

Background Removal of French University Diplomas

Tanmoy Mondal¹, Quoc Bao Dang², Mickaël Coustaty², Petra Gomez-Krämer², and Jean-Marc Ogier²

¹ Zenith, INRIA, Montpellier, France

² L3i, University of La-Rochelle, France

`tanmoy.mondal@inria.fr, {quoc.bao.dang, petra.gomez, mickael.coustaty, jean-marc.ogier}@univ-lr.fr`

Abstract. Separation of foreground text from noisy or textured background is an important preprocessing step for many document image processing problems. In this work we focus on decorated background removal and the extraction of textual components from French university diploma. As far as we know, this is the very first attempt to resolve this kind of problem on French university diploma images. Hence, we make our dataset public for further research, related to French university diplomas. Although, this problem is similar to a classical document binarization problem but we have experimentally observed that classical and recent state of the art binarization techniques fails due to the different complexity of our dataset. In French diplomas, the text is superimposed on decorated background and there is only a small difference of intensities between the character borders and the overlapped background. So, we propose an approach for the separation of textual and non-textual components, based on Fuzzy-C-Means clustering. After obtaining clustered pixels, a local window based thresholding approach and the Savoula binarization technique is used to correctly classify pixels, into the category of text pixels. Experimental results show convincing accuracy and robustness of our method.

Keywords: Text-Graphics Separation · French University Diplomas · Document Binarization · Fuzzy C-Means Clustering.

1 Introduction

Separation of text foreground from page background is an important processing step for the analysis of various kind of documents e.g. historical, administrative, scanned and camera captured documents. This separation enables various document analysis algorithms (such as character recognition, localization of stamps, logos, signatures etc.) to focus only on the written text without the disturbing background noise. If texts and graphics are located in separate regions in the page, localizing the region of texts and graphics is easier in comparison with texts superimposed on background graphics [1].

In this paper, we are focusing on the proper extraction of all the text components from French university diploma, which contains a complex background. The objective is to secure university diploma and to reduce the possibilities of fraud. The core idea is to encode a unique and secret number in the diploma while printing it. This encoding is done by changing the shape of certain characters and each modified character shape is associated with a specific digit (this association is only known to some specific people of the university). Hence, later to check the authenticity of any scanned student's diploma, we need proper and undeformed extraction of text characters which is required for decoding the secret and unique number, associated with each diploma. All French university diplomas are printed on the same decorated and authenticate thick paper/parchment (in French called as : *parchemin*), only and strictly fabricated by the National Printing House (in French called as: *Imprimerie Nationale*). So, any French university diploma has the same background and only the foreground text changes for different universities/institutes (moreover scanned diploma will diversify due to scanning effects). Moreover, the text fonts, size and text styles also varies for different universities/institutes. In every diplomas, each individual student's information e.g. name, date-of-birth, place-of-birth etc. changes also. All these variation together, makes the binarization process harder. This complex background is highly superimposed with the foreground textual components which makes it complex to separate the textual components from the background (see Fig. 2a for an example of diploma). This problem can also be seen as separation of graphical background from the textual foreground. On the other hand, this problem has a high resemblance with historical degraded document binarization. Historical degradation includes non-uniform background, stains, faded ink, ink bleeding through the page and un-even illumination [2]. The state of the art text-graphics separation approaches do not perform well in our case because unlike general text-graphics separation data e.g. maps, floor plans etc. [1], the texts in diploma images are fully superimposed upon the decorated colored background.

There exist several categories of binarization techniques, which are often evaluated on the dataset of popular document image binarization contest (DIBCO) [11]. But these categories of binarization approaches have several constraints. Many common binarization approaches are based on the calculation of local or global thresholds and image statistics, which doesn't work for our case (see experimental section in 4). So as a contribution of this paper, we propose a new algorithm for the separation of text components from graphical background. The algorithm starts by structured forest based fast gradient image formation followed by textual region selection by using image masking for the initial filtering of textual components. Then Fuzzy-C-Means clustering is applied to separate textual and non-textual components. As the clustering may result in some deformation of the textual components (characters), hence we also propose character reconstruction technique by using a local window based thresholding approach and Savoula binarization algorithm [12]. This technique helps to correctly classify some extra pixels into the category of text pixels, which helps to reconstructs/recover the missing character pixels. As far as we know, this is the very initial

proposed attempt of removing background to obtain texts from French university diplomas.

The remainder of this paper is organized as follows. First, Section II summarizes the work related to text/graphics segmentation in general. Then, Section III provides an overview of the method proposed. Subsequently, experimental results are described and analyzed in Section IV. Finally, Section V concludes the paper and gives an overview of future work.

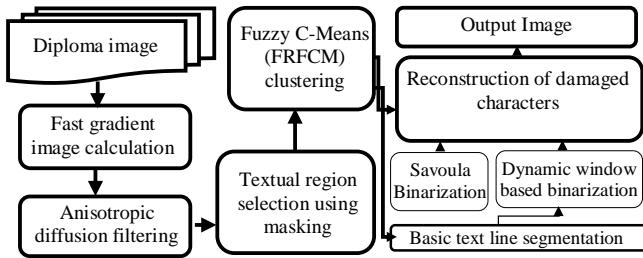


Fig. 1: The proposed algorithm architecture. The portions in bold are the main steps of the algorithm while the two non-bold blocks are the auxiliary steps needed for the “reconstruction of damaged character” block

2 Related Work

By definition, text binarization means the labeling of each pixel in the image as text or background which has a good resemblance with our problem. Due to the unavailability of any previous work, we have outlined here some of the related work from the domain of document image (mainly historical) binarization. The existing binarization techniques in the literature can be categorized into two principal categories: learning-free and learning-based approach. Several binarization techniques have been proposed in the past decade but very recently the trend in document binarization tends to machine learning based methods, mostly relying on deep-learning based techniques.

Most deep learning based image binarization algorithms use convolutional neural network (CNN), e.g. by Tensmeyer and Martinez [13], or variations thereof, such as the deep supervised network (DSN) approach proposed by Vo et al. [14]. Due to the requirement of sufficient amount of training data and specially in our case, where no ground truth exists, the training based approaches are not useful. Moreover, recently the learning-free approaches [8] have shown high potential and comparable accuracy with respect to learning-based approaches. In [2], a local binarization method based on a thresholding with dynamic and flexible windows is proposed. Jia et al. [8] proposed an approach based on structural

symmetry of pixels (SSP) from text strokes for binarization. Using the combination of SSP and Fuzzy C-Means (FCM) clustering (the FRFCM technique [7]), Mondal et al. [10] also proposed an improved and fast binarization technique for historical document images.

While Deep Learning methods is getting more and more attention in the community, FCM [7] based models are among the popularly used methods for image segmentation. We avoided to use any learning based methods (e.g. deep neural networks) for background removal due to the non availability of pixel's level ground truth (GT). It is also highly cumbersome and expensive to generate such GT for our high resolution experimental data set. These methods rely on the Fuzzy set theory which introduce fuzziness for the belongingness of each image pixels in certain class. Such clustering technique, named as (FRFCM) [7], is used as one of the main processing step for our proposed algorithm. So, here we also provide a brief background and state of the art methods on FCM techniques to justify our choice. It is superior to hard clustering as it has more tolerance to ambiguity and retains better original image information. As FCM only considers gray-level information without considering the spatial information, it fails to segment images with complex texture and background or images corrupted by noise. So, the FCM algorithm with spatial constraint (FCM_S) [7] was proposed, which incorporates spatial information in the objective function. However, FCM_S is time consuming because the spatial neighbor term is computed in each iteration. To reduce the execution time, two modified versions, named as FCM_S1 and FCM_S2 [7] were proposed. These algorithms employ average filtering and median filtering to obtain the spatial neighborhood information in advance. However, both FCM_S1 and FCM_S2 are not robust to Gaussian noise, as well as to a known noise intensity.

The Enhanced FCM (EnFCM)[7] algorithm is an excellent technique from the viewpoint of low computational time as it performs clustering based on gray level histograms instead of pixels of summed image. However, the segmentation results by EnFCM is only comparable to that produced by FCM_S. To improve the segmentation results, the Fast Generalized FCM (FGFCM) [7] was proposed. It introduces a new factor as a local similarity measure, which guarantees both noise immunity, detailed preservation of image segmentation. Along with that it also removes the requirement of empirical parameter α in EnFCM and performs clustering on gray level histograms. But FGFCM needs more parameter settings than EnFCM. A robust fuzzy local information C-Means clustering algorithm (FLICM) is introduced in [7], which is free from parameter selection. This algorithm replaces the parameter α of EnFCM by incorporating a novel fuzzy factor into the objective function to guarantee noise immunity and image detail preservation. Although FLICM overcomes the image segmentation performance but the fixed spatial distance is not robust to different local information of images. Hence, we use a significantly fast and robust algorithm based on morphological reconstruction and membership filtering (FRFCM) [7] for image segmentation.

The aforementioned state of the art binarization methods are developed for the binarization of historical document images, by testing some of them, we

observe unsatisfying results. In the following section, we propose a new method for separating the textual foreground from textured background.

3 Proposed Method

In this section, the proposed algorithm is explained and it's overall architecture is shown in Fig. 1. At first, the structural random forest based gradient image formation technique is applied on the image $\mathcal{I}(x, y)$ (see Fig. 2a).

3.1 Structural Forest Based Fast Gradient Image Formation

A real time gradient computation for the objective of edge detection is proposed by Dollár et al. [4], which is faster and more robust to texture presence than current state of the art approaches; e.g. sobel [4]. The gradient computation (see Fig. 2b) is performed based on the present structure in local image patches and by learning both an accurate and computationally fast edge detector.

3.2 Anisotropic Diffusion Filtering

An anisotropic diffusion [6] filter is applied on the gradient image (see Section 3.1) to reduce image noises (appears due to scanning) without removing significant contents of the image; e.g. edges, lines and other details (see Fig. 2c).

3.3 Selection of Textual Regions from Original Image

After filtering the image, edges are detected by the Canny edge detector (see Fig. 2d). Only better gradient images are obtained by Dollár et al. [4] (refer to Section 3.1) but to generate the binary edges, we have applied Canny edge detection algorithm. The high and low threshold of Canny are computed automatically as ϕ (equal to Otsu's threshold value) and $0.5 \times \phi$. The idea was not to use hand-crafted thresholds.

The edge image is then dilated by using a 7×7 ³ rectangular kernel to connect the broken edges, caused by the Canny algorithm (see Fig. 2e). It can be observed that it contains holes and gaps inside the character shapes (see Fig. 2e). These holes/gaps are filled by applying the well known flood fill⁴ based hole filling approach (see Fig. 2f). Let us denote this hole filled image as $\mathcal{H}(x, y)$. Now original gray scale image pixels from the gray scale image corresponding to this hole filled image are simply obtained by creating a blank image of the same size as the original image and initialized by 255 (let's say $\mathcal{T}(x, y)$; see Fig. 2g)). This process corresponds to applying a mask⁵ using image $\mathcal{H}(x, y)$. Now a clustering

³ Considering the high image resolution, we choose 7×7 kernel size. However, this parameter should be adapted to fit other dataset requirements.

⁴ https://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous_transformations.html?highlight=floodfill

⁵ background is white (255) and foreground is black(0)

technique is applied to the masked image $\mathcal{T}(x, y)$ for classifying the text pixels from the background pixels.

$$\mathcal{T}(x, y) = \begin{cases} \mathcal{I}(x, y); & \text{if } \mathcal{H}(x, y) = 0. \\ 255; & \text{otherwise.} \end{cases} \quad (1)$$

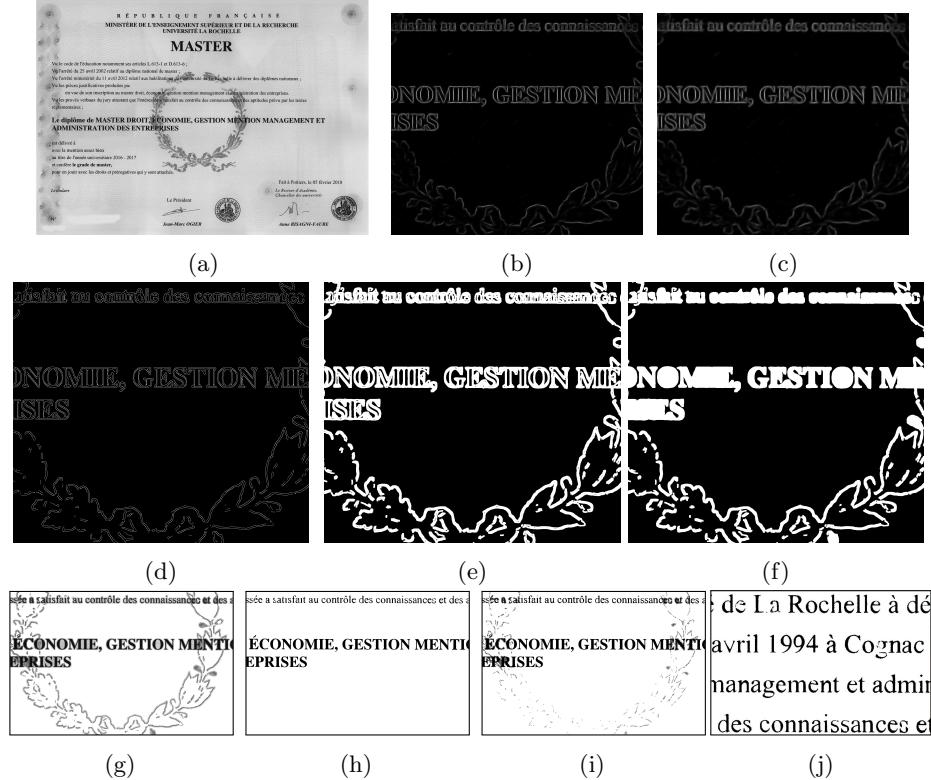


Fig. 2: Various preprocessing steps: (a) Original gray image (b) Gradient image obtained by [4] (c) Anisotropic filtered image (d) Edge detected by canny (e) Dilated images after canny edge detection (f) Hole filled image after dilation (g) Selected textual regions $\mathcal{T}(x, y)$ (h) Only sure text pixels after clustering (i) Sure and confused pixels after clustering (j) Existing issue of character deformation after clustering (sure text pixels only).

