



HAL
open science

Watercolor, segmenting images using connected color components

Sébastien Eskenazi, Petra Gomez-Krämer, Jean-Marc Ogier

► **To cite this version:**

Sébastien Eskenazi, Petra Gomez-Krämer, Jean-Marc Ogier. Watercolor, segmenting images using connected color components. International Conference on Pattern Recognition, Aug 2018, Beijing, China. hal-01900022

HAL Id: hal-01900022

<https://hal.science/hal-01900022>

Submitted on 20 Oct 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Watercolor, segmenting images using connected color components

Sébastien Eskenazi
L3i, University of La Rochelle
Avenue Michel Crépeau
17042 La Rochelle, France
Email: sesken02@univ-lr.fr

Petra Gomez-Krämer
L3i, University of La Rochelle
Avenue Michel Crépeau
17042 La Rochelle, France
Email: petra.gomez@univ-lr.fr

Jean-Marc Ogier
L3i, University of La Rochelle
Avenue Michel Crépeau
17042 La Rochelle, France
Email: jean-marc.ogier@univ-lr.fr

Abstract—In the context of document security systems, there is a growing need for a stable segmentation method. State-of-the-art document image segmentation methods are not stable as they use several parameters and thresholds such as binarization. Hence, this paper presents a new method for segmentation based on a new definition of connected color components and a new model of human vision. Our algorithm produces results that are three to four times more stable than state-of-the-art superpixel segmentation methods while maintaining a similar segmentation accuracy.

I. INTRODUCTION

Document forgery is becoming a significant issue due to the easy use of image processing software. It does not only concern passports and identity cards, but also proofs of address such as electricity bills and birth certificates to obtain identity documents, pay slips and tax assessments to obtain consumer credits, or business documents such as invoices or contracts. These documents frequently change between paper and digital format, creating different versions of a document (the original digital document, the printed document, the printed and scanned document...). Ensuring the security of all these versions of a document is called hybrid security. In the context of the content-based signature for hybrid document integrity check [1], there is a need for a stable segmentation algorithm as it is the basis for document image analysis methods (layout, text and graphic analysis) with the objectif of obtaining a robust hash of the document's content.

The principle of a stable algorithm is that the algorithm is capable to produce similar (respectively dissimilar) outputs given similar (respectively dissimilar) inputs. Given a function (the algorithm) $f : X \mapsto Y$, a binary similarity function s_X for its input space X and a binary similarity function s_Y for its output space Y , then f is stable with respect to s_X and s_Y if and only if

$$\forall \{x_1, x_2\} \in X^2, s_X(x_1, x_2) = s_Y(f(x_1), f(x_2)) \quad (1)$$

Here, s_X defines if two documents are similar modulo the print and scan noise and s_Y defines if two segmentation results are similar. In contrast to accuracy, which is usually used for evaluating algorithms, stability does not need any ground truth. Anyhow, stability is often a trade-off with respect to accuracy.

The authors of [2], have shown that current document segmentation algorithms are not stable. The main issues of

instable document image processing algorithms are parameters and thresholds such as binarization [3]. Thus, we present here a new parameter-free color image segmentation algorithm. It is based on an extension of connected components to color images: connected color components (CCCs). Our algorithm performs similarly to the state of the art in terms of accuracy, but it is three to four times more stable.

Our contributions are the following: 1) We extend the definition of connected components to color images and present an algorithm to segment an image into CCCs. 2) To this end, we present a new model of human vision that, to our knowledge, contains more features than existing ones. Notably, it represents the gradient sensitivity of the human eye and its spatio-colorimetric sensitivity.

The remainder of this paper is organized as follows. We first present the state of the art of superpixel algorithms in Section II. In Section III, we formalize the notion of connected color components. Then, we present in Section IV a new model of human vision for the CCC segmentation. Based on this model, we develop in Section V a CCC segmentation algorithm, watercolor, together with a post-processing method. Finally, we show some results in Section VI and compare them to the state of the art before discussing and concluding our work in Section VII.

