Compound Image Segmentation for Multiscale Recurrent Pattern Coding

Luís Lucas, Nuno M. M. Rodrigues, Sérgio M. M. de Faria

Instituto de Telecomunicações, Portugal; ESTG, Instituto Politécnico de Leiria, Portugal;
e-mails: luisflucas@hotmail.com, nuno.rodrigues, sergio.faria@co.it.pt.

Abstract **– In this paper we propose a new compound image segmentation algorithm, specially developed for a digital document encoding scheme based on the Multidimensional Multiscale Parser (MMP) encoding algorithm. In order to increase the MMP algorithm performance, a block based segmentation defines a mask that divides the input image into two types of blocks: text-like blocks and non-text, or smooth image-like blocks. Different versions of the MMP algorithm are then applied on each image layer.**

The segmentation algorithm presented in this paper is based in two steps. The first step relies on morphological filtering and image gradient analysis. In the second step, false text blocks are eliminated using a morphological based algorithm. Experimental results show that this algorithm is not only efficient for our encoder, but also improves the segmentation accuracy in relation to other methods proposed in the literature.

I. Introduction

Automatic segmentation of compound images plays an important role in block classification-based compound image encoding schemes. These methods divide the image into different regions, that are independently compressed with different encoding algorithms. Typical compound image segmentation algorithms reported in literature [1]-[4], aim to separate the text and background (smooth) components of the image. In [1], a simple block-based segmentation algorithm is presented, where texture modes are analysed using block histograms. In [2], the gradient ranges in the pixels of each block are used for classification. These are fast methods, due to their block based nature, but experimental tests revealed some inefficiencies. The algorithm in [1] fails for documents with small text, grey (non-black) text or noisy background images. The gradient based algorithm performs better, but also fails when text gradient is low, or when the smooth portions of the images present high gradient. Practical tests reveal that these are common on scanned documents.

There are also other algorithms proposed for text detection, that are not block-based. A morphological text segmentation algorithm [4] presented generally better results but still had some limitations. This was the case when very small or very large text characters are used, or when very short, isolated strings exist in the document. In [3], strings of text are extracted by scanning and grouping connected components. In this algorithm, there exist some limitations in inter-characters gap, characters resolution, among others. Another limitation of some of these methods is that they are not able to detect white text in dark backgrounds.

In this research we intend to implement an automatic block based segmentation algorithm, that is both generally efficient but also suited to create a segmentation mask for compound document encoding method, based on the Multidimensional Multiscale Parser (MMP) algorithm [5], [6]. MMP is a block-based encoder. In our encoding scheme we use two MMP-based encoders: one optimised for text-like block compression and other for compressing smooth blocks [7].

Typical segmentation algorithms [1], [2] strongly depend on the image's features. When the quality of the documents is poor, the resulting mask has a very high error rate. Other existing methods are designed mainly for Optical Character Recognition (OCR) and are not efficient for MMP usage. A study of the MMP encoder's features revealed that the segmentation process should be able to detect text blocks, even for noticeable background noise. Also it is important to detect graphics regions as text blocks, that will be processed by an MMP encoder specifically design for blocks with high pixel variations. A study of some previously proposed methods revealed some inefficiencies in both these requisites, which led us to the investigation of a new, more efficient method.

The proposed segmentation algorithm uses a first stage, based on morphological filters for image enhancement and block gradient analysis of the enhanced document. In a second stage, some inadequate classifications as text blocks are corrected by using an algorithm that determines if isolated groups of blocks detected as text are actually text.

Section II presents a description of the proposed two-stage segmentation algorithm. Section III discusses some experimental results, where the efficiency of this segmentation is shown. Finally, section IV presents some conclusions about this work.

II. Segmentation Algorithm

This section presents a description of the two-stage segmentation algorithm, represented in the flowchart of Figure 1. The first stage corresponds to a set block-based steps and the second stage is a segmentation refinement step that is not block based.

A. First stage of segmentation

Two morphological filters are first applied to the gray-level global image, in order to enhance it for segmentation. These filters are denominated *bothat* and *tophat* filters and they are based on dilation and erosion operators [8]. A closed image is obtained when a dilation is followed by an image erosion, using a common structuring element (in our case, square of 7×7 pixels was used). An opened image is the result of an image erosion followed by a dilation, using the same structuring element for both operations. A *bothat* filtering corresponds to a subtraction of the original image from a morphologically closed version of the image. This filter is used to remove a bright background from a dark foreground. In case of text, background is well removed if it is clearer than text. Images areas that have smooth variations of pixels do not have foreground, so they are successfully removed. Image areas with white text over a dark background are successfully identified by using a *tophat* filter. *Tophat* filtering is a subtraction of a morphologically

This work has been supported by Fundação Para a Ciência e Tecnologia (FCT), under project PTDC/EEA-TEL/66462/2006

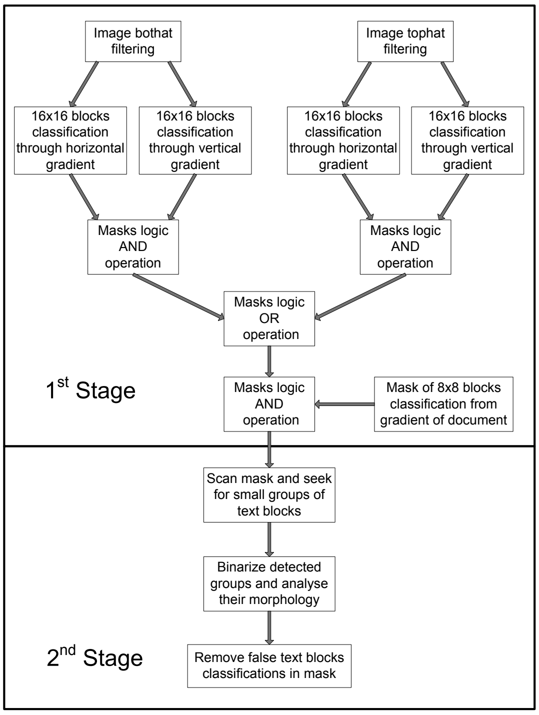

Figure 1 - Segmentation algorithm flowchart.

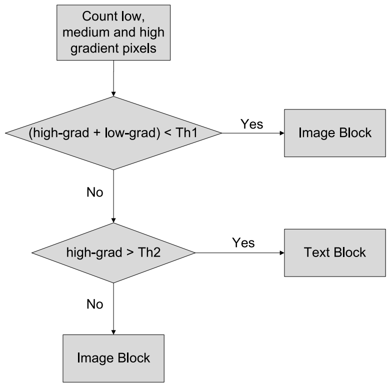

Figure 2 - Flowchart of gradient based algorithm.

opened image from the original image.

Our segmentation algorithm applies both morphological filters separately, in order to allow detection of text regions with different colours.The gradient of each image block is then used for block classification. The gradient of a function of two variables, $F(x, y)$ is defined as

$$\nabla F = \frac{F_x}{\partial x}\hat{i} + \frac{F_y}{\partial y}\hat{j}. \quad (1)$$

This stage of the algorithm is an adaptation of [2]. It divides the resulting gradient matrix into three levels of gradient pixels, defined by a threshold: low, medium and high gradient pixels. A text block is characterised by high gradient pixels between letters and background transitions, and low gradient pixels inside letter symbols or in background. An image block has mainly low gradient pixels or medium gradient pixels due to its smooth behaviour. These features must happen in horizontal and vertical gradient directions, on both text and image blocks. In order to satisfy these conditions the vertical and horizontal gradients are calculated in the full image, and low, medium and high gradient pixels are counted in each block for both gradient directions.

The classification algorithm for each calculated gradient is related to the flowchart presented in Figure 2, where *Th1* is 60% and *Th2* is 1%. Detection of hybrid and smooth blocks, used in the original algorithm, was ignored since is not important for our approach. The levels threshold we set so that low gradients have values lower than 10 while high gradients have a threshold equal to 35, considering a grayscale document with 8 bits resolution. This high gradient threshold was set to a small value in order to successfully detect text regions with poor sharpness.

The algorithm performs better for large block sizes, that provide more accurate gradient values. Nevertheless increasing the block size makes it more probable for a block to contain both text and image, compromising the coding efficiency. A block size of 16×16 pixels was chosen as a compromise between better segmentation and coding efficiency for documents with small characters.

The two masks that resulted from the use of the horizontal and vertical gradients are then combined using a logical AND operator. Since a text block corresponds to a 1 in the mask and an image block corresponds to a 0, this means that a block is considered text only if the corresponding block on both calculated masks is a text block.

Since this gradient algorithm is used for both *bothat* and *tophat* filters, two masks are created using the previously described method. These two different masks are again combined into a new mask using a logic OR operation. This means that a block will be classified as text if it is marked as text in any of the previous masks. Since MMP may use both 8×8 and 16×16 blocks, our segmentation should originate a 8×8 block mask. In order to pass from a 16×16 to a 8×8 mask an efficient algorithm was investigated. The horizontal and vertical gradient of the original document are calculated, and for each 8×8 block it is verified if exists any high gradient pixel. A 8×8 block is considered as text if both horizontal and vertical gradients of the original image in that block have, at least, one high gradient pixel. Using this procedure all text regions are detected, but some other zones with high pixels' variation are also classified as text. We perform a logic AND operation between 16×16 mask and the calculated 8×8 pixel blocks mask. This means that any original 16×16 image block originates four image blocks in a new 8×8 mask, because of the AND operation. One 16×16 text block may originate from one to four text blocks in the new 8×8 mask, depending on previous calculated 8×8 mask.

B. *Second stage of segmentation*

The first stage of the algorithm is composed by a set of low complexity steps, that achieve reasonable segmentation results. Nevertheless, for some smooth image regions with high pixel variations we still obtain blocks classified as text blocks. In order to overcome this situation we have developed a refinement procedure, that is described in this subsection.

The used procedure is based on the algorithm presented in [3], that is used to detect connected components in an image. Our algorithm works in a similar fashion, but the objective is to detect and delete false text block candidates. Analysing the 8×8 block mask, that results from the first stage of the algorithm, misclassified text blocks are still noticeable. These blocks appear mainly isolated or in small groups. In order to identify these blocks, we first analyse the mask from first stage. Then we detect groups of 4-connected text blocks, with less than T_{text} blocks (in our case $T_{text} = 300$). A region is considered as being *4-connected* if all of its elements are connected either to an horizontal or a vertical neighbour.

Each 4-connected group of blocks is then surrounded by a rectangular box, thus defining a region denominated R_n, with $n = 1, \ldots, Number\_regions$. The following steps describe the algorithm used for correcting misclassified blocks. This method is based mainly in the morphology of each region R_n.

For each rectangular region of the original image, I_n, corresponding to the blocks defined by each R_n:

1. Determine I_n^B, a binarised version of I_n, using a threshold T_B. The value of the threshold is adaptively determined for each block using Otsu's method [9];
2. Invert I_n^B if the number of white pixels is larger than the number of black pixels. This step assures that the final region has black background and white foreground, with the background being the dominant region of I_n^B;
3. Using an 8-connected criterium, determine each independent group of white pixels in I_n^B. A region is considered as being *8-connected* if all of its elements are connected to *any* of their neighbours. Let D_m be each of the groups identified in this step;
4. Circumscribe each D_m object with a rectangular box, in order to define a new region D_m^r. Note that D_m^r will thus have black and white pixels;
5. Discard all D_m^r objects with less than 5 pixels;
6. Let T_B be the actual number of text blocks in R_n and T_D be the number of D_m^r objects in R_n. If any of the conditions:
   (a) $T_B < 25$ and $T_D \geq 1$;
   (b) $25 \geq T_B < 100$ and $T_D \geq 2$;
   (c) $T_B \geq 100$ and $T_D \geq 5$;
   is true, go to step 7, else classify R_n as image and return;
7. Apply a refinement procedure to R_n (explained next) and confirm R_n as a text region or classify it as an image area.

The refinement used in step 7 is very important to the overall performance of the method. It is based on a set of tests that classify R_n as image if:

1. R_n only has one small object (D_m has less than 50 pixels);
2. R_n has several (more than 5) objects D_m with one of them having more than 50% of the total area of R_n and being composed by more than 30% of white pixels;
3. R_n has less than 6 objects and more than 30% of its border pixels are white;
4. R_n has more than 5 objects but all of them touch at least three margins of R_n;
5. more than one third of the total number of objects correspond to isolated white pixels;
6. more than one third of the total number of objects correspond to isolated black pixels;

Step 1 classifies as image, regions with only one small object. Step 2 identify large objects, that ordinarily would not be text regions, but may correspond to tables or other graphic areas if only a few white pixels are present. Experimental observations revealed that image objects tend to cluster around the margins of R_n, specially in small areas. On the other hand, text objects are rarely connected to three or more of the margins of R_n. These observations led to the implementation of steps 3 and 4 of the refinement procedure. Steps 5 and 6 eliminate regions that were previously considered as text due to a high number of isolated white or black pixels. If all of these conditions do not occur, the region continues to be classified as text. All used threshold values were based in experimental tests with a different number of compound, natural and text images.

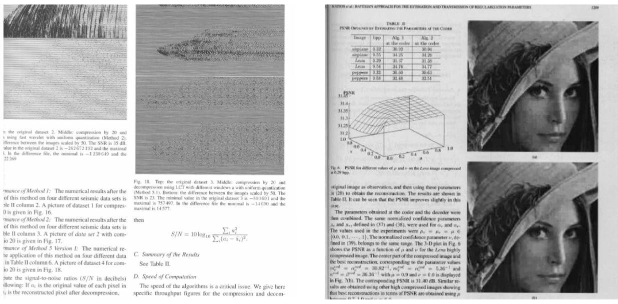

Figure 3 - Test images scan_0009_bottom (left) and PP1209 (right).

The final segmentation mask uses 8×8 blocks. Since MMP works primarily with 16×16 blocks, we have implemented an additional step to convert 8×8 blocks to 16×16 blocks. If a square group of four 8×8 blocks has two or more text regions, the corresponding 16×16 block is considered as a text block. If only one of the four 8×8 blocks is a text block, but it is connected to other(s) in its neighbourhood, the corresponding 16×16 block is also considered as a text block. Else, the 16×16 is classified as an image block.

III. Experimental Results

In this section, we present the experimental results for the proposed segmentation method. The results for two compound images (represented in Figure 3) are presented. Both images were scanned from a paper version: image *PP1209* was scanned from page 1209 of the *IEEE Trans. on Image Proc.*, vol. 9, no. 7, Jul. 2000, and image *scan_0009_bottom* was scanned from page 1801 of the *IEEE Trans. on Image Proc.*, vol. 10, no. 12, Dec. 2001. These and other used test images are available for download at *www.estg.ipleiria.pt/∼nuno/MMP*.

Figure 4 shows the segmentation results for image *scan_0009_bottom*, comparing the output of first stage with the final segmentation, for 8×8 blocks. These results show that the first stage of our algorithm detects all text zones, but some image areas are misclassified as text. After the second stage, the block classification is improved, due to the previously described refinement procedure. Note that this particular image is not very simple to segment, since many image areas have higher gradient than the text zones. This is the reason for the misclassifications resulting from the algorithm proposed in [2], that may be observed in Figure 4 (right). Nevertheless, the proposed algorithm achieved very good segmentation accuracy.

In Figure 5 we show the results for image PP1209. This test image has been traditionally used in MMP tests [5]-[7] and is characterised by a severe noise level in the background region. This is the case for many images resulting

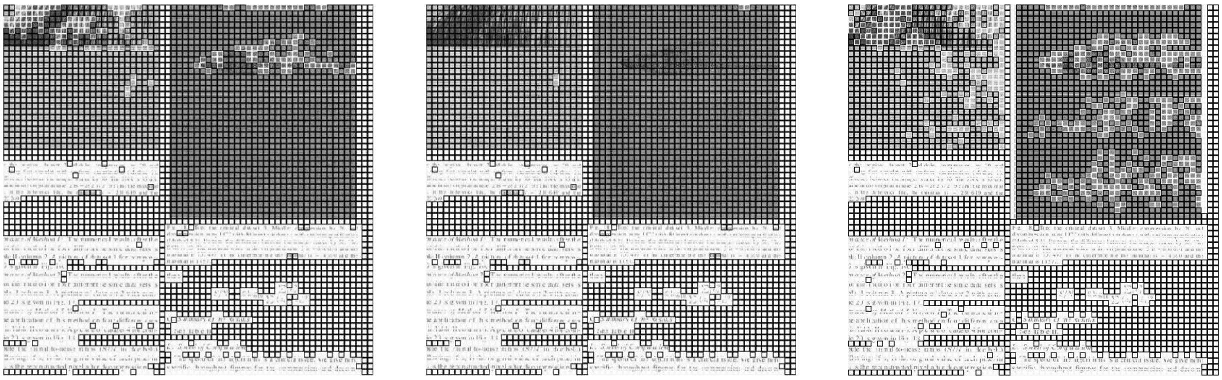

Figure 4 - Segmentation results for image scan_0009_bottom: (left) stage one of the proposed algorithm; (center) stage two of the proposed algorithm; (right) result of [2]. In these images, white squares represent text blocks while black squares represent image areas.

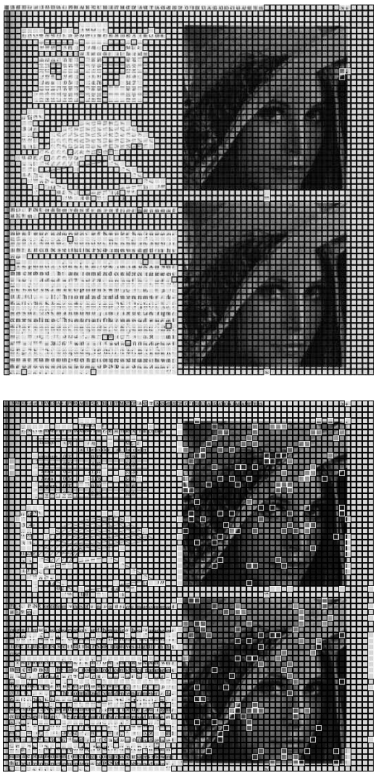

Figure 5 - Segmentation of image PP1209 using the proposed algorithm (top) and the method in [1] (bottom).

from document scanning. For these images, the histogram's modes based algorithm [1] has shown severe classification errors. This is not the case for the proposed method.

Note that the graphic area of the image has been classified as text blocks, due to the text-like features of the histogram and gradient of this area. In order to determine if in fact these were text blocks in stead of graphics, one could apply an OCR algorithm to detect text characters. Nevertheless, for our purpose it is better to classify this region as text blocks. The presence of high gradient pixels and the absence of a structure that is adequate to the use of predictive techniques mean that better results will be achieved by the MMP encoder optimised for text regions.

IV. Conclusion

In this paper we propose a new efficient segmentation algorithm for compound images. The proposed method combines the ideas of morphological-based and gradient-based classification schemes. The new method circumvents the typical errors observed in these algorithms and produces good segmentation results even for noisy images. Also, a new refinement procedure was introduced in order to eliminate some classification errors associated with text regions.

Experimental results show that the proposed method produces efficient segmentation masks to be used in a multiscale recurrent pattern segmentation-based compound image encoder.

References

[1] Xin Li and Shawmin Lei, "Block-based segmentation and adaptive coding for visually lossless compression of scanned documents", *IEEE International Conference on Image Processing*, vol. 3, pp. 450–453, October 2001.

[2] W Ding, D Liu, Y He, and F Wu, "Block-based fast compression for compound images", *International Conference in Multimedia & Expo*, pp. 809–812, 2006.

[3] L. A. Fletcher and R. Kasturi, "A robust algorithm for text string separation from mixed text/graphics images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 6, pp. 910–918, November 1988.

[4] Y. M. Y. Hasan and L. J. Karam, "Morphological text extraction from images", *IEEE Transactions on Image Processing*, vol. 9, no. 11, pp. 1978–1983, November 2000.

[5] M. de Carvalho, E. da Silva, and W. Finamore, "Multidimensional signal compression using multiscale recurrent patterns", *Elsevier Signal Processing*, , no. 82, pp. 1559–1580, November 2002.

[6] N. M. M. Rodrigues, E. A. B. da Silva, M. B. de Carvalho, S. M. M. de Faria, and V. M. M. da Silva, "On dictionary adaptation for recurrent pattern image coding", *Image Processing, IEEE Transactions on*, vol. 17, no. 9, pp. 1640–1653, Sept. 2008.

[7] N. C. Francisco, R. N. R. Sardo, N. M. M. Rodrigues, E. A. B. da Silva, M. B. de Carvalho, S. M. M. de Faria, Vitor M. M. Silva, and M. J. C. S. Reis, "A compound image encoder based on the multiscale recurrent pattern algorithm", *SIGMAP2008 - International Conference on Signal Processing and Multimedia Applications*, July 2008.

[8] J. Serra, *Image Analysis and Mathematical Morphology*, New York: Academic, 1982.

[9] N. Otsu, "A threshold selection method from gray-level histograms", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.