Document detection in videos captured by smartphones using a saliency-based method

Minh Ôn Vũ Ngọc, Jonathan Fabrizio, Thierry Géraud
EPITA Research and Developement Laboratory (LRDE)
Le Kremlin-Bicetre, France
Email: firstname.lastname@lrde.epita.fr

***Abstract*—Smartphones are now widely used to digitize paper documents. Document detection is the first important step of the digitization process. Whereas many methods extract lines from contours as candidates for the document boundary, we present in this paper a region-based approach. A key feature of our method is that it relies on visual saliency, using a recent distance existing in mathematical morphology. We show that the performance of our method is competitive with state-of-the-art methods on the ICDAR Smartdoc 2015 Competition dataset.**

***Keywords*-Document detection, Visual saliency, Mathematical morphology, Smartphone-based acquisition, Dahu pseudo-distance, Image segmentation.**

I. Introduction

In nowadays world, the demand for using digital document is increasing because of its convenience in searching, storing, retrieving, etc. A traditional way to digitize paper is using a scanner machine, which is heavy, costly, and usually not portable. With the development of smartphone cameras, many people use them to acquire documents. Digitizing papers in images or videos captured by smartphones is not the same procedure as scanning: images captured by smartphones do contain a background. Therefore, the first step of the digitization process is the extraction of the document region from the scene. In this paper, our goal is to segment automatically documents in an acceptable run time.

Images captured using smartphones lead to many issues during the digitization procedure. The scene contexts are unknown, the lighting conditions are variable, and the illumination is not homogeneous. Images can be noisy. Furthermore, the camera is handheld, this can lead to out-of-focus or motion blur.

To detect documents in images, the most usual approach is to detect lines from the contours of the document as candidates for segmentation [1]. This strategy is very popular as we can see in the survey [2] on camera-based analysis of documents and in numerous algorithms submitted to the SmartDoc competition [3]. Here we explore a new approach using a method based on visual saliency. Over the past decades, visual saliency detection methods have been widely used as efficient tools for objects detection and recognition. They rely on the computation of a saliency map that highlights the salient objects. Recently, a new distance has been introduced, the Minimum Barrier Distance (MBD) [4], which has been proved to be effective for saliency detection. An advantage of the MBD is its robustness to noise and blur. That is why it has been extensively used in the salient object detection field [5–8]. Unfortunately, this distance is very difficult to compute, then a new pseudo-distance has been derived from it: the Dahu pseudo-distance [9], which can be computed efficiently and quickly (we will abusively call it the Dahu distance in the sequel for the sake of simplicity).

In this paper, we combine a salient object detection approach that relies on the Dahu distance with a hierarchical image simplification and segmentation to localize documents in images and videos.

The main contributions of this paper are:

- A segmentation algorithm based on the Dahu distance.
- A scheme for document detection by combining visual saliency with image segmentation. The tracking method is used as well to follow the segmented document in the video.
- A study and a comparison of our method with state-of-the-art, proving that our method is fast, competitive and can deal with most situations.
- An application of the Dahu distance.

The paper is organized as follows. In Sec. II, we recall some of the main state-of-the-art methods relative to document detection. Sec. III gives a brief recall of the theoretical tools used in the proposed method.Sec. IV presents our method. In Sec. V, we evaluate our method and compare it with the state-of-the-art. The conclusions and perspectives are discussed in Sec. VI.

II. Related works

Document detection in images captured by smartphones is an important topic. That is why the challenge 1 of the ICDAR 2015 Smartdoc competition [3] focuses on the evaluation of document detection/segmentation algorithms. Eight submissions were made; these methods can be classified into two categories depending on the used strategy: the most common strategy is to rely on line detection; the other is a hierarchical tree-based representation of the image. Seven methods among the proposed ones extract lines in the image as candidates for document segmentation. The Canny edge detector [10], Hough transform [11], and LSD algorithm [12] are used to detect lines in images. Although it is the most common strategy, these methods cannot work well if the document is curled.

Among them, two methods outperform the others. The ISPL-CVML one uses the LSD algorithm to get vertical and horizontal segments on the down-sampled image, then

color and edge features are exploited to select document boundaries. The SmartEngines method [13] uses several algorithms to detect segments in the image, then builds a graph of these segments. A quadrangle of a possible document is constructed from this graph while considering the weights and angles of edges. The final quadrangle is obtained by applying a Kalman filter based on some local descriptors.

The hierarchical tree-based representation method [14] of the LRDE gets the highest score. They compute the energy of each node of the tree, which consists of two terms measuring how the shape fits the quadrilateral form and how "noisy" the object is (text lines and figures, etc.) and then select the best candidate. Nevertheless, this method is slow.

Besides these methods, the Smart IDReader [1] method combines series of algorithms depending on the class of documents. A Viola-Jones method is applied as a decision tree of strong classifiers for document detection [15].

Geodesic Object proposal [16] method starts with using six seeds to cover all of the objects in the image. The signed geodesic distance transform computed from each seed which is specified with an image region, is then evaluated for being the best candidate document.

In [17], an approach is proposed to detect identity documents by using a saliency-based method. The Dahu distance is used to compute the saliency map, then a simple thresholding is applied to segment the identity document. However, this last procedure is not sufficient to extract whole documents, and also the method has difficulty in choosing the best threshold for all images in the dataset.

Recently, a CNN-based method [18] has been proposed; it considers that locating a document is equivalent to looking for four corners in the document. The AlexNet architecture is first used to predict the four corners of the document, then a shallow convolutional neural network is used to refine the prediction. However, it does not handle correctly occlusions or side effects.

III. Background

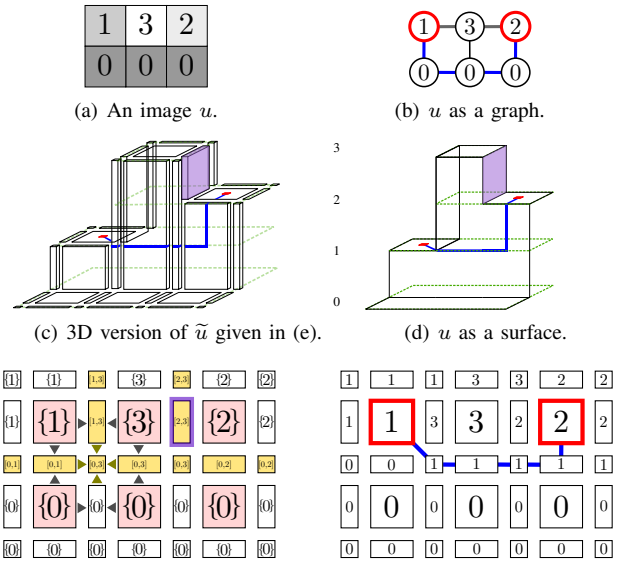
This section gives a brief introduction recalling the theoretical tools involved in our method.

A. Tree-based Image representation

An image can be represented in a hierarchical way. Some hierarchical representations are based on threshold decompositions: the connected components, obtained by thresholding a gray-level image, are related thanks to the inclusion relationship. The simplest ones are the min-/max-trees [19] (based on upper/lower thresholds sets). A more natural one, the tree of shapes [20, 21], is a fusion of these two trees; its nodes represent then shapes, that is, the filled-in connected components of the upper/lower threshold sets of the image.

B. Hierarchy of image segmentations

Image segmentation decomposes an image into several meaningful sets of pixels (called regions) sharing common



(a) An image u. (b) u as a graph.

(c) 3D version of $\widetilde{u}$ given in (e). (d) u as a surface.

(e) Interval-valued image $\widetilde{u}$. (f) A minimal path in a $\overline{u} \leqslant \widetilde{u}$.

Figure 1. Image representations for computing barrier distances [9].

features (color, intensity, texture, etc). Hierarchical segmentations provide segmentations at different levels in which the segmentation at the coarser level is composed of regions from segmentation at finer detail levels (i.e., regions at upper-level nodes are merged from their children nodes at lower levels). The segmented tree structure has been applied to object detection [22], image simplification and segmentation [14, 23].

C. The Minimum Barrier Distance (MBD)

The MBD has been defined in [4, 24, 25], in which a gray-level image (Fig. 1(a)) is considered as a vertex-valued graph (Fig. 1(b)). Let $\pi = \langle ..., \pi_i, ... \rangle$ denote the path of pixels on the graph. The MBD between x and x' in u is:

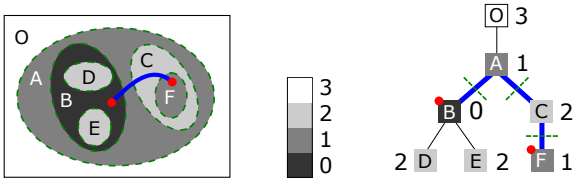
$$d_u^{\text{MB}}(x, x') = \min_{\pi \in \Pi(x, x')} \left(\max_{\pi_i \in \pi} u(\pi_i) - \min_{\pi_i \in \pi} u(\pi_i) \right), \quad (1)$$

where $\Pi(x, x')$ denotes the family of all paths that connect two points x and x'. The MBD is thus the minimum value of the barrier strength along a path between two points.

An example of the MBD is illustrated in Fig. 1(b): the minimal path between two red points x, x' is depicted in blue and corresponds to the sequence of values $\langle 1, 0, 0, 0, 2 \rangle$; we obtain then $d_u^{\text{MB}}(x, x') = 2$.

D. The Dahu distance

The continuous version of the MBD, called the Dahu distance, is defined in [9] and considers an image (Fig. 1(a)) as a surface (Fig. 1(d)). However, a scalar function is not well-suited to describe its elevation. In [9], 2D cubical complexes are used to describe this surface. A 2D cubical complex is a set of elements: 2D, 1D and 0D elements, where 2D elements correspond to the original pixels, 1D and 0D elements are inter-pixels, which take the interval value from its adjacency 2D elements. The inter-pixel is a transition step between two pixels, which is a way to get a discrete topology. It is illustrated in Fig. 1(c), where the



(a) Image u. (b) Gray scale. (c) Tree $\mathfrak{S}(u)$.

Figure 2. The tree of shapes of an image allows to easily express and compute the Dahu distance and saliency maps [9].

purple part corresponds to 1D element with the purple border in Fig. 1(e).

The scalar image $\overline{u}$ in Fig. 1(f) is included in the interval-valued image $\widetilde{u}$ depicted in Fig. 1(e). Their inclusion relationship is denoted by $\Subset$. The Dahu distance is defined as:

$$d_u^{\text{DAHU}}(x,\, x') \;=\; \min_{\overline{u} \Subset \widetilde{u}} \; d_{\overline{u}}^{\text{MB}}(h_x,\, h_{x'}) \tag{2}$$

$$\;=\; \min_{\overline{u} \Subset \widetilde{u}} \; \min_{\pi \in \Pi(h_x,\, h_{x'})} \tau_{\overline{u}}(\pi), \tag{3}$$

where h_x is the 2D element which corresponds to x. The Dahu distance is then in some way the extension of the MBD considering all the possible scalar images $\overline{u}$, which are included in the interval-valued image $\widetilde{u}$. The optimal path, depicted in blue, between the two red points in the interval-valued image gives a distance of 1 between x and x'.

E. The Dahu distance computed on the tree of shape (ToS)

At a glance, we can see that the Dahu distance is hard to compute: we have to look for the minimal path in all the possible scalar images $\overline{u}$. However, this new distance can be efficiently computed thanks to the tree of shapes (ToS) [20, 26] (Fig. 2). Intuitively, a minimal path between the two red points x and x' in the image space in Fig. 2(a) is equivalent to the path between the two nodes t_x and $t_{x'}$, which contain respectively the two points x and x', see Fig. 2(c). These two paths cross the same set of level lines (illustrated as dashed lines in both the image space and the tree space). The Dahu distance is then defined as the MBD between two nodes on the tree $\mathfrak{S}(u)$:

$$\begin{aligned} d_u^{\text{DAHU}}(x,\, x') \;&=\; d_{\mathfrak{S}(u)}^{\text{MB}}(t_x,\, t_{x'}) \\ &=\; \max_{t \,\in\, \dot{\pi}(t_x,\, t_{x'})} \mu_u(t) \;-\; \min_{t \,\in\, \dot{\pi}(t_x,\, t_{x'})} \mu_u(t), \end{aligned} \tag{4}$$

where $\mu_u(t)$ denotes the gray-level assigned with the node t of $\mathfrak{S}(u)$, and $\dot{\pi}(t_x,\, t_{x'})$ is a path in the tree between t_x and $t_{x'}$.

IV. The Proposed Method

We detect documents thanks to saliency: the brightest pixels in the saliency map are considered as candidates for document detection. Beside the saliency map, a method for image segmentation, which exploits the Dahu distance and the histogram of the color of pixels from the super-pixels is applied. Then a max-tree of the final visual saliency map, which is a combination of the saliency map and the image segmentation is constructed. The document features are computed at each node of the tree. The idea is to consider the local maxima of the energy map as candidates for document detection. To enhance the document detection during a video stream, a simple tracking method compares the positions of the shapes in consecutive frames. The whole process is illustrated in Fig. 3.

A. Saliency based on the Dahu distance

We assume that we have a high contrast between the document and the background, and the border of the image is mostly background. Thus, we consider pixels along the border of the image as seed nodes to compute the visual saliency map [27]. The corresponding set of nodes $T_{X'}$ with a set of points X', on the tree $\mathfrak{S}(u)$ is:

$$T_{X'} \;=\; \{\, t_{x'} \,;\, x' \in X' \,\} \;\subseteq\; \mathfrak{S}(u). \tag{5}$$

A saliency map S_u^{DAHU} of an image u based on the Dahu distance from a set of pixels X' can be computed by:

$$S_u^{\text{DAHU}}(x, X') \;=\; \min_{x' \in X'} d_u^{\text{DAHU}}(x,\, x') \;=\; S_{\mathfrak{S}(u)}^{\text{MBD}}(t_x, T_{X'}). \tag{6}$$

The value of each point x in S_u^{DAHU} corresponds to the value of $S_{\mathfrak{S}(u)}^{\text{MBD}}$ at node t_x, which can be computed by a propagation method using a priority queue. In fact, the saliency map $S_u^{\text{DAHU}}(x, X')$ is computed instantly on the ToS $\mathfrak{S}(u)$, whatever the set X'. Note that the ToS can be computed in quasi-linear time w.r.t. the number of pixels [21, 28] in the image, and can be parallelized [29].

B. Image simplification and segmentation

Different from the method developed at LRDE [14], which looks for a document among hundreds of thousands of nodes in the ToS of the original image, we propose an image simplification and segmentation method based on the Dahu distance to reduce the number of image elements into tens of nodes for max-tree construction (Sec. IV-C). The process starts with the SLIC algorithm [30] to partition an image into several small regions called super-pixels.

Let $G = (V, E)$ denote a graph where V denotes super-pixels and $E \subseteq V \times V$ denotes the edges joining these super-pixels. Each edge $e_{ij} = (v_i, v_j) \in E$ is assigned a weight that measures the dissimilarity between the two super-pixels v_i and v_j. A minimum spanning tree (MST), which is built thanks to Kruskal's algorithm [31], is used to simplify the graph of super-pixels. Two super-pixels which have a similar appearance tend to be connected in the MST. On the contrary, edges with larger weights tend to be removed. The distance $D(R_i, R_j)$ between two connected super-pixels R_i and R_j is used as an edge weight on the MST and is defined as:

$$D(R_i, R_j) = \alpha \times d_u^{\text{DAHU}} + \beta \times d_c, \tag{7}$$

where d_c is a measure of the difference between the color histogram of two neighbors; d_u^{DAHU} is the Dahu distance between the center marker C_i and C_j (3×3 pixels) of two neighboring super-pixels R_i and R_j:

$$d_u^{\text{DAHU}}(C_i, C_j) \;=\; \min_{x_i \in C_i} \min_{x_j \in C_j} d_u^{\text{DAHU}}(x_i,\, x_j). \tag{8}$$

The Dahu distance between two center markers C_i and C_j is the minimum of the Dahu distance between all of the pixels x_i and x_j inside these markers.

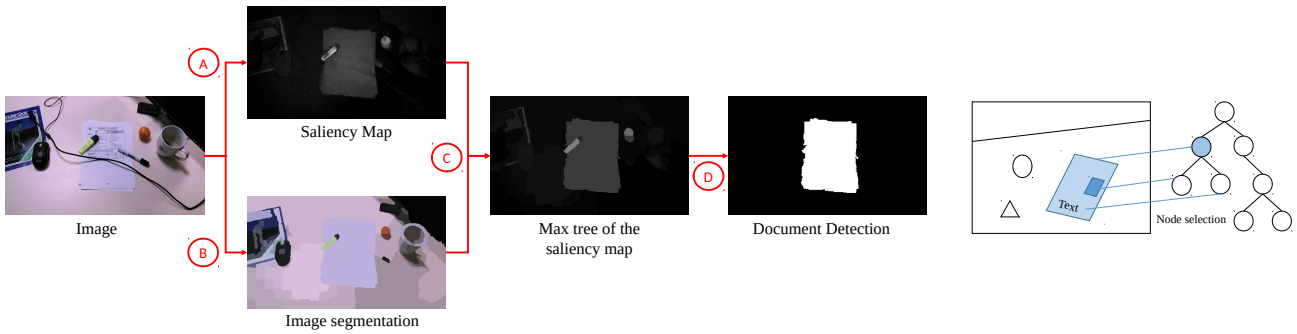


Figure 3. On the left: A scheme for document detection. A visual saliency map is computed with considering that mostly boundary pixels are the background. In parallel, an algorithm for image segmentation is adopted by using the Dahu distance and histogram of colors of pixels on superpixels. A max-tree of the visual saliency map is constructed. Then a candidate document is segmented from the max-tree. On the right: Document detection from the max-tree. A candidate document tends to have a quadrilateral shape, also the top line is parallel with the bottom line (respectively with the left line and the right line). On another hand, the document region is brighter in the saliency map.