II. STATE OF THE ART

The nearest segmentation algorithms to what we are proposing are superpixel algorithms. They try to optimize an objective function of the form:

$$E(S) = H(S) + C(S) \quad (2)$$

where S is a partition of the image space, H is a function estimating the homogeneity of the regions of the partition and C is a function estimating their compactness. Furthermore, they frequently add another constraint, which is the number of superpixels to produce. This creates two internal contradictions. Firstly, many homogenous regions are not compact. This makes it impossible to satisfy (2) with a proper partition. Secondly, the number of homogenous regions in a given image is unknown a priori and may vary from one image (or document) to another. Hence, defining the number of superpixels beforehand will most likely lead to a wrong

number of superpixels. These issues are even more acute, when considering document images. We would expect to have one superpixel per printed character, but we do not know the number of characters in the image. Regarding the compactness constraint, characters are far from having a compact or disk-like shape.

Superpixel algorithms have been surveyed in [4]. Moreover, Achanta et al. [5] provide a concise overview of the main techniques of the field and benchmarks them, and Stutz [6] does a thorough review and benchmark work and concludes that superpixel technologies are completely mature for 2D still images. Roughly, they all tend to ascend or descend a gradient. The only exception is the case of SLIC superpixels [5] and the algorithms that use it, [7] and [8], as SLIC is based on k-means clustering.

As the definition of the number of superpixels in advance is likely to produce a wrong number of superpixels, we can discard the algorithms that require it. Among the remaining algorithms, the algorithm of [9] has seven parameters, which will make it very difficult to train. Finally, the only remaining algorithms that are relevant to our problem are those of [10]–[12]. Anyhow, these algorithms make a trade-off between geometric and color distance and as a consequence have difficulties to adapt to the variations of the object size. This is why, we base our algorithm on a model of human vision that we describe below.

III. CONNECTED COLOR COMPONENTS

In a binary image a connected component is a connected area with uniform color. When considering color or gray level images, the definition of a connected region remains the same. However, because of the print and scan noise in the document image, it would not be relevant to consider the regions with a strictly uniform color such as exactly the same RGB values for color images. Hence, uniform colors are considered to be what a human observer sees as uniform colors. Furthermore, there can be color or intensity gradients that can mark boundaries or not.

Finally, the goal of our algorithm is to produce a set of connected regions, where each region has a perceptually uniform color or is a continuous color gradient i.e. we want to produce the largest regions that do not contain any visually significant boundaries inside them. From a mathematical point of view, we want to create a partition of the image space I into a set of regions $\{S_i\}$. We consider each S_i to be open and to be a partition of I . $\{S_i\}$ must satisfy the following condition:

$$\bar{I} = \bigcup_i \bar{S}_i \text{ and } \forall i, j, i \neq j, S_i \cap S_j = \emptyset \quad (3)$$

where $\bar{\cdot}$ denotes the topological closure operator. The difference to common definitions of a partition is that we enforce the fact that the S_i are open. This means that the regions contain the pixel areas, but not the pixel borders. Based on this, neighbor regions can be defined. Let u and v be two arc connected regions. They are neighbor regions if and only if

$$u^\circ \cap v^\circ = \emptyset \text{ and } \bar{u} \cap \bar{v} \neq \emptyset \quad (4)$$

where $^\circ$ denotes the interior operator.

Let I be an image space and $\{S_i\}$ a partition of it, then $\{S_i\}$ is a CCC partition of I if and only if

$$\forall S_i, \forall \{u, v\} \subset S_i^2, \phi(u, v) \leq t \\ \forall i \neq j, \phi(S_i, S_j) > t \quad (5)$$

where u, v are two neighbor regions. ϕ is a perceptual color distance function and t is a threshold above which a human observer would consider that there is a significant color difference. They are presented in Section IV.

