line connecting the two components as shown in Figure 4.

3.3.2. Text Line Building. Given the initial line segment specified by the centres of the seed pair of components, the algorithm searches symmetrically on both sides of the line segment on the orientation defined by the segment for more components that could belong to the text line. The search space is defined based on the distance of the two initial components (*d_0*), and the orientation of the line segment as shown in Figure 5. An angle threshold of *θ_0=35°* and a maximum distance of *$1.5 \cdot d_0$* is used to specify the search space.

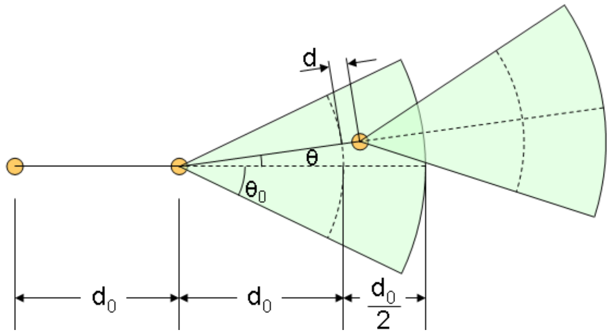


Figure 5. Adding components to the text line.

All the components that lie within the search area specified are checked and the ones that have dissimilar height or thickness are discarded. For this test, the same equations (1) and (2) are used as before. The components that pass this test are ranked based on how far they lie from the expected location and the expected direction based on the following equation.

$$S = W_\theta \cdot \frac{\theta}{\theta_0} + W_d \cdot \frac{d}{d_0} \qquad (3)$$

The weights used are *W_θ=0.7* and *W_d=0.3*. The component that yields the lowest score is chosen as the next component of the line, and the search process is repeated, this time centred on the new component as shown in Figure 5.

3.3.3. Text Line Selection. Following the above process a number of potential text lines can be identified for each component (one for each seed-pair built with the component). These text-lines are examined and either one or none is selected as the most probable text line for the component (since each character can only belong to a single line).

This is achieved by first filtering all the text lines that comprise less than 3 components. A further filtering is then performed based on the inter-component distance; that is the distance between successive components on the line. We ask that the maximum inter-component distance is no more than 1.9 times the minimum inter-component distance. A factor of less than 2 was deliberately chosen as otherwise it is possible to obtain lines that jump over characters (select every second character on the line).

Finally, of all the lines that passed the filters above we choose the one that has the minimum average inter-component distance. This process (steps 3.3.1 to 3.3.3) is repeated for every component in the image, and the final set of text-lines produced by the algorithm is returned as the textual content of the image.

4. Results and Discussion

We defined pixel-level ground-truth information (each pixel labelled as text or background) for 50 of the images of our dataset. The result of the algorithm was assessed based on the ground-truth and metrics for precision, recall and fall-out were calculated based on the following equations.

$$P = \frac{|retrieved \cap relevant|}{|retrieved|} \qquad (4)$$

$$R = \frac{|retrieved \cap relevant|}{|relevant|} \qquad (5)$$

$$F = \frac{|retrieved \cap \overline{relevant}|}{|\overline{relevant}|} \qquad (6)$$

“Retrieved” is the set of the pixels that belong to the components of the text lines identified by our algorithm. “Relevant” is the set of pixels that belong to characters as specified in the ground-truth while “Non-Relevant” the set of pixels of the background.

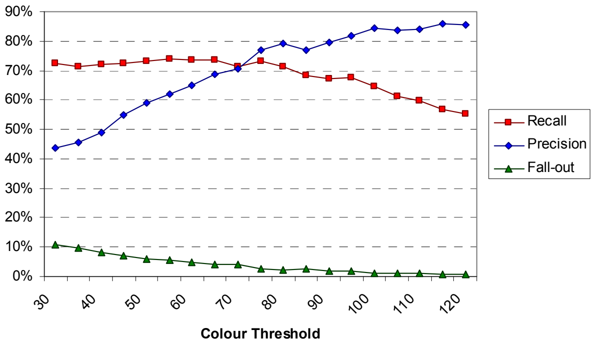
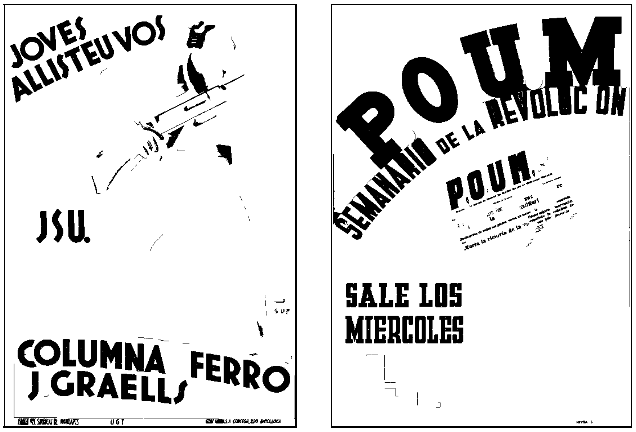


Figure 6. Precision, Recall and Fall-out for different colour thresholds.