3.4 Fuzzy C-Means (FRFCM) Clustering Algorithm

Following to the aforementioned state of the art on FCM clustering mentioned in Section 2, Lei and Jia et al.[7] proposed FRFCM algorithm with a low computational cost which can achieve good segmentation results for various types

Method Name	Time Required	Method Name	Time Required
FCM_S1	232.15	FGFCM_S2	2.23
FCM_S2	201.14	ENFSCM	0.74
FCM_M1	3.07	FLICM	662.25
FGFCM	2.15	FRFCM	0.27
FGFCM_S1	1.79		

Table 1: Time required (in seconds) by the different clustering algorithm on a cropped diploma image.

of images and can also achieve high segmentation precision. It employs morphological reconstruction (MR) to smooth the images in order to improve noise immunity and image detail preservation. Results obtained by various aforementioned clustering techniques on a small cropped image of 822×674 are shown in Table 1. It can be seen that FRFCM has an impressive computational time.

Therefore, FRFCM is faster than other aforementioned improved FCM algorithms. By setting the number of clusters to 3 (sure text pixels, confused text pixels and background pixels), we apply FRFCM clustering on $\mathcal{T}(x, y)$ (i.e. on the image shown in Fig. 2g), which gives a clustered image (see Fig. 2i). The two deeper intensities are visible in Fig. 2i except white, which represent the background. Among these two deeper intensities, the darker one represents the intensities of *sure text pixels* (the image formed by only these pixels are shown in Fig. 2h) and the other ones represent *confused pixels*. The following mentioned techniques are applied to qualify these pixels as texts or background.

Some character shapes are deformed due to clustering (see Fig. 2j) but this problem mainly comes from the gradient image formation followed by the Canny edge detection. Due to the low contrast between the background and foreground at these specific regions of the image, the gradient image fails to clearly signify text regions followed by the failure of Canny's edge detection. This is shown in Fig. 3a (top left: zoomed portion of dilated image after canny, top right: hole filled image, bottom left: textual region separation bottom, right: clustered image). Note that even before performing the dilation operation on the Canny image, some portions of the image were already missing. The following technique is applied to recover the missing portions and to reconstruct the image.

3.5 Savoula Binarization

We apply Savoula binarization [12] on the original image at this step of the algorithm (see Fig. 3b). The following thresholding formula is used:

$$\mathcal{S} = m(1 - k(1 - \frac{\sigma}{R})) \quad (2)$$

where k is equal to 0.2 and $R = 125$ is a gray-level range value and m , σ represents the mean and standard deviation of image window respectively.

3.6 Basic and Fast Text Line Segmentation

Text lines are roughly segmented based on a horizontal projection of the binarized image, obtained by the FRFCM clustering (the sure text pixels). After obtaining the heights of each text line, a simple but efficient approach to calculate the average of these values are described in Algorithm 1. This technique⁶ is helpful for removing outliers from the set of values which helps to calculate a better average value. The values are stored in $Arr[items]$ and the intelligent average is obtained in the variable $AvgVal$.⁷

Algorithm 1: INTELLIGENT AVERAGING

Input: $Arr[items]$

Output: $AvgVal$

- 1 Divide the elements in 10 equal spaced bins (we use *hist* function of Matlab)
 - 2 From the bin centers of each bins, get the lower and upper limit of each bin
 - 3 Sort the bins based on number of elements exists in each bin
 - 4 Only consider top two bins
 - 5 Choose those values which belongs to these two top bins (using upper and lower limits of each bin). Let's call them refined values (denoted as $Refined[items]$)
 - 6 Calculate mean (M) and standard deviation (S) from these refined values
 - 7 Now again obtain the re-refined values by verifying if it belongs to :

$$(M - S) \leq Refined[items] \leq (M + S)$$
 - 8 Finally calculate mean from these re-refined values
-

3.7 Window Based Image Binarization

We apply a fast window based binarization technique on the original gray image. This technique is inspired by the method in [2]. From the previous step of text line segmentation, we get the very first (*textStart*) and the last image row (*textEnd*), which contains text. As we already have the average height of text lines ($AvgVal$) so by using this information, we divide the image of considerable region (region between *textStart* and *textEnd*) into equal size horizontal stripes. Now each horizontal stripes is divided into 20 equal parts⁸ and each of

⁶ Please note that, may be for our case of diploma images, as there are not many text lines, this approach may not show high difference in results, compared to classical averaging but when there are more data, this technique shows better performance.

⁷ We use *hist* function of Matlab toolbox to calculate histogram of values, divided into 10 bins.

⁸ Considering the resolution of image, we choose this value as 20. The objective is to get small textual region for local binarization. We have not properly tested this chosen value so the performance can be slightly improved by the proper choice of this value by evaluating on some test dataset.

these parts called ("window") on which the binarization is performed by using the following Equation 3 and 4.

$$\sigma_{adaptive} = \left[\frac{\sigma_W - \sigma_{min}}{\sigma_{max} - \sigma_{min}} \right] \times maxIntensity \quad (3)$$

$$\mathcal{T}_W = \mathcal{M}_W - \frac{\mathcal{M}_W \times \sigma_W}{(\mathcal{M}_W + \sigma_W)(\sigma_{adaptive} + \sigma_W)} \quad (4)$$

where $\sigma_{adaptive}$ is the gray level value of the adaptive standard deviation of the window. σ_W is the standard deviation of the window, σ_{min} and σ_{max} are the minimum and maximum standard deviation of all the windows (20 here), $maxIntensity$ is the maximum intensity value of the horizontal stripe, \mathcal{M}_W is the mean value of the pixel intensities of the window and \mathcal{T}_W is the threshold of the window. The obtained binarized image is denoted by $WinImg^{binary}(x, y)$ (see Fig. 3c). It can be seen that although the simple Savoula binarization has comparatively performed well but it fails to remove the centered decorated background (called "couronne" in French) whereas the binarized image from window Based Image Binarization is prominently keeping most of the background decorated portions. So, none of these can be directly used for our case.

3.8 Reconstruction of Deformed Characters

We propose a technique for the reconstruction of deformed characters with the help of Savoula and Window based binary image. This reconstruction is required as the shape of the characters should remain intact for proper decoding of the secret number, as mentioned in introduction (see section 1). The obvious text pixels (sure text pixels), obtained from the fuzzy clustering are copied into a new image (called $\mathcal{K}(x, y)$). Now, for each confused pixel (line 7 in Algorithm 2) in the fuzzy clustered image, we check first if the same pixel is a foreground pixel in Savoula image (Υ) and also in window based binary image (χ). If they are foreground pixels (line 8 in Algorithm 2) then we traverse at the neighborhood of this pixel with the window size of $winSz \times winSz$ and count the number of such pixels which are foreground both in $\Upsilon(x, y)$ and $\chi(x, y)$. If this number exceeds 20% of the window size ($winSz \times winSz$) (line 17 in Algorithm 2), then we mark this pixel as text pixel otherwise it is marked as background pixel. The value of $winSz$ is taken as ($stroke width / 2$). The *stroke width* is calculated by using the approach, mentioned in [3]. The recovered and background separated image is shown in Fig. 3d and one the zoomed portion is shown in Fig. 3e).

4 Results and Discussion

In this section, we will mention and discuss the obtained results on dataset of university diplomas.

Algorithm 2: CHARACTER RECONSTRUCTION

Input: $\gamma, \beta, MRows, NCols$
Output: \mathcal{H}_{smooth}

```

1 maskSz  $\leftarrow 5$                                  $\triangleright$  mask size for reconstruction
2 winSz  $= maskSz/2$ 
3 combi_Img  $\leftarrow \gamma$ 
4 sum  $\leftarrow 0$ 
5 for  $iR \leftarrow 0$  to  $(M - 1)$  do
6   for  $jC \leftarrow 0$  to  $(N - 1)$  do
7     /* Verifying the confused pixels in clustered image */ 
8     if  $(\beta(iR, jC) \neq 0) \& (\beta(iR, jC) \neq 255)$  then
9       /* check if this pixel is text in both in savoula and */
10      /* windowed binary image */ 
11      if  $(\Upsilon(iR, jC) = 0) \& (\chi(iR, jC) = 0)$  then
12         $sCnt \leftarrow 0; x1 \leftarrow 0; y1 \leftarrow 0$ 
13        for  $k \leftarrow -winSz$  to  $winSz$  do
14          for  $c \leftarrow -winSz$  to  $winSz$  do
15            /* counting the number of strong pixels at the */
16            /* surrounding */ 
17            if  $(\Upsilon(y1, x1) == 0) \& (\chi(y1, x1) == 0)$  then
18              |  $sCnt ++$ 
19            if  $(sCnt \geq ((0.2 \times winSz^2) - 1))$  then
20              |  $combi_Img(iR, jC) = 0$ 
21              |  $\beta(iR, jC) = 0$ 
22
23 ** $\gamma$  (sure text pixels),  $\beta$  (sure and confused text pixels),  $MRows$  (Total
number of image rows),  $NCols$  (Total number of image columns).

```

4.1 Dataset

To evaluate the proposed algorithm, we have created the French university diploma dataset described in the following.

4.1.1 Diploma Data set : Total 40 images were scanned in colored and gray level (mentioned as Dip_Col and Dip_Gray in Table. 2) by "Fujitsu" scanner in 300 (20 images) and 600 (20 images) DPI (of size is 7000×5000 pixels approx.).

4.1.1.1 Ground Truth Preparation : Due to no prior availability of dataset, we have created a dataset and corresponding GT. Due to high image size, it is cumbersome to create the GT manually. So, we adapted a semi manual approach by manually selecting the range of the gray level threshold for foreground / text pixels⁹. Then the obtained images are manually corrected by GIMP software.¹⁰

⁹ Data and available at : https://github.com/tanmayGIT/University_Diploma_BackGd_Removal.git

¹⁰ These documents are real *University of La Rochelle, France* diplomas of students. We avoid to generate fake diplomas as it is forbidden by the law. For the privacy

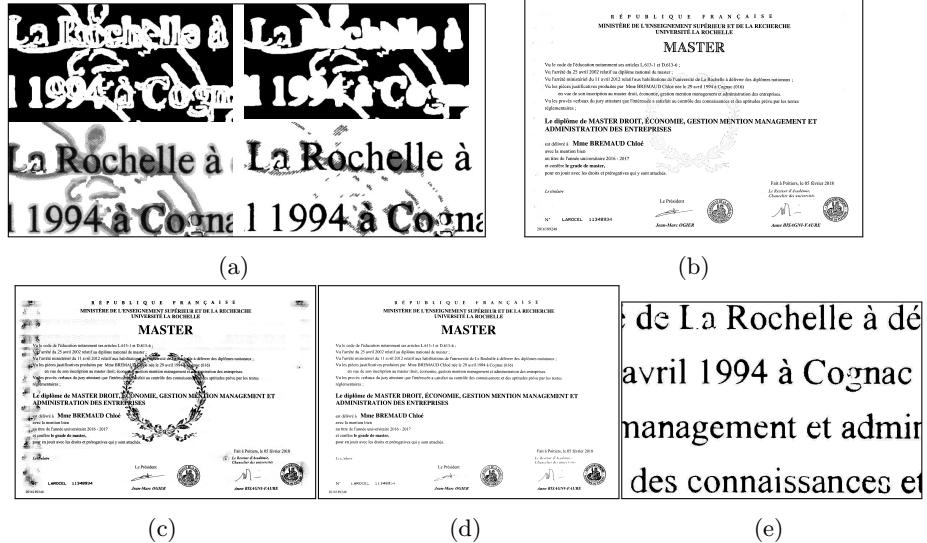


Fig. 3: (a): Reasons of character deformation (b) Binary image by Savoula technique (c) Binary image by Dynamic window based technique (d) The text separated image after reconstruction (e) Zoomed portion after image reconstruction

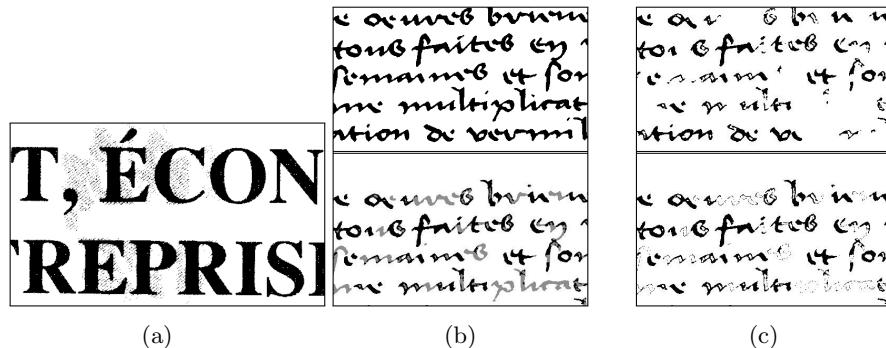


Fig. 4: (a): The issues with prepared GT (b) *Top* : One example of GT image from DIBCO dataset; *Bottom* : Clustered image (sure and confused text pixels) (c) *Top* : Only sure text pixels from clustered image *Bottom* : After the image reconstruction

purpose, the confidential information e.g. name, date of birth, unique number of diploma etc. are manually removed from the gray image (for color images, it was first converted into gray)

Table 2: Result on University Diploma Dataset. The best results of each metric are highlighted in bold.

Dataset	No. of Images	Method Name	F-Measure (%) (↑)	PSNR (↑)	DRD (↓)
300_Col	10	Proposed	89.89	20.90	5.56
		Niblack	10.26	2.03	501.80
		Savoula	87.70	19.98	7.90
		Wolf-Jolin	83.90	18.60	11.22
		Bataineh [2]	68.88	14.80	30.24
		Gatos [5]	84.74	18.80	10.75
		Mondal [10]	84.40	19.05	9.39
		Jia [9] (for 10 images)	71.88	15.48	25.84
300_Gray	10	Proposed	93.62	22.90	2.99
		Niblack	10.61	1.85	486.97
		Savoula	91.98	21.91	4.70
		Wolf-Jolin	89.40	20.50	6.56
		Gatos [5]	91.08	21.23	5.55
		Bataineh [2]	74.59	15.78	23.54
		Mondal [10]	86.88	19.89	7.19
		Jia [9] (for 8 images)	87.91	19.93	8.07
600_Col	10	Proposed	92.05	21.91	3.59
		Niblack	10.83	2.09	446.60
		Savoula	89.36	20.58	5.77
		Wolf-Jolin	85.88	19.16	8.40
		Bataineh [2]	68.96	14.71	27.59
		Gatos [5]	85.36	18.91	9.07
		Mondal [10]	88.48	20.48	5.86
		Jia [9] (for 8 images)	85.12	19.26	8.32
600_Gray	10	Proposed	95.14	24.04	1.76
		Niblack	10.96	1.90	420.37
		Savoula	92.12	21.89	4.03
		Wolf-Jolin	89.63	20.50	5.59
		Bataineh [2]	74.92	15.75	20.44
		Gatos [5]	91.30	21.25	4.57
		Mondal [10]	89.33	20.76	4.94
		Jia [9] (for 6 images)	83.60	18.70	11.05

* For the comparison purpose, only training free best performing methods are chosen from literature.

4.2 Evaluation Protocol

We have used *F-Measure (FM)*, *Peak Signal to Noise Ratio (PSNR)* and Distance Reciprocal Distortion Metric (DRD) as the evaluation metrics (same as

DIBCO competition [11]). It can be seen from Table 2, that the performance of our technique is quite promising and achieves an average accuracy (F-Measure) of 93% on four parts of the complete data set. The accuracies could be further improved by a better preparation of the GT (mainly by proper removal of background pixels because our algorithm correctly classifies the background pixels). It can be seen from Fig. 4a that the GTs are not properly cleaned/corrected, which affected the statistical result. The proposed technique performed better than the classical (e.g. Niblack, Savoula and Wolf-Jolin’s [11]) binarization techniques. However, among these three classical binarization techniques, the Savoula binarization performed comparatively better than others. But as mentioned before that these classical binarization techniques are mainly unable to remove the centered decorated background (i.e. “couronne”) (see Savoula binarization result in Fig. 3b and 5a). We also compared the proposed approach

Method Name	Time Required	Method Name	Time Required
Niblack [11]	3.03 (C++)	Savoula [11]	3.06 (C++)
WolfJolin [11]	3.06 (C++)	Bataineh [2]	1.03 (C++)
Gatos [5]	3.21 (C++)	Mondal* [10]	421 + 33.43 + 1031 = 1485.43 (Matlab and C++)
Jia [9]	128 (.exe provided by authors)	Proposed**	35.8 + 14 + 5.8 = 55.6 (Matlab and C++)

*Mondal et.al : Background estimation (421 sec. in Matlab) + gradient computation (33.43 sec. in C++) + remaining portion of the algorithm (1031 sec. in Matlab).

**Proposed : Section 3.1, 3.2, 3.3 (35.8 sec. in C++) + Section 3.4, 3.6 (14 sec. in Matlab) + Section 3.5, 3.7, 3.8 (5.8 sec. in C++)

Table 3: Time required (in seconds) by the different algorithm on a diploma image of 7016×4960 pixels.

with some more recent state of the art binarization techniques. A dynamic window based local thresholding approach⁹ is proposed by Bataineh et.al [2]. The comparative results shown in Table 2 of this algorithm are not very promising. It can be seen in Fig. 3c that this technique is unable to properly remove the decorated background graphics. Our proposed technique has also outperformed the popularly known binarization technique by Gatos et.al [5], whose statistical accuracy is close to the one by Savoula et.al [11] (see Table 2).

Using structural symmetry of pixels (SSP), a fuzzy c-means clustering based binarization technique is recently proposed by Mondal et.al [10]. Although the accuracy shown in Table 2 is good but our proposed method outperforms it by reasonable margin. As shown in Fig. 5b, the approach by Mondal et.al misses character pixels which deforms the shape of characters and it is also unable to properly remove the background “couronne”. Another SSP based technique is proposed by Jia et.al [9] which has listed some of the best accuracies on DIBCO

data sets [11]. But this technique is also failed to properly remove the decorated background from diplomas ¹¹(see Fig. 5c and Table 2). Our algorithm also has outperformed this technique with a suitable margin.

The computational time of different algorithm is mentioned in Table 3. Due to the implementation facility, we have implemented the prototype of this algorithm using Matlab and C++¹². In comparison with other state of the art binarization algorithm e.g. Niblack [11], Gatos [5] etc., our technique takes more computational time but it is able to outperform these state of the art algorithms in terms of accuracy. Whereas, even using Matlab (slower than C++ or Python language) for implementing one portion of our algorithm, it is lot more faster than the algorithm by Jia et.al [9] ¹³ and Mondal et.al [10].

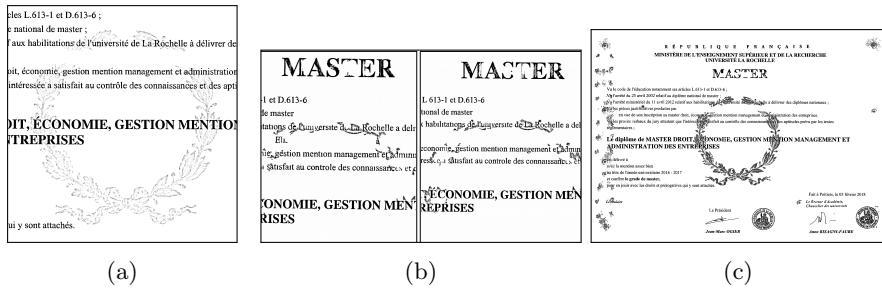


Fig. 5: (a): The result of Savoula binarization (b) Some results of binarization by technique of Mondal et.al.[10] (c) Binarization results by Jia et.al [9]

5 Conclusion

In this paper, we have proposed a technique to remove background and to obtain texts from French university diplomas. Our proposed method performs satisfactorily well on diploma images (see Table 3). All experiments are done by using Intel i7-8850H CPU, 32 GB RAM (in C++ and Matlab R2018a). It can be concluded that the proposed approach performs satisfactorily well on diploma images compared to many other state of the art techniques.

Acknowledgment

This work is supported by the Region Nouvelle Aquitaine and European Union under the project "Sécurisation et Authentification des Diplômes" in the "pro-

¹¹ We have used the author's provided executable but due to some unresolved memory related errors of OpenCV library, we were unable to run it on some images⁹.

¹² In future, we plan to make an optimal implementation of complete algorithm in C++, using OpenCV library.

¹³ we have used the optimized executable, provided by the authors

gramme opérationnel FEDER/FSE 2014-2020” (grant number P2017-BAFE-46). We highly thank the anonymous reviewers for their comprehensive reviews.

References

1. Ahmed, S., Liwicki, M., Dengel, A.: Extraction of Text Touching Graphics Using SURF. In: 2012 10th IAPR International Workshop on Document Analysis Systems. pp. 349–353. No. March, IEEE (mar 2012)
2. Bataineh, B., Abdullah, S.N.H.S., Omar, K.: An Adaptive Local Binarization Method for Document Images Based on a Novel Thresholding Method and Dynamic Windows. *Pattern Recognition Letters* **32**(14), 1805–1813 (2011)
3. Bolan Su, Shijian Lu, Chew Lim Tan: A Robust Document Image Binarization Technique for Degraded Document Images. *IEEE Transactions on Image Processing* **22**(4), 1408–1417 (2013)
4. Dollar, P., Zitnick, C.L.: Structured Forests for Fast Edge Detection. In: ICCV (2013)
5. Gatos, B., Pratikakis, I., Perantonis, S.: Adaptive degraded document image binarization. *Pattern Recognition* **39**(3), 317–327 (mar 2006)
6. Gerig, G., Kbler, O., Kikinis, R., Jolesz, F.A.: Nonlinear Anisotropic Filtering of MRI Data. *IEEE Transactions on Medical Imaging* (1992)
7. He, L., Nandi, A.K., Jia, X., Zhang, Y., Meng, H., Lei, T.: Significantly Fast and Robust Fuzzy C-Means Clustering Algorithm Based on Morphological Reconstruction and Membership Filtering. *IEEE Transactions on Fuzzy Systems* **26**(5), 3027–3041 (2018)
8. Jia, F., Shi, C., He, K., Wang, C., Xiao, B.: Document Image Binarization Using Structural Symmetry of Strokes. *ICFHR-2016* pp. 411–416 (2016)
9. Jia, F., Shi, C., He, K., Wang, C., Xiao, B.: Degraded document image binarization using structural symmetry of strokes. *Pattern Recognition* **74**, 225–240 (2018)
10. Mondal, T., Coustaty, M., Gomez-Krämer, P., Ogier, J.m.: Learning Free Document Image Binarization Based on Fast Fuzzy C-Means Clustering. In: ICDAR. pp. 1384–1389. IEEE (2019)
11. Pratikakis, I., Zagoris, K., Barlas, G., Gatos, B.: ICDAR2017 Competition on Document Image Binarization (DIBCO 2017). *ICDAR* **1**(Dibco), 1395–1403 (2018)
12. Sauvola, J., Pietikäinen, M.: Adaptive document image binarization. *Pattern Recognition* (2000)
13. Tensmeyer, C., Martinez, T.: Document Image Binarization with Fully Convolutional Neural Networks. *ICDAR* **1**, 99–104 (2018)
14. Vo, Q.N., Kim, S.H., Yang, H.J., Lee, G.: Binarization of degraded document images based on hierarchical deep supervised network. *Pattern Recognition* **74**, 568–586 (2017)