Compared to a superpixel definition, the main difference lies in the absence of any shape, number or other constraint that would antagonize the color uniformity constraint. This leads to a better posed problem. Furthermore, our approach is different from blob detectors such as MSER. MSER methods are based on the use of multiple thresholds, e.g. on gray scale images or in the watershed computation, and check for regions that remain stable over several thresholds.

IV. HUMAN VISION MODEL

The idea of a human vision model is to remove insignificant elements and to extract the information as the human eye sees it. Several studies have been done to understand and model the inner workings of the human eye. An overview of the field can be found in [13]. The main model of human vision that has been proposed in computer vision is that of [14], but it only provides a saliency map of the image without any information on the perceived colors, regions or gradients.

Our guiding principle is not to reproduce the physical mechanisms of the human eye and the human brain, but rather to find image processing algorithms with similar sensitivity. Our model encompasses four characteristics of the human eye: its spatial sensitivity, its colorimetric sensitivity, its spatio-colorimetric sensitivity, and its contrast sensitivity.

A. Spatial sensitivity

Optometrists have studied the spatial acuity of the human eye [15]. The minimum separable is its separation power e.g. the minimum distance between two points that we can distinguish. The Ophthalmology Congress of Naples in 1909 set the normal minimum separable to one arc-minute.

For a scan of a document, the angular minimum separable a can be converted into dot per inch (dpi) resolution using the reading distance d in centimeters as $dpi = \frac{2.54}{d \times \tan(a)}$. The reading distance is typically between 30 and 40 cm yielding a resolution between 200 and 300 dpi.

B. Colorimetric sensitivity

The human spectral eye sensitivity does not match the spectrum of an RGB image. Furthermore, there are several RGB color spaces. Here, we will consider that all images use the sRGB color space. This is motivated by the fact that recent screen technologies are calibrated for the sRGB color space with negligible error.

The most perceptually uniform color space is the Lab color space. It is a non-linear color space, which is defined

in the ISO/CIE norm 11664-4:2008 (CIE S 014-4/E:2007). According to [16], the just noticeable color difference in the Lab space with the Euclidean distance is of approximately 2.3. The number of discernible colors is of approximately 2 million colors [17]. Thus, a 24 bit color representation should be sufficient to represent accurately all visible colors. Considering all this, we will convert sRGB images to the Lab color space and use the Euclidean distance to measure color differences.

C. Spatio-colorimetric sensitivity

It is easier to differentiate large patches of colors rather than small color dots. Hence the size of the CCCs and the scale of color variations influence our perception of color differences. The variation of color perception with the size of the stimuli (the color region) has been already studied [18]–[20], but only for regions with a viewing angle of at least 1° . This is at least 30 times larger than the minimum separable (Section IV-A).

We can expect that the ability to distinguish colors is relative to the amount of light from that specific colored region received by the eye sensors. Hence, it should be a combination of light intensity and the area of the light emitting region. Assuming that two regions are illuminated by the same illuminant under identical conditions, the light dependency is already taken into account in the Lab color space. Thus, only the dependency with the region area remains. We assume that above a certain region area the maximal eye color discrimination is reached and thus the color distance should not change. The spatio-colorimetric distance is:

$$\phi(u, v) = \left[\min \left(\left(\frac{S(u)}{S_L} \right)^2, 1 \right) \cdot \Delta L(u, v)^2 + \min \left(\left(\frac{S(u)}{S_a} \right)^2, 1 \right) \cdot \Delta a(u, v)^2 + \min \left(\left(\frac{S(u)}{S_b} \right)^2, 1 \right) \cdot \Delta b(u, v)^2 \right]^{1/2} \quad (6)$$

where ΔL , Δa and Δb are the differences between the values of the corresponding Lab channels for two neighbor regions u and v . It varies linearly with the region area S . It is an adaptation of the Euclidean distance with three parameters, S_L , S_a and S_b , that correspond to the region size above which one can clearly distinguish colors.

We validated this model with several experiments that have shown that S_L , S_a and S_b are all below half a degree. In the following we use $S_L = 17.5$, $S_a = 27.2$ and $S_b = 89.1$ square pixels at 100 dpi and scale them with the processing resolution. This is the function ϕ of (5). Since it extends the Euclidean distance, the threshold t should be equal to 2.3 (see Section IV-B).

D. Contrast sensitivity

The human eye incorporates an edge detector/filter and thus is very sensitive to edges. On the opposite its acuity is reduced for uniform regions. A model was proposed in [21] for the contrast sensitivity. However it involves 9 parameters and only deals with one dimensional stripe patterns. Adapting it to a 2D set of regions of arbitrary shape would be extremely difficult without a lot more experimental data.

However, the field of image processing has recently seen the apparition of filters with properties similar to those of the human eye: edge-preserving filters. After reviewing the main algorithms, e.g. [22]–[29], we found that the domain transform filter [29] represents the best the contrast sensitivity of the human eye. We use three iterations, a spatial range of approximately 1 cm and a value range of 0.2.

V. WATERCOLOR CCC SEGMENTATION

Our CCC segmentation is based on the watershed algorithm, hence its name watercolor. We also provide a post-processing algorithm which uses our spatio-colorimetric distance ϕ to merge regions of similar colors.

A. Watercolor

The general idea is to apply the watershed algorithm, which is a parameter-free segmentation algorithm, on the inter-pixel color distances in order to identify regions with small local color variations e.g. regions of uniform colors or of small color gradients. The first three steps use our human vision model to build a representation of the image. Watercolor is computed as:

1) Resizing to 200 dpi (Section IV-A)

2) Domain transform filtering (Section IV-D)

3) Conversion to the Lab color space (Section IV-B)

4) Color distance map computation: The goal of the color distance map is to represent the color distances/gradients between the image pixels. Gradient maps and other similar metrics do not provide precise information with each neighbor of a given pixel. Using a graph representation would involve costly processing, so we use a matrix representation of color distances. This implies the use of a four-connected pixel neighborhood since we cannot represent the two diagonal distances with one diagonal pixel.

Let be m and n the dimensions of the input image I . The color distance map M , preserving the gradient directions, of size $2m \times 2n$ is computed as follows. We first compute the color distances between horizontal neighbor pixels (7), then between vertical neighbor pixels (8), for the pixels (9) and for the diagonals (10) as:

$$M(2(i-1)+1, 2j) = d(I(i, j), I(i, j+1)) \quad (7)$$

$$M(2i, 2(j-1)+1) = d(I(i, j), I(i+1, j)) \quad (8)$$

$$M(2(i-1)+1, 2(j-1)+1) = \min_4(2(i-1)+1, 2(j-1)+1, M) \quad (9)$$

$$M(2i, 2j) = \max_4(2i, 2j, M) \quad (10)$$

where $\max_4(i, j, M)$ and $\min_4(i, j, M)$ return respectively the maximal and minimal value of M in the four-neighborhood around the coordinates (i, j) and d is the Euclidean Lab color distance.

5) Denoising: Some colorimetric noise may remain creating local minima, which in turn create superfluous regions. The goal of the denoising is to remove these minima. All values below a distance threshold $t_d = \frac{\sigma(M)}{c_2} + c_1$ are set to 0, where $\sigma(M)$ is the standard deviation of the distance

map M . The constant $c_1 = 0.2$ is equivalent to 10 % of the just noticeable color difference and the constant $c_2 = 7.5$ is estimated empirically.

The threshold t_d makes the algorithm adaptable to both the image content and the colorimetric noise. It may be argued that c_2 is dependent on the image content and thus a parameter. However, we make the hypothesis that the image content dependency is contained in $\sigma(M)$ and that the factor c_2 reflects how much the human visual system can adapt to this content variation and more precisely to the color variations. The idea behind this is that the human eye and the way we analyze an image adapts to how much color variations are present in the image, measured by $\sigma(M)$. Increasing the image resolution will span the same color difference over a larger number of pixels hence the values of c_1 and c_2 are resolution dependent.

6) Watershed: It produces a region index map N of the same size as the color distance map. This region index map puts in correspondence each value of the color distance map with the index of the region to which it belongs.

7) Image space region extraction: It converts the region index map N of the color distance map into one for the input image O of size $m \times n$ as:

$$O(i, j) = N(2(i - 1) + 1, 2(j - 1) + 1) \quad (11)$$

This is a straightforward sampling of the index region map on the locations of the pixels in the color distance map M . The regions produced for the input image do not have any border between them, thus removing the ambiguity related to border pixels.

B. Post-processing (spatio-colorimetric sensitivity)

Watercolor oversegments the CCCs because it is not capable of handling the high color variations that occur on a small scale and are imperceptible for a human observer. To solve this issue we use our spatio-colorimetric distance ϕ (6) to merge these regions as described in Algorithm 1.

The algorithm merges neighboring regions with a spatio-colorimetric distance below the color threshold t . The color of a region is computed for each channel as the median of the channel values of the pixels of the region (line 4). The median reduces sensitivity to noise.

The merging process uses two steps to be independent from the processing order of the regions. Firstly (lines 9–12), it lists the pairs of regions to merge together *without* merging them. Secondly (lines 13–14), it merges the regions found by the first step. This independence is important as it makes the algorithm more stable. It will produce similar results even if the image is rotated or reversed. The merging is performed by increasing region size with the loops on lines 7 and 9.

Finally, some regions that are produced are color gradients. In this case, the color of the region can differ significantly from the color of some pixels of the region and it would be unsafe to merge this region based on its color. The list *canbmerged* identifies such regions. Its computation occurs at the same time as that of the region colors (lines 2–6).

Algorithm 1 Post-processing algorithm for watercolor

Input: the list of region indexes *regidxlist* and the input color image I .

Parameters: color threshold t .

Output: the list of region indexes *regidxlist*

```

1: create empty list of colors regcolors
2: create list of merging rights canbmerged = false
3: for  $i = 0; i \leq \text{regidxlist.size}(); i++$  do
4:    $\text{regcolors}(i) = \text{median}(I(\text{regidxlist}(i)))$ 
5:   if  $\phi(\text{regcolors}(i), I(\text{regidxlist}(i))) \leq 4t$  then
6:      $\text{canbmerged}(i) = \text{true}$ 
7: for  $i = 1; i \leq \text{maximum region size}; i++$  do
8:   create null merging list merge
9:   for all region  $j$  with a size  $i$  do
10:    find neighbor region of region  $j$  with
11:      $\text{canbmerged}(i) == \text{true}$  and closest color  $n$ 
12:     if  $\phi(\text{regcolors}(j), \text{regcolors}(n)) \leq t$  then
13:        $\text{merge}(j) = n$ 
14:     merge all regions  $j$  with  $\text{merge}(j)$  if it is not null
15:   update regidxlist and regcolors
16: return regidxlist

```

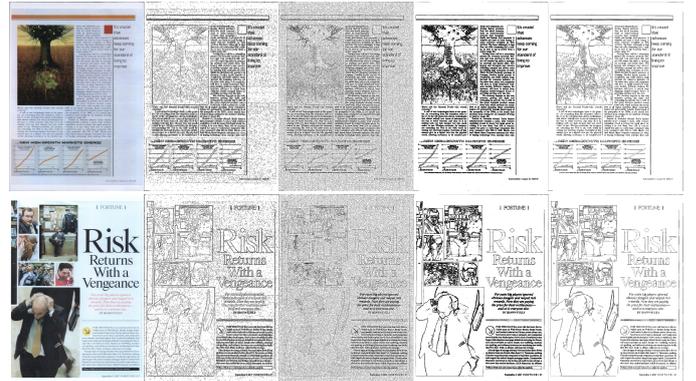


Fig. 1: Segmentation results, from left to right: original image, FH, QS, WC and WCP. The images produced by QS look gray because of the many region boundaries.