Also, the Chi-square distance d_c between two color histograms H_i and H_j (computed on the quantized colors of all pixels in the regions R_i and R_j) is defined by:

$$d_c(R_i, R_j) = \exp(-\frac{1}{2}\sum_{k=1}^{m}\frac{[H_i(k) - H_j(k)]^2}{H_i(k) + H_j(k)}) \quad (9)$$

When the MST is computed, we segment the tree. There exist several works of hierarchical image segmentation based on energy minimization [32, 23]. Here, we adopt the Mumford-Shah functional proposed in [33]. A general energy functional has the following form: $E_{\lambda_s} = \lambda_s E_{re} + E_{fi}$, where E_{re} is a regularization term, E_{fi} is a data fidelity term, and λ_s is a parameter able to control the simplification or segmentation degree of the algorithm. The higher value of λ_s is, the coarser the segmentation degree is. The data fidelity term E_{fi} is computed from the scalar luminance with $l = (r+g+b)/3$. It is actually the variance of the luminance of each node on the MST. The regularization term E_{re} is equal to the contour length $|\partial R|$ corresponding to the node R. The total energy of R has the following expression:

$$E(R) = \sum_{x \in R} \left\| l(x) - \bar{l}(R) \right\|^2 + \lambda_s(|\partial R|) \quad (10)$$

With a fixed value λ_s, the optimal cut is chosen from additive laws of composition. The energy on the parent node R is compared with the sum of the energy of all the children nodes T_i^R. The parent node is kept if it satisfies this condition : $E(R) \leq \sum E(T_i^R)$.

C. Max tree of a visual saliency map

In this step, we combine the saliency map with the image segmentation method, which is mentioned in the previous section. The saliency value of each region R_i is the average of the saliency map of every pixel in the region.

$$S_u^{\text{DAHU}}(R_i) = \frac{\sum_{x \in R_i} S_u^{\text{DAHU}}(x)}{|R_i|} \quad (11)$$

After this combination, we get the final saliency map. As illustrated in Fig. 3, the candidate document is brighter than the background. Thus, we construct a max-tree representation directly on the graph of regions. The next step is to find the candidate document, which is considered as a local maximum in the max-tree (seen as a graph).

D. Document detection

Assume that the candidate document is represented in the max-tree, then the document segmentation problem is to find the document in the tree space. To do that, we assign an attribute to each region that corresponds to a node on the max-tree. Here, we borrow one prior knowledge from the document information that is the document has a quadrilateral shape. We compute sequentially the attribute on every node of the tree and we observe how much these attributes fit with the document criteria. Our criteria are the followings:

1. A ratio that measures how much a shape boundary of a node A is close to the best fitting quadrilateral $Quad(A)$:

$$E_f(A) = \frac{|A \cap Quad(A)|}{|A \cup Quad(A)|} \quad (12)$$

2. The angles between the top (resp. the bottom) lines, denoted by TL (resp. BL), and between the left (resp. the right) lines, denoted by LL (resp. RL):

$$E_a(A) = \frac{\cos(TL, BL) + \cos(LL, RL)}{2} \quad (13)$$

3. The saliency map value of each node of the tree:

$$E_s(A) = S_u^{\text{DAHU}}(A) \quad (14)$$

The final attribute is computed by this equation:

$$E(A) = E_f(A) \times E_a(A) \times E_s(A) \quad (15)$$

Once the attribute $E(A)$ is available, we can look for the "most likely" node on the tree maximizing this attribute function. Fig. 3 shows the node selection procedure.

E. Tracking a document between frames

To implement document detection in video streams, a tracking method is used to compare the document position between the previous frame and the current frame. Based on the node attributes computed in the previous section, we select the best three nodes in the tree as candidate documents, and then we look for this document position in the previous frame. The current detected shape A_t^* is the one that minimizes the distance to the shape A_{t-1}^* in the previous frame.

$$A_t^* = A_t^k : k = \min\{i | 1 \leq i \leq 3 : d(A_{t-1}^*, A_t^i)\} \quad (16)$$

where $d(X, Y)$ is the Jaccard index.

V. Experimental Results

We evaluate in this section the use of the Dahu distance on document detection.

A. Dataset and Evaluation

To perform the evaluation, we use the ICDAR 2015 SmartDoc challenge 1 dataset [3]. These videos are taken by a Google Nexus 7 tablet for a total of 25K frames with a resolution of 1920×1080 on six types of document, that are placed over 5 different backgrounds. The document pages are placed inside the image (and never hit the boundary of the image). The dataset is challenging (variable lighting condition, inhomogeneous background, motion blur and out-of-focus blur). Especially, the fifth background is complex with many objects placed near the document or even over it.

To evaluate the performance of the method, the Jaccard index between the detected document A and the ground truth G is used:

$$JI = area(G \cap A)/area(G \cup A) \tag{17}$$

B. Experiments and Results

We start with reducing the size of each frame by a factor of 2. We also convert an image to $L^*a^*b^*$ space to mimic the human vision. Then the ToS is built on L^* and b^* channels of each frame (the contrast between the document and the background is not sufficient on the other channel).

The SLIC algorithm [30] is adopted to segment an image into 300 super-pixels. The values $\alpha = 5$ and $\beta = 1$ are chosen to emphasize the Dahu distance. Variations on them do not change results so far. The value $\lambda_s = 8000$ is low enough to avoid under-segmentation of the document.

Quantitative results on the Smartdoc 2015 dataset are shown in Fig. 4. Our method achieves the second highest overall score over 12 methods. The difference with the first ranked method (LRDE) is negligible (0.972 vs 0.97), but we are about 16 times faster (1 min vs 3.7s). Our method is better than the other methods in the competition (even with SmartEngines method [13] which is ranked first on background 1, 2 and 3). Especially, it fails on the most difficult case: background 5 (shortly Bg. 5). In this evaluation, we do not compare our method with SEECS-NUST-2 [18] method because of the following reasons: they use highly correlated training and testing data. For background 5, they used 50% of each video for training, next 20% for validation, and only 30% for testing. It is not a good strategy because:

- the training and testing dataset are too much similar (the accuracy on Bg. 5 decreased from 0.94 to 0.66 when all samples extracted from Bg. 5 "testing video" were removed from training [18]),
- the testing dataset is different from the other methods.

In Fig. 5, we show the results of our method on some challenging images. Our method is well handled with blurred, illumination variation cases. Even in some tedious cases such as the superposition of documents, non-straight boundaries document, partially occluded documents or the document slightly hits the boundary of the image, our method succeeds.

Method	Bg 1	Bg 2	Bg 3	Bg 4	Bg 5	Overall	Runtime
A2iA-1	0.972	0.801	0.912	0.635	0.189	0.779	?
A2iA-2	0.960	0.806	0.912	0.826	0.189	0.809	?
ISPL-CVML	0.987	0.965	0.985	0.977	0.856	0.966	?
LRDE [14]	0.987	0.978	0.989	0.984	0.861	0.972	**1min**
NetEase	0.962	0.955	0.962	0.951	0.222	0.882	?
SEECS-NUST	0.888	0.826	0.783	0.781	0.011	0.739	?
RPPDI-UPE	0.827	0.910	0.970	0.365	0.216	0.741	?
SmartEngines [13]	0.989	0.983	0.990	0.979	0.688	0.955	?
L. R. S. Leal [16]	0.961	0.944	0.965	0.930	0.412	0.895	0.43s
LRDE-2 [34]	0.905	0.936	0.859	0.903	?	?	0.04s
Ours	0.985	0.982	0.987	0.980	0.848	0.97	**3.7s**
Smartdoc ave. [3]	0.9465	0.9031	0.9377	0.8122	0.4041	0.8552	?

Figure 4. Quantitative results on Smartdoc 2015 competitions data. The red (resp. blue) color denotes the best (resp. second) result in each background. Our method gets the second highest overall score. It is competitive with the LRDE method [14], but about 20 times faster than their method.

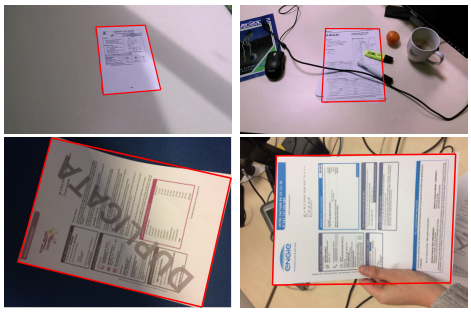


Figure 5. Some qualitative results of our method. These images show the robustness of our method to illumination, blur and curled document.

Concerning the tests, we used an Intel i7 2.6 GHz CPU with 8 GB of RAM. The speed can be improved as we use a naïve implementation of the method. The total time (excluding I/O time) of our method depends on the size of the image and the number of super-pixels. Fig. 6 demonstrates the compromise between the executed time of the process and the overall score. If we increase the scaling parameter and decrease the number of super-pixels, the executed time is much shortened, while the accuracy remains acceptable. Our method achieves an overall score of 0.962 at run time equal to 0.65 second, which is almost 100 times faster than the method of the LRDE [14].

VI. Conclusions and Perspectives

We have proposed a new method for document detection in videos captured by smartphones, with very few *a priori* knowledge on the documents and the images. We have shown the efficiency of the visual saliency for document detection. We have also presented a new method for image partitioning which relies on the Dahu distance. Our scheme:

- is very fast,
- offers a good compromise between speed and accuracy,
- works in most situations.

This article is also the opportunity for us to illustrate an application of the Dahu distance. The presented scheme is very fast and our next step is to make it work on smartphones.

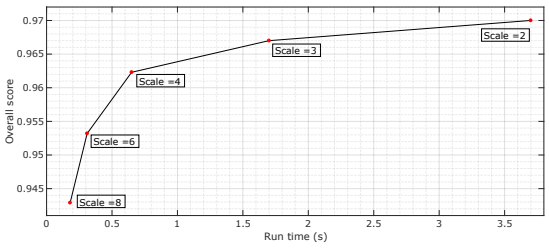


Figure 6. The compromise between the executed time (image resolution i.e. image scale) and the overall accuracy. Even at low resolution, our method achieves an overall score of 0.962 for a run time equal to 0.65s.

ACKNOWLEDGMENT

This work has been conducted in the context of the MOBIDEM project, part of the "Systematic Paris-Region" and "Images & Network" Clusters (France). This project is partially funded by the French Government and its economic development agencies.

REFERENCES

[1] K. Bulatov, V. V. Arlazarov, T. Chernov, O. Slavin, and D. Nikolaev, "Smart idreader: Document recognition in video stream," in *Proc. of ICDAR*, vol. 6. IEEE, 2017, pp. 39–44.

[2] J. Liang, D. Doermann, and H. Li, "Camera-based analysis of text and documents: A survey," *Inter. Journal on Document Analysis and Recognition*, vol. 7, no. 2, pp. 84–104, 2005.

[3] J.-C. Burie *et al.*, "ICDAR 2015 competition on smartphone document capture and OCR (SmartDoc)," in *Proc. of ICDAR*, 2015, pp. 1161–1165.

[4] R. Strand *et al.*, "The minimum barrier distance," *CVIU*, vol. 117, no. 4, pp. 429–437, 2013.

[5] R. Strand, K. C. Ciesielski, F. Malmberg, and P. K. Saha, "The minimum barrier distance," *Computer Vision and Image Understanding*, vol. 117, no. 4, pp. 429–437, 2013.

[6] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Minimum barrier salient object detection at 80 fps," in *Proc. of ICCV*, 2015, pp. 1404–1412.

[7] W.-C. Tu, S. He, Q. Yang, and S.-Y. Chien, "Real-time salient object detection with a minimum spanning tree," in *Proc. of CVPR*, 2016, pp. 2334–2342.

[8] X. Huang and Y. Zhang, "Water flow driven salient object detection at 180 fps," *Patt. Rec.*, vol. 76, pp. 95–107, 2018.

[9] T. Géraud, Y. Xu, E. Carlinet, and N. Boutry, "Introducing the Dahu pseudo-distance," in *Proc. of ISMM*, ser. LNCS, vol. 10225, 2017, pp. 55–67.

[10] L. Ding and A. Goshtasby, "On the canny edge detector," *Pattern Recognition*, vol. 34, no. 3, pp. 721–725, 2001.

[11] R. Dida, "Use of the hough transformation to detect lines and curves in pictures," *Magazine Communications of the ACM*, vol. 15, no. 1, 1972.

[12] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "Lsd: A fast line segment detector with a false detection control," *IEEE Trans. on Patt. Ana. and Mach. Intel.*, vol. 32, no. 4, pp. 722–732, 2010.

[13] A. Zhukovsky *et al.*, "Segments graph-based approach for document capture in a smartphone video stream," in *Proc. of ICDAR*, vol. 1. IEEE, 2017, pp. 337–342.

[14] Y. Xu, E. Carlinet, T. Géraud, and L. Najman, "Hierarchical segmentation using tree-based shape spaces," *IEEE Trans. on Patt. Ana. and Mach. Intel.*, vol. 39, no. 3, pp. 457–469, 2017.

[15] A. Minkina, D. Nikolaev, S. Usilin, and V. Kozyrev, "Generalization of the viola-jones method as a decision tree of strong classifiers for real-time object recognition in video stream," in *Inter. Conf. on Ma. Vi.*, vol. 9445. ISOP, 2015, p. 944517.

[16] L. R. Leal and B. L. Bezerra, "Smartphone camera document detection via geodesic object proposals," in *Computational Intelligence (LA-CCI)*. IEEE, 2016, pp. 1–6.

[17] M. Ô. V. Ngoc, J. Fabrizio, and T. Géraud, "Saliency-based detection of identy documents captured by smartphones," in *IAPR International Workshop on DAS*, 2018, pp. 387–392.

[18] K. Javed and F. Shafait, "Real-time document localization in natural images by recursive application of a cnn," in *Proc. of ICDAR*, vol. 1. IEEE, 2017, pp. 105–110.

[19] P. Salembier, A. Oliveras, and L. Garrido, "Antiextensive connected operators for image and sequence processing," *IEEE TIP*, vol. 7, no. 4, pp. 555–570, 1998.

[20] V. Caselles and P. Monasse, *Geometric Description of Images as Topographic Maps*, ser. LNM. Springer, 2009, vol. 1984.

[21] T. Géraud, E. Carlinet, S. Crozet, and L. Najman, "A quasi-linear algorithm to compute the tree of shapes of n-D images." in *Proc. of ISMM*, ser. LNCS, vol. 7883, 2013, pp. 98–110.

[22] Y. Xu, T. Géraud, and L. Najman, "Context-based energy estimator: Application to object segmentation on the tree of shapes," in *Proc. of IEEE ICIP*, 2012, pp. 1577–1580.

[23] Y. Xu, T. Géraud, and L. Najman, "Hierarchical image simplification and segmentation based on Mumford-Shah-salient level line selection," *Pattern Recognition Letters*, vol. 83, no. 3, pp. 278–286, 2016.

[24] K. C. Ciesielski *et al.*, "Efficient algorithm for finding the exact minimum barrier distance," *Computer Vision and Image Understanding*, vol. 123, pp. 53–64, 2014.

[25] R. Strand *et al.*, "The minimum barrier distance: A summary of recent advances," in *Proc. of DGCI*, ser. LNCS, vol. 10502. Springer, 2017, pp. 57–68.

[26] P. Monasse and F. Guichard, "Fast computation of a contrast-invariant image representation," *IEEE Transactions on Image Processing*, vol. 9, no. 5, pp. 860–872, 2000.

[27] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," *Proc. of ECCV*, pp. 29–42, 2012.

[28] E. Carlinet, S. Crozet, and T. Géraud, "The tree of shapes turned into a max-tree: A simple and efficient linear algorithm," in *Proc. of IEEE ICIP*, 2018, pp. 1488–1492.

[29] S. Crozet and T. Géraud, "A first parallel algorithm to compute the morphological tree of shapes of nD images," in *Proc. of IEEE ICIP*, 2014, pp. 2933–2937.

[30] R. Achanta *et al.*, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. on Patt. Ana. and Mach. Intel.*, vol. 34, no. 11, pp. 2274–2282, 2012.

[31] J. B. Kruskal, "On the shortest spanning subtree of a graph and the traveling salesman problem," *Proc. of the American Mathematical society*, vol. 7, no. 1, pp. 48–50, 1956.

[32] B. R. Kiran and J. Serra, "Global–local optimizations by hierarchical cuts and climbing energies," *Pattern Recognition*, vol. 47, no. 1, pp. 12–24, 2014.

[33] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on pure and applied mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[34] É. Puybareau and T. Géraud, "Real-time document detection in smartphone videos," in *Proc. of IEEE ICIP*, 2018, pp. 1498–1502.