



**HAL**  
open science

# Learning Free Document Image Binarization Based on Fast Fuzzy C-Means Clustering

Tanmoy Mondal, Mickaël Coustaty, Petra Gomez-Krämer, Jean-Marc Ogier

► **To cite this version:**

Tanmoy Mondal, Mickaël Coustaty, Petra Gomez-Krämer, Jean-Marc Ogier. Learning Free Document Image Binarization Based on Fast Fuzzy C-Means Clustering. 2019 International Conference on Document Analysis and Recognition (ICDAR), Sep 2019, Sydney, Australia. pp.1384-1389, 10.1109/ICDAR.2019.00223 . hal-02555586

**HAL Id: hal-02555586**

**<https://hal.science/hal-02555586>**

Submitted on 27 Apr 2020

**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.

# Learning Free Document Image Binarization Based on Fast Fuzzy C-Means Clustering

Tanmoy Mondal, Mickaël Coustaty, Petra Gomez-Krämer, Jean-Marc Ogier  
L3i, La-Rochelle University, France  
{tanmoy.mondal, mickael.coustaty, petra.gomez, jean-marc.ogier}@univ-lr.fr

**Abstract**—In this paper, a novel local threshold binarization method using fast Fuzzy C-Means clustering is proposed. Historical document images with non-uniform background, stains, faded ink are first processed by removing the background using inpainting based method. Then using Fuzzy C-Means clustering is used to cluster out the pixels into three main clusters : sure text pixels, sure background pixels and confused pixels which may or may not be labeled as text. Based on the structural symmetry of pixels (SSP), these confused pixels are then classified into text or background pixels. The SSP is defined as those pixels around strokes whose gradient magnitudes are big enough and whose directions are opposite. As the gradient map is our basis for computing the SSP, we further propose to estimate the background surface first and to extract potential SSP in the compensated image so as to deal with degradations of document images such as uneven illumination, low contrast and stain. To prove the effectiveness of our method, tests on eight public document image datasets are performed and the experimental results show that our method outperforms other local threshold binarization approaches on both F-measure and PSNR.

**Keywords:** Binarization, Fuzzy C-Means, Background Removal, Stroke Width Estimation.

## I. INTRODUCTION

Document image binarization is one of the primary preprocessing steps for various document image processing methods. To minimize the impact of existing document degradation which often appears in historical documents and for many classical document image analysis tasks (e.g. page segmentation, OCR etc.), it is often desirable to first binarize the images. The topic of binarization is a well-studied problem, as evidenced by the popularity of the Document Image Binarization Contest (DIBCO) [1] and the Handwritten Document Image Binarization Competition (H-DIBCO), held from 2009 to present. These competitions aim to address document binarization problem by introducing challenging benchmarking datasets to evaluate the recent advancement. However, the best performing competition results show a good scope for improvement.

Many kinds of degradation can be present in historical document images which includes non-uniform background, stains, faded ink, ink bleeding through the page, and un-even illumination. In this paper, we propose a binarization technique<sup>1</sup> based on Fuzzy C-Means clustering and structural properties of text strokes. Initially the image is cleaned by estimating the background and then normalizing the original image by it. Then the normalized background image is processed by

the Fuzzy C-Means clustering technique which gives pixel clusters. At the end, these pixel clusters are classified based on the structural symmetry of pixels (SSP) (originally proposed by Jia et al. [2]) of pixel clusters. The SSP has the capacity to distinguish between text and non-text pixels. As shown in Fig. 1a, the SSP is defined as the pixels around the strokes, which contain both text and non-text candidate pixels and if these pixels are text pixels then they have opposite gradient directions. So, to classify pixels based on SSPs, we use the density, gradient direction and a neighborhood threshold based voting scheme of image clusters (connected components) to decide whether a pixel cluster is text or not. The overall architecture of the algorithm is mentioned in Fig. 1b. We have shown the robustness and accuracy of our proposed approach on 8 public dataset from popular DIBCO competition [1].

The remainder of this paper is organized by presenting the related recent state of the art techniques in section II. Then, we introduce the proposed binarization method and respective explanations in section III. The experimental results and discussions are presented in section IV and the conclusion is drawn in section V.