VI. COMPARISON WITH THE STATE OF THE ART

Considering the state of the art, we benchmark our algorithm (watercolor (WC) and watercolor with post-processing (WCP)) against the Quick-Shift (QS) algorithm [12] and the algorithm of Felzenszwalb and Huttenlocher (FH) [11]. These algorithms are not constraint by a parameter fixing the number of superpixels, as explained in Section II, thus not biasing our evaluation. We use two benchmarks: the L3iDocCopies segmentation benchmark [2] to evaluate the stability of our algorithm and the Berkeley segmentation benchmark 500 (BSDS500) [30] to evaluate its segmentation accuracy, to show that our algorithm is versatile and can also perform well on natural scene images.

A. L3iDocCopies benchmark

L3iDocCopies is a document segmentation evaluation benchmark focused on the stability of the algorithms e.g. their ability to have similar results on several copies of the same document containing print and scan noise. It uses 990 images of 18 copies of 55 documents.

It uses the false positive rate, the normalized standard deviation of the number of regions S_R and the mean number of regions \bar{n}_R for evaluating the stability [2]. The false positive rate relies on a layout descriptor that is not capable of handling more than 100 regions, so we will not use it. The two other metrics evaluate the variation of the number of regions produced by the algorithm over several copies of a given document.

B. BSDS500 benchmark

BSDS500 is a natural scene image segmentation benchmark. It is composed of 500 images out of which 200 are for testing purposes. This benchmark has been adapted to the evaluation of superpixel algorithms with several performance indicators: boundary precision (BP), boundary recall (BR) and F-measure (BF), undersegmentation error (UE) and achievable segmentation accuracy (ASA). BP, BR and BF evaluate how close the segmentation boundaries are to the ground truth. Since the algorithms may not have an absolute spatial accuracy, there is a distance tolerance to consider that a boundary pixel in the ground truth matches one in the result. We used the default distance tolerance equal to 0.75 % of the image diagonal, which corresponds to approximately 4.3 pixels in this data set. UE evaluates the bleeding of the superpixels (or CCCs) across edges and ASA provides an upper bound on the accuracy that could be reached by a segmentation algorithm using the proposed partition of the image.

C. Results

We present now the results of all algorithms on the L3iDocCopies and BSDS500 benchmarks.

a) *L3iDocCopies*: No evaluation had been done for QS and FH on document images, hence we had to devise a proper set of parameters for them. For FH, for document images scanned at 300 dpi, we used the standard deviation for the Gaussian blur $\sigma = 0.8$ (as proposed in [11]), the scale parameter $k = 100$, and the minimal region size equal to 10 pixels to keep the details of the characters. k and the region size units are in pixels, hence we scaled them for the documents at 600 dpi. QS required more memory than the one available on our computer for the document images at 600 dpi, so they were rescaled to 300 dpi. We kept the ratio between the color and geometric distances $\tau = 0.75$ and the maximal distance of 7 pixels. However, the proposed value of the scale parameter, 3, was too large to keep the character details, hence it was set to 2.

Table I summarizes the results. We obtained the lowest S_R for WC (WCP is not far), as well as the lowest \bar{n}_R for WCP. Hence, our algorithm clearly outperforms the state of the art and is three to four times more stable. While the good

Perf. Ind.	FH	QS	WC	WCP
S_R	0.41	0.50	0.11	0.15
\bar{n}_R	38 170	155 246	142 077	11 303

TABLE I: Performance of the algorithms on L3iDocCopies.

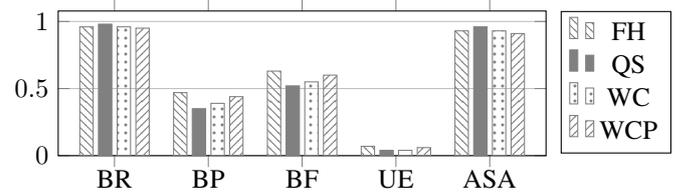


Fig. 2: Performance of the algorithms on BSDS500.

normalized standard deviation of the number of regions S_R for WC could be attributed to the large number of regions, which would mechanically produce a small value for S_R , this performance is maintained for WCP, which produces four times less regions than FH and QS. Fig. 1 shows the results of the algorithms on two typical images of the data set. FH and QS have difficulties producing relevant regions due to their inability to deal with scale variations, while WC and WCP produce only one region for the background.

b) *BSDS500*: For this benchmark we used the parameters for QS and FH that seemed best according to [6]. Our algorithm needs only the dpi of the image to process. Since we had no way of estimating it, we chose to process the images at their full resolution and did not resize them.

The performance results of the algorithms are presented in Fig. 2. All algorithms have very good values for BR, UE and ASA. BP and BF are worse as by design the algorithms produce too many regions and too many boundaries. While not being the best, our algorithm has a better BP than QS, which means that it produces less superfluous regions. It also



Fig. 3: Segmentation results, from top to bottom: original image, ground truth, results of FH, QS, WC and WCP.

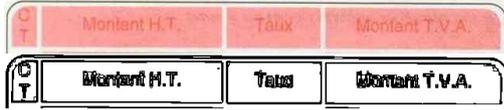


Fig. 4: WC segmentation of red text on a red background.

has a better UE than FH. The main difference lies in the visual appearance of the segmentation results (Fig. 3). Our algorithm produces cleaner segmentation results than QS and FH especially in uniform regions.

Hence, on this benchmark the proposed algorithm matches the state of the art without any parameter tuning and produces cleaner segmentation results. Depending on the need, one may use WC for a fast processing or WCP for cleaner results.

D. Further analysis

Our algorithm performs successfully on images with textured background (with one CCC per texture element as expected), on gray level images, on text with any shape, direction and size and on images with overlapping components. For instance, Fig. 4 presents its ability to segment red text on a red background. However, it has issues with high compression JPEG noise on textured areas as it creates superfluous regions in the noisy areas.

VII. CONCLUSION

In this paper we proposed a parameter-free connected color component segmentation algorithm named watercolor. This algorithm extends the definition of connected components to color images and is based on a new model of human vision, which includes several characteristics previously neglected by the computer vision community.

We compared our algorithm with superpixel segmentation algorithms. It outperforms them by a factor three to four in terms of stability on document images and matches their performance on natural scene images. Moreover, our algorithm is content agnostic and thus versatile, and can adapt to challenging situations such as color gradients, textured or colored backgrounds.

Anyhow, the algorithm is not flawless. It has issues with high compression JPEG noise on textures, which could be alleviated with specific JPEG denoising algorithms. The post-processing might be improved i.e. the estimation of the constants S_L , S_a and S_b of the spatio-colorimetric distance, or by integrating region noise statistics in the region merging.

ACKNOWLEDGMENT

This work is financed by the French National Research Agency (ANR) project SHADES referenced under ANR-14-CE28-0022 and by the Town community of La Rochelle.

REFERENCES

[1] S. Eskenazi, P. Gomez-Krämer, and J.-M. Ogier, "When document security brings new challenges to document analysis," in *International Workshop on Computational Forensics (IWCF)*, 2015, pp. 104–116.
 [2] —, "Evaluation of the stability of four document segmentation algorithms," in *Document Analysis Systems (DAS)*, 2016, pp. 1–6.

[3] —, "Let 's be done with thresholds !" in *International Conference on Document Analysis and Recognition (ICDAR)*, 2015, pp. 851–855.
 [4] H. Zhu, F. Meng, J. Cai, and S. Lu, "Beyond pixels : a comprehensive survey from bottom-up to semantic image segmentation and cosegmentation," *J Vis Commun Image R*, vol. 34, no. 1, pp. 12–27, 2016.
 [5] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE PAMI*, vol. 34, no. 11, pp. 2274–2282, nov 2012.
 [6] D. Stutz, "Superpixel segmentation using depth information," RWTH Aachen University, Aachen, Tech. Rep., 2014.
 [7] P. Neubert and P. Protzel, "Compact Watershed and preemptive SLIC: on improving trade-offs of superpixel segmentation algorithms," in *International Conference on Pattern Recognition (ICPR)*, 2014, pp. 996–1001.
 [8] A. Schick, M. Fischer, and R. Stiefelhagen, "Measuring and evaluating the compactness of superpixels," in *International Conference on Pattern Recognition (ICPR)*, 2012, pp. 930–934.
 [9] X. Pan, Y. Zhou, C. Zhang, and Q. Liu, "Flooding based superpixel generation with color, compactness and smoothness constraints," in *International Conference on Image Processing (ICIP)*, 2014, pp. 4432–4436.
 [10] D. Comaniciu and P. Meer, "Mean Shift: a robust approach toward feature space analysis," *IEEE PAMI*, vol. 24, no. 5, pp. 603–619, 2002.
 [11] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, pp. 167–181, 2004.
 [12] A. Vedaldi and S. Soatto, "Quick Shift and kernel methods for mode seeking," in *European Conference on Computer Vision (ECCV)*. Springer, 2008, pp. 705–718.
 [13] M. D. Fairchild, *Color appearance models*, 3rd ed. John Wiley and Sons, 2013.
 [14] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE PAMI*, vol. 20, no. 11, pp. 1254–1259, 1998.
 [15] D. Miller, P. Schor, and P. Magnante, "Optics of the normal eye," in *Ophthalmology*, 3rd ed. Elsevier, 2009, ch. 2.7, pp. 52–60.
 [16] G. Sharma, "Color fundamentals for digital imaging," in *Digital Color Imaging Handbook*. CRC Press, 2002, ch. 1, p. 816.
 [17] J. M. M. Linhares, P. D. Pinto, and S. M. C. Nascimento, "The number of discernible colors in natural scenes," *J Opt Soc Am A*, vol. 25, no. 12, pp. 2918–2924, 2008.
 [18] CIE, "CIE 208:2014 effect of stimulus size on colour appearance," *Color Research & Application*, vol. 39, no. 5, p. 518, 2014.
 [19] M. Withoutouk, K. A. G. Smet, and P. Hanselaer, "Brightness prediction of different sized unrelated self-luminous stimuli," *Optics Express*, vol. 23, no. 10, pp. 13455–13466, 2015.
 [20] K. Xiao, M. R. Luo, C. Li, G. Cui, and D. Park, "Investigation of colour size effect for colour appearance assessment," *Color Research & Application*, vol. 36, no. 3, pp. 201–209, 2011.
 [21] P. G. J. Barten, "Physical model for the contrast sensitivity of the human eye," in *Human Vision, Visual Processing, and Digital Display*, 1992, pp. 57–72.
 [22] M. Nagao and T. Matsuyama, "Edge preserving smoothing," *Computer Graphics and Image Processing*, vol. 9, no. 4, pp. 394–407, 1979.
 [23] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE PAMI*, vol. 12, no. 7, pp. 629–639, 1990.
 [24] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *International Conference on Computer Vision (ICCV)*. IEEE, 1998, pp. 839–846.
 [25] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation," *ACM TOG*, vol. 27, no. 3, p. 1, 2008.
 [26] R. Fattal, "Edge-avoiding wavelets and their applications," *ACM TOG*, vol. 28, no. 3, p. 1, 2009.
 [27] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE PAMI*, vol. 35, no. 6, pp. 1397–1409, 2013.
 [28] K. N. Chaudhury, D. Sage, and M. Unser, "Fast O(1) bilateral filtering using trigonometric range kernels," *IEEE TIP*, vol. 20, no. 12, pp. 3376–3382, 2011.
 [29] E. S. L. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM TOG*, vol. 30, no. 4, p. 1, 2011.
 [30] P. Arbelaez, M. Maire, C. C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE PAMI*, vol. 33, no. 5, pp. 898–916, 2011.