We repeated the experiment for different values for the colour threshold used for segmentation (see section 3.1) and in each case we calculated the precision, recall and fall-out values. The results are plotted in Figure 6. As the threshold increases, less noisy segmentations are produced and the algorithm is not tricked by combinations of noise components that resemble text lines, therefore precision increases correspondingly and fall-out drops. After a given threshold though, characters start being merged with the background, and recall therefore drops.

Here we report results using a threshold of *75*, as a good trade-off between precision and recall. Depending on the requirements of the final application a different threshold could be selected. . For this threshold we obtain *P=76.9%*, *R=73.2%* and *F=2.6%.*



(a) (b)

Figure 7. Final results for the images of Figure 1.

Some image results are shown in Figure 7. The described algorithm is able to identify individual text lines. Nevertheless, the evaluation performed here focuses on the segmentation aspect, and disregards the text line information. It is reasonable to question this choice, as alternatively we could attempt to OCR each individual text-line and report on the OCR success. This is not as straightforward as it seems though as further post-processing has to take place (e.g. unwrapping curved text lines, joining words that have been identified as separate "text-lines", updating the OCR dictionary with appropriate Catalan terms and acronyms required by the historical context of this collection etc.). This is intended future work.

It is also difficult to perform a direct comparison to other existing methods, as generally results reported are obtained through visual assessment or bounding-box based ground truth is used (which is only adequate for assessing text localisation but not segmentation).

5. Conclusions

A method was described to extract the textual content from complex colour images. The proposed method makes no assumptions as to the size, orientation or colour of the characters, while it is tolerant to characters that do not follow a straight baseline. We evaluated our method on a dataset of poster images from the Spanish Civil War and achieved promising segmentation results with precision and recall above 70%.

6. Acknowledgements

This work was partially supported by the Spanish projects TIN2008-04998, CONSOLIDER INGENIO 2010 (CSD2007-00018), the fellowship 2006 BP-B1 00046 and a fellowship under the scheme 2009 FI-DGR. The images used were kindly provided by the National Archive of Catalunya.

7. References

[1] K. Sobottka, H. Kronenberg, T. Perroud, and H. Bunke, "Text extraction from colored book and journal covers ", *International Journal on Document Analysis and Recognition*, Springer, Vol. 2, 2000, pp. 163-176.

[2] H. Hase, M. Yoneda, S. Tokai, J. Kato and C.Y. Suen, "Color segmentation for text extraction", *International Journal on Document Analysis and Recognition*, Springer, Vol. 6. 2004, pp. 271-284.

[3] C. Strouthopoulos, N. Papamarkos and A.E. Atsalakis, "Text extraction in complex color documents", *Pattern Recognition*, Elsevier, Vol. 35, 2002, pp. 1743-1758.

[4] D. Karatzas and A. Antonacopoulos, "Colour text segmentation in Web images based on human perception", *Image and Vision Computing*, Elsevier, Vol. 25(5), 2007, pp. 564-577.

[5] V. Wu, R. Manmatha and R.M. Riseman, "TextFinder: An automatic system to detect and recognize text in images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21(11), 1999, pp. 1224-1229.

[6] C. Jung, Q. Liu and J. Kim, "A new approach for text segmentation using a stroke filter", *Signal Processing*, Elsevier, 2008, Vol. 88, pp. 1907-1916.

[7] T. Retornaz and B. Marcotegui, "Scene text localization based on the ultimate opening", Proceedings of the 8th International Symposium on Mathematical Morphology, 2007, pp. 177-188.

[8] H.K. Kim, "Efficient Automatic Text Location Method and Content-Based Indexing and Structuring of Video Database", *Journal of Visual Communication and Image Representation*, Elsevier, Vol. 7, 1996, pp. 336-344.

[9] D. Karatzas, "Text Segmentation in Web Images Using Colour Perception and Topological Features", PhD Thesis, University of Liverpool, UK, 2002.