## II. RELATED WORK

Numerous document image binarization approaches have been proposed in the literature and are reported by continuous DIBCO competition reports [1]. By definition, text binarization means labeling of each pixel in the image as text or background. The existing binarization techniques in the literature can be categorized into two principal categories : i) learning-free and ii) learning-based approaches. Several binarization techniques have been proposed in the past decade which are learning free and are based on hand-crafted image processing steps. But very recently the trend in document binarization has shifted to machine learning based methods, mostly relying on deep-learning based techniques for image binarization. A LSTM based binarization technique proposed by Afzal et al. in [3], incorporates both local and global information. By considering the image as a 2D sequence of image pixels, a 2D LSTM is employed for the classification of each pixel as text or background. A CNN based technique is proposed in [4] to classify each image pixel as foreground or background from the sliding window of size  $19 \times 19$ , centered at the pixel to be classified. They reported an F-measure of 87.74 on *DIBCO-2013* compared to 92.90 achieved by the competition winner. Tensmeyer and Martinez [5] proposed

<sup>1</sup>The source code is available at : [https://github.com/tanmayGIT/Document\\_Binarization\\_SSP](https://github.com/tanmayGIT/Document_Binarization_SSP)

a method by applying CNN at multiple image scales. Their model is trained to optimize a continuous version of the pseudo F-Measure metric. Another similar approach as the one in [3] is proposed in [6], which is based on a grid based LSTM cell and it also use a pseudo F-Measure based weight loss function.

Although these aforementioned learning based approaches have shown nice performance but due to the compulsory requirement of a sufficient amount of training data which is costly and cumbersome to produce, learning based binarization techniques would not be useful in case of less/no amount of ground truth data. Recently the learning-free approaches [2] have shown high potential and comparable accuracy with respect to learning-based approaches. Jia et al. [2] proposed an approach based on structural symmetry of pixels (SSP) from text strokes for binarization. A local threshold value is computed by using structural symmetric pixels (SSP) of the region so as to suppress the non-text pixels and also to maintain the text ones. Howe proposed a method which formulates image binarization as energy minimization over a MRF [7]. The unary energy terms are computed from the Laplacian and a pairwise connection is determined by Canny edge detection. The exact minimization of the energy function can be obtained by solving the equivalent Max Flow Problem [7]. A variant of [7] is proposed in [8] where the authors use statistical racing procedure, named *I/F-Race*, to automatically tune the parameters for two binarization algorithms reasoned on the perception of objects by distance (POD) and POD combined with a Laplacian energy based technique. By combining the local image contrast and the local image gradients to determine edges between text and background, a method is proposed by Su et al. in [9] which won the *DIBCO-13* competition.

### III. PROPOSED METHOD

The complete architecture of the proposed algorithm is mentioned in Fig. 1a. Given a document image, if in color (see Fig. 2a), it is first converted to gray scale ( $\mathcal{I}(x, y)$ ) and then the following technique is used for stroke width calculation.

#### A. Stroke Width Calculation

We have taken the approach mentioned in [9] for the calculation of stroke width. The underlying idea is to compute an adaptive local contrast image based on the neighboring maximum and minimum intensity value of each pixel to properly detect the stroke edge pixels of the document text. Then edges are detected based on the combination of Canny's edge detector and Otsu's thresholding technique. After that the stroke width ( $W_{stroke}$ ) is calculated by traversing each image row and by finding the distance between two edge pixels (whose binary image value is 1 and has higher gray level intensity than the next pixel, having binary value 0 and has lower intensity) and then averaging such distances all together.

#### B. Background Image Estimation

As the original image may contain some unnecessary background noise such as bleed-through background, uneven illumination and ink stains etc. Therefore, the background surface

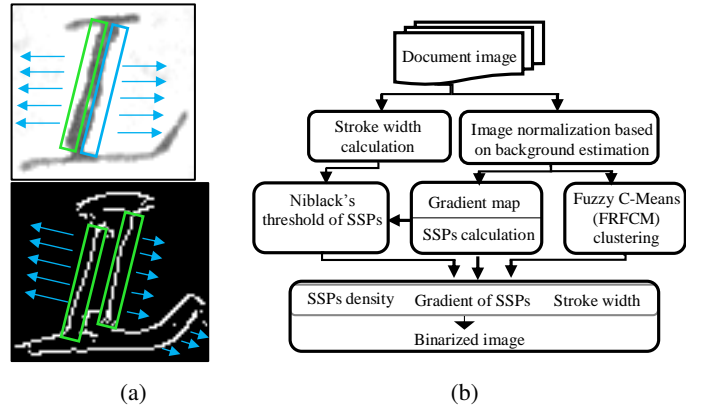


Figure 1: (a) The illustration of structural symmetric pixels (SSP). Top: In the original image, pixels around the stroke contain both text and background candidate pixels. Bottom: The SSP candidate pixels and each of such pixels should have opposite gradient direction (shown in blue colored arrow) (b) The overall architecture of the proposed algorithm.

is first estimated to compensate the variation of degradation by using the algorithm of Ntirogiannis et al. [10]. The underlying idea is to perform image inpainting by using Niblack's binarized image as a mask. The inpainting is performed in 5 passes. The first 4 passes are performed in *LRTB*, *LRBT*, *RLTB* and *RLBT* directions where *L*, *R*, *T* and *B* refer to *Left*, *Right*, *Top* and *Bottom* respectively. In each *pass* and for each *mask* or foreground pixel, an average of *non-mask* pixels in the 4-connected neighborhood (cross-type) is calculated and this new pixel value is considered as a *non-mask* pixel for the consecutive computation of the remaining passes. At the final (fifth) image pass, all the four images from the four previous inpainting passes are considered and the minimum intensity out of four values are kept for each pixel as the estimation of background surface. After that, the document degradation compensation or normalization ( $I_{norm}(x, y)$ ) is performed as:

$$I_{norm}(x, y) = \begin{cases} 255 \times \frac{I(x, y)}{B(x, y)} & I(x, y) < B(x, y); B(x, y) > 0 \\ 255 & otherwise \end{cases} \quad (1)$$

where  $I$  and  $B$  denote the original and estimated background image respectively. It can be seen from the background removed image shown in Fig. 2b that the noisy background can be properly removed and low contrast stroke edge pixels caused by the ink stain of the original image are restored.

1) *SSP Extraction*: As mentioned before, the SSPs are defined as the pixels around the stroke which contain both text and background pixels. The text pixels can be distinguished by their large gradient magnitude and opposite gradient orientations. As the image contains noisy background (e.g. faint character and bleed-through background) so even after background normalization, one single threshold (e.g. using Otsu's threshold as in [2]) can not properly find SSP candidates when the normalized image does not follow a bimodal intensity

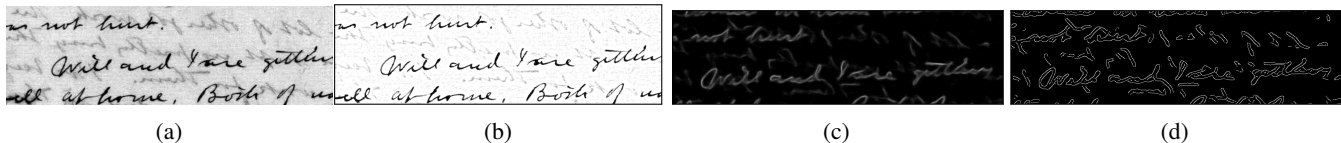


Figure 2: Various preprocessing steps: (a) Original gray image. (b) Background removed image. (c) FRFCM clustering on background removed image. (d) Image with SSP candidates

distribution. So, we need a better pixel gradient mapping technique to maintain the completeness of the character edges which should be neither broken nor fused.

By considering the above mentioned complexities, we propose to use a real time and robust gradient computation technique, originally proposed by Dollár et al. [11]. The gradient computation (see Fig. 2c) is performed based on the present structure in local image patches and by learning both an accurate and computationally fast edge detector. This technique is order of magnitude faster than many competing state of the art approaches. It obtains real time performance and shows robust results. After obtaining the gradient image, we apply Canny’s edge detection technique which uses bimodal thresholds to detect the edges and these edge pixels are regarded as SSP candidates. Let’s denote the image with SSP candidates as  $I_{SSP}$  where pixels of SSP candidates are labeled as 1 and others are labeled as 0 (see Fig. 2d).

$$I_{SSP}(p) = \begin{cases} 1 & p \in \text{potential SSP} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

### C. Fuzzy C-Means Clustering Algorithm

The pixels of the image with suppressed background are clustered by using the Fuzzy C-Means clustering technique. Based on fuzzy set theory, the Fuzzy C-mean (FCM) [12] is one of the most used methods for image segmentation and its success chiefly attributes to the introduction of fuzziness for the belongingness of each pixel. It is superior to hard clustering as it has more tolerance to ambiguity and retains better original image information. We use a fast and robust Fuzzy C-Means clustering named as *FRFCM Clustering* for our work.

1) *FRFCM Clustering*: It employs morphological reconstruction to smooth images in order to simultaneously improve noise immunity and image detail preservation. FRFCM modifies the membership partition by using a faster membership filtering instead of the slower distance computation between pixels with local spatial neighbors and their clustering centers which leads to low computational complexity. Therefore, FRFCM is faster than other improved FCM algorithms.

The FRFCM clustering is applied on  $I_{norm}$  and the image pixels are divided into 5 clusters called as *cluster-1*, *cluster-2*, *cluster-3*, *cluster-4* and *cluster-5*. The pixels in *cluster-1* have an intensity closer to 0 so these pixels can be surely labeled as *Sure Text Pixels* and the pixels in *cluster-5* can be labeled as background pixel as their values are close to white or background. Whereas the pixels in *cluster-2* and *cluster-3* have an intensity close to *cluster-1* but not very far from

*cluster-5*, so we name these pixels as confused pixels. The pixels in *cluster-4* are close to *cluster-5* and far from *cluster-1* so we also label these pixels as background pixels (*cluster-5*).

### D. SSP based threshold selection

Inspired by the idea in [2], we have used the structural properties of strokes (SSP) to classify the pixels of *cluster-2* and *cluster-3*. A blank image of the same size as the original image is initially formed and named as  $I_{Bin}$ . All the pixels belonging to *cluster-1* (which we call as sure text pixels) are labeled as foreground pixels in  $I_{Bin}$  and all the pixels from *cluster-4* and *cluster-5* (which we call as sure background pixels) are labeled as background pixels in  $I_{Bin}$ . Now the task is to correctly label pixels belonging to *cluster-2* and *cluster-3* as text or non text pixels. In the following section, we propose a technique to correctly label such pixels (first for the pixels in *cluster-2* and then for the pixels in *cluster-3*) as either foreground or background using the concept of SSPs. Although the concept of SSP is taken from [2] but it is modified and adapted according to our need by hierarchically labeling confused text pixels from *cluster-2* and *cluster-3*.

1) *Judging the density of potential SSP*: The SSPs are surrounded by text and non-text pixels. Whereas, potential text pixels should be surrounded by many SSP candidate pixels. So, we calculate the number of SSP around a pixel  $p$ . If this number is smaller than a threshold, then  $p$  will be defined as background (0). This process is mentioned as follows:

$$N_{total}(p) = \sum_{q \in N_p} S(q) \quad (3)$$

$$I_{Bin}(p) = \begin{cases} 1 & N_{total}(p) \geq \alpha \times W_{stroke} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where  $N_p$  is the neighborhood window of the pixel  $p$  and  $N_{total}(p)$  denotes the total number of local SSPs corresponding to pixel  $p$ .  $W_{stroke}$  denotes the stroke width in the image, which is an automatically calculated important parameter and has been used crucially for correctly labeling a pixel.  $\alpha$  is a coefficient of threshold which is taken as 0.3 in this work. So, the Equation 4 signifies that if any pixel has more than 30% SSP pixels around it then it is considered as a text pixel.

We observe that, instead of labeling each pixel individually, it is more logical to label a group of pixels, which belongs to the same cluster and are connected together. Because it was seen experimentally that due to the conditional check of Equation 4, some pixels were qualified as text and others were not, even if they were all connected. Hence, we propose to

label a set of connected pixels (by finding CC) together if the percentage of total number of pixels ( $\zeta$ ) of any CC can satisfy the condition mentioned in Equation 4. The value of  $\zeta$  is empirically set 0.3 which means that if at least 30% of the pixels in a CC satisfies the constraint, mentioned in Equation 4 then all the pixels of that component are labeled as text.

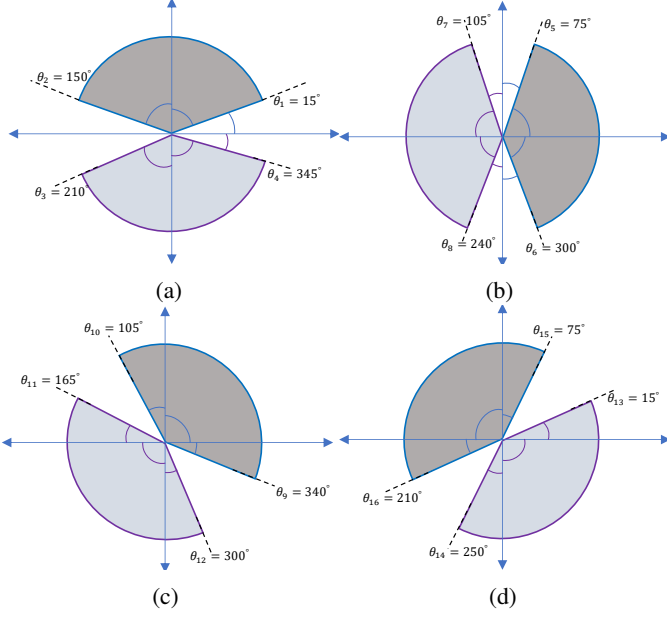


Figure 3: The eight angle ranges of gradient orientations for direction symmetry judgments.

2) *Judging the symmetry of potential SSP*: The local SSP corresponding to text pixels must also satisfy the direction symmetry constraints otherwise the pixel should belong to the background class. Therefore, we label any confused pixel  $p$  as background if its neighborhood SSP have symmetric gradient orientation because to be a text pixel, the neighborhood SSP should have opposite gradient orientation (see Fig. 1a).

The symmetry of any confused pixel is computed by dividing the angle plane of  $360^\circ$  degrees with a range of  $135^\circ$  degrees. It can be seen in Fig. 3 that the angle plane is divided into 8 groups ( $AnglePlane_{i=1,2,\dots,8}$ ) (see Equation 5).

$$N_{orint}(p, i) = \sum_{q \in N_p, Orint(q) \in A_i} S(q) \quad (5)$$

where  $Orint(q)$  denotes the gradient orientation of the pixel  $q$ . The gradient orientation can belong to any of these 8 angle groups ( $AnglePlane_{i=1,2,\dots,8}$ ). The total number of pixels belonging to each of these angle groups is calculated and their maximum is computed (shown in Equation 6) to determine which group has the maximum number of pixels.

$$N_o(p, i) = \max_i N_{orint}(p, i) \quad (6)$$

where  $N_o(p)$  denotes the count of maximum number of pixels for each group. Therefore, to be a text pixel, it needs to satisfy

the SSP density and gradient symmetry condition together:

$$I_{Bin}(p) = \begin{cases} 1 & N_o(p) \geq \beta \times N_{total}(P) \\ 0 & otherwise \end{cases} \quad (7)$$

The  $\beta$  is a coefficient threshold which is set to 0.75 as in [2].

3) *Niblack threshold based pixel voting*: In addition to the checking of above mentioned two constraints, the pixels are also checked based on a multiple neighborhood threshold based voting system. For each SSP pixel, we calculate a threshold value, based on its neighborhood (of the size of the stroke width). The threshold ( $\mathfrak{T}(x, y)$ ) is calculated by using the following mentioned Niblack technique [10]. A Fixed parameter setting is used here i.e. the window size  $w$  is taken as the stroke width and  $k$  is set to  $-0.2$ . The  $m(x, y)$  and

$$\mathfrak{T}(x, y) = m(x, y) + k \times \sigma(x, y) \quad \forall (x, y) \in SSP \quad (8)$$

$\sigma(x, y)$  are the mean and standard deviation of the intensity of SSP within a neighborhood window around the pixel  $(x, y)$ .

For each candidate pixel ( $p$ ) which has passed the above mentioned two conditions (Equation 6 and 7) are then verified through this threshold based voting system condition also. For each candidate pixel ( $p$ ), we find the SSPs, exists at it's neighborhood (of size equal to stroke width) and take into account the threshold values associated to each of the neighborhood SSPs. Then a voting is given to this pixel ( $p$ ) by checking it's gray scale intensity against the calculated thresholds of surrounding SSP pixels. If the gray value of  $P$  is greater than each neighborhood SSP thresholds then the score is incremented by 1 otherwise it is decremented by 1.

$$score(p) = \begin{cases} score(p) + 1 & \text{if } I_{norm}(p) < Thresh(x, y) \\ score(p) - 1 & otherwise \end{cases} \quad (9)$$

$$I_{bin}(p) = \begin{cases} 1 & \text{if } score(p) > 1 \\ 0 & otherwise \end{cases} \quad (10)$$

As mentioned in Equation 10, if any pixel has a score more than 0, then it is considered as text or foreground pixel. So, if  $\zeta$  number of pixels of any CC qualify the criteria then all the pixels of that particular CC are labeled as text otherwise they are labeled as background. This voting scheme is proposed to qualify each candidate pixels ( $p$ ) based on the votes given by neighboring SSPs. The idea here is to qualify as text pixel, each candidate pixels should be voted by multiple SSP thresholds and this voting is done by comparing any pixel's ( $p$ ) intensity with the thresholds of surrounding SSPs.

Until now, we have labeled the "confused pixels" from *cluster-2* in the above described process (see Fig. 4b). It can be seen that except the background (white), the foreground pixels have three different intensities (see the zoomed portion in Fig. 4b), where the darker ones are sure text pixels. The image formed by mapping only darker pixels is shown in Fig. 4c (at left). After labeling the confused pixels from *cluster-2*, the binary image is shown at the middle in Fig. 4c (see the green circled portion to visualize the inclusion of pixels from

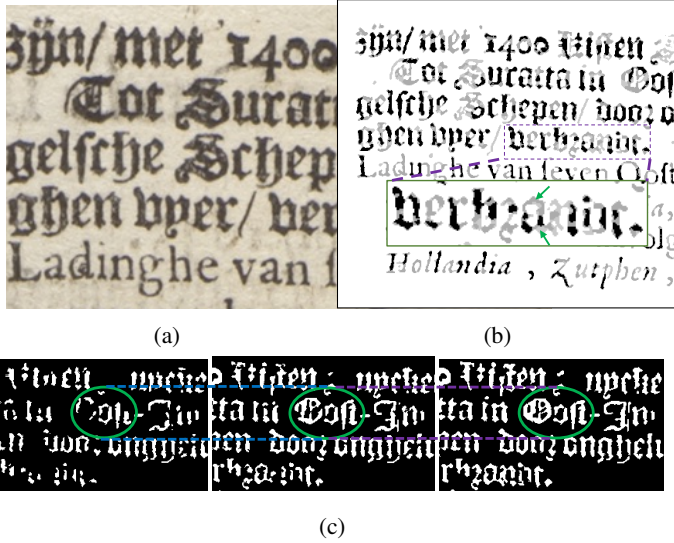


Figure 4: (a) The original image. (b) The clustered pixels based on the suppressed background image. The green colored highlighted portion of image show two different shades of clustered pixels denoted as the pixels of *cluster-2* and *cluster-3*. (c) Left: only the sure text pixels. Middle: sure and qualified text pixels from *cluster-2*. Right: the final binarization result after qualifying the pixels from *cluster-3* and post-processing.

*cluster-2* in comparison to the left image). Now to label the pixels from *cluster-3*, we use the middle image of Fig. 4c. All the foreground pixels (say  $I_{bin}^{strong}$ ) are named as “strong pixels”. Before checking the three aforementioned conditions (refer to Equation 4, 7 and 10), we check whether  $\zeta$  times of pixels of any CC are surrounded by more than  $(\alpha \times N_p)$  amount of “strong pixels” (see Equation 12) to label the pixels from any particular component of *cluster-3*.

$$N_{total}^{strong}(p) = \sum_{q \in N_p} I_{bin}^{strong}(q) \quad (11)$$

where  $N_{total}^{strong}(p)$  denotes total number of strong pixels around any pixel and  $N_p$  denotes the surrounding pixels.

$$I_{Bin}(p) = \begin{cases} 1 & N_{total}^{strong}(p) \geq \alpha \times N_{total}(p) \\ 0 & otherwise \end{cases} \quad (12)$$

Where  $N_{total}(p)$  is the total number of neighborhood pixels and  $\alpha$  is empirically set as 0.3. If this condition (Equation 12) is fulfilled then only the other three conditions (refer to Equation 4, 7 and 10) are checked for each pixel. Likewise, all the pixels of any CC are labeled as text if and only if  $\kappa$  ( $\zeta \times$  total number of pixels in CC) amount of pixels fulfill all of these four conditions.

After obtaining the foreground/binary image by the above described process, it is further improved by removing the small and noisy artifacts based on its size and neighborhood condition. If the size of any CC is less than the half of the stroke width and it has no SSP candidate pixels at its neighborhood

(diameter is equal to stroke width) then it is removed as noisy artifact (the final binary image is shown at the right side of Fig. 4c). The big artifacts are already removed thanks to the gradient orientation based filtering and as the pixels are validated based on the local SSP (having neighborhood size equals to twice of the stroke width) threshold based voting scheme, so therefore the obtained binary image (Equation 12) cannot contain large artifacts. So, only small noisy artifacts have to be removed as the post-processing step.

## IV. RESULTS AND DISCUSSION

The proposed method is evaluated on several public datasets from the previous document image binarization competitions: DIBCO’09, DIBCO’11, DIBCO’13, H-DIBCO’10, H-DIBCO’12, H-DIBCO’14, H-DIBCO’16 and DIBCO’17. The first three and last datasets contain both handwritten and machine-printed images while the other ones only contain handwritten images (for more details, see in [1]). We have used the same evaluation metrics as the ones usually used in the DIBCO competitions. These metrics are F-measure, Peak Signal-to-Noise Ratio (PSNR) and Distance Reciprocal Distortion metric (DRD). The experimental results are shown in Table-II. The proposed method is compared with several popular binarization methods on the 8 datasets. In Table. II,  $\uparrow$  implies that a higher value of F-measure and PSNR is desirable whereas the  $\downarrow$  implies that a lower value of DRD is desirable. It can be seen that the proposed method achieves good scores with respect to F-measure PSNR and DRD compared to other best performing methods from the literature. It can be seen that although in some cases the method proposed by Howe [7] and Lerore [1] performed better, but in the case of DIBCO-14 and DIBCO-17 dataset, our method has shown interesting and competitive results.

### A. Comparison with other SSP based technique

As mentioned earlier, the concept of SSP for document image binarization is introduced by Jia et al. [2]. In their approach all image pixels are firstly considered as text pixels and then each of these pixels are checked against the SSP density and orientation based criteria. Due to this reason this technique is computationally expensive compared to our algorithm where initial clustering of pixels by Fuzzy C-Means based clustering technique helps to classify only few pixels. Moreover, unlike our algorithm, the technique of Jia et al. [2] is a pixel based labeling technique, which does not take into consideration belongingness of any single pixels in a group of similar pixels. In our case, thanks to an initial clustering the belongingness of pixel is considered and are processed in a group instead of treating them individually, which highly fastens up the pixel labeling process.

### B. Time Complexity

The speed of our method highly depends on the image size so we estimate the runtime in second per mega pixel. All the experiments were performed on an Intel Core i5 processor with 16 GB RAM computer using Matlab 2018a. Due to the

unavailability of computational time of the state of the art techniques it becomes little difficult to compare our technique with others. Therefore, in the following Table I, we mention the time needed<sup>2</sup> to perform the primary processing steps of the algorithm for an image of size  $370 \times 870$  pixels.

Table I: Time required by the different steps of the algorithm.

| Algorithmic Steps   | Required Time (in secs) |
|---|-------------------------|
| Stroke width calculation  | 0.3                     |
| Background estimation and image normalization   | 3.5                     |
| FRFCM clustering  | 0.23                    |
| Other remaining steps (labeling pixels from <i>cluster-2</i> and <i>cluster-3</i> , noisy artifacts removal etc.) | 4.16                    |
| <b>Total time taken</b>   | 8.3 (approx.)           |

## V. CONCLUSION

In this paper, we have proposed a robust and fast binarization technique based on Fuzzy C-Means clustering and structural properties of text strokes. The proposed technique is robust and learning free except some simple parameter to be settled ( $\alpha$ ,  $\beta$  etc.). The experimental results are shown on challenging DIBCO and HDIBCO datasets to demonstrate the effectiveness of the proposed technique. The experimental results shows that we have achieved competitive accuracy.

## ACKNOWLEDGMENT

This work is supported by the region “Nouvelle Aquitaine” and “European Union” under the project “S curisation et Authentification des Dipl mes” in the “programme op rationnel FEDER/FSE 2014-2020”(Grant Number : P2017-BAFE-46).

## REFERENCES

- [1] I. Pratikakis, K. Zagoris, G. Barlas, and B. Gatos, “ICDAR2017 Competition on Document Image Binarization (DIBCO 2017),” *ICDAR*, vol. 1, no. Dibco, pp. 1395–1403, 2018.
- [2] F. Jia, C. Shi, K. He, C. Wang, and B. Xiao, “Document Image Binarization Using Structural Symmetry of Strokes,” *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*, pp. 411–416, 2016.
- [3] M. Z. Afzal, J. Pastor-Pellicer, F. Shafait, T. M. Breuel, A. Dengel, and M. Liwicki, “Document Image Binarization using LSTM: A Sequence Learning Approach,” in *HIP ’15*. New York, New York, USA: ACM Press, 2015, pp. 79–84.
- [4] J. Pastor-Pellicer, S. Espa na-Boquera, F. Zamora-Mart nez, M. Zeshan Afzal, M. J. Castro-Bleda, M. Z. Afzal, M. J. Castro-Bleda, M. Zeshan Afzal, and M. J. Castro-Bleda, “Insights on the Use of Convolutional Neural Networks for Document Image Binarization,” in *Lecture Notes in Computer Science*, vol. 9095. Springer, 2015, pp. 115–126.
- [5] C. Tensmeyer and T. Martinez, “Document Image Binarization with Fully Convolutional Neural Networks,” *ICDAR*, vol. 1, pp. 99–104, 2018.
- [6] F. Westphal, N. Lavesson, and H. Grahn, “Document Image Binarization Using Recurrent Neural Networks,” *DAS 2018*, pp. 263–268, 2018.
- [7] N. R. Howe, “Part-Structured Inkball Models for One-Shot Handwritten Word Spotting,” *ICDAR*, pp. 582–586, aug 2013.

<sup>2</sup>note that the time can be reduced by proper Matlab based code optimization which is not done for this research work.

Table II: Result on DIBCO Datasets(best results are shown in bold and if the proposed algorithm is performed well then it is shown in cyan color.)

| Dataset  | No. of Images | Method Name    | F-Measure (%) (↑) | PSNR (↑)     | DRD(↓)      |
|----------|---------------|----------------|-------------------|--------------|-------------|
| DIBCO_09 | 10            | Proposed       | 81.33             | 16.05        | 4.29        |
|          |               | Lelore[1]      | <b>94.05</b>      | 21.43        | 2.11        |
|          |               | Su[1]          | 89.76             | 19.55        | 4.19        |
|          |               | Jia[2]         | 91.37             | 18.49        | -           |
|          |               | Howe[1]        | 93.73             | <b>21.85</b> | <b>2.10</b> |
| DIBCO_10 | 10            | Proposed       | 86.80             | 15.35        | 4.99        |
|          |               | Lelore[1]      | <b>94.05</b>      | 21.43        | 2.11        |
|          |               | Su[1]          | 89.76             | 19.55        | 4.19        |
|          |               | Howe[1]        | 93.73             | <b>21.85</b> | <b>2.10</b> |
| DIBCO_11 | 16            | Proposed       | 83.68             | 16.41        | 4.85        |
|          |               | Lelore[1]      | <b>94.05</b>      | 21.43        | 2.11        |
|          |               | Su[1]          | 89.76             | 19.55        | 4.19        |
|          |               | Howe[1]        | 93.73             | <b>21.85</b> | <b>2.10</b> |
| DIBCO_12 | 14            | Proposed       | 86.50             | 18.33        | 3.15        |
|          |               | Lelore[1]      | <b>94.05</b>      | 21.43        | 2.11        |
|          |               | Su[1]          | 89.76             | 19.55        | 4.19        |
|          |               | Howe[1]        | 93.73             | <b>21.85</b> | <b>2.10</b> |
| DIBCO_13 | 16            | Proposed       | 87.97             | 18.45        | 4.82        |
|          |               | Lelore[1]      | 90.78             | 20.54        | 3.59        |
|          |               | Su[1]          | 87.70             | 19.59        | 4.21        |
|          |               | Jia[2]         | 89.50             | 19.30        | -           |
|          |               | Howe[1]        | <b>91.34</b>      | <b>21.29</b> | <b>3.18</b> |
| DIBCO_14 | 10            | Proposed       | 96.19             | 20.62        | 2.08        |
|          |               | Lelore[1]      | 96.14             | 21.88        | 1.25        |
|          |               | Su[1]          | 94.38             | 20.31        | 1.95        |
|          |               | Howe[1]        | <b>96.49</b>      | <b>22.24</b> | <b>1.08</b> |
| DIBCO_16 | 10            | Proposed       | 83.02             | 16.29        | 6.47        |
|          |               | Lelore[1]      | 87.21             | 17.36        | <b>5.27</b> |
|          |               | Su[1]          | 84.75             | 17.64        | 5.64        |
|          |               | Howe[1]        | <b>87.47</b>      | <b>18.05</b> | 5.35        |
| DIBCO_17 | 20            | Proposed       | 86.27             | 15.44        | <b>5.70</b> |
|          |               | Wei et al. [1] | <b>86.49</b>      | 16.33        | 6.57        |

- [8] R. G. Mesquita, R. M. A. Silva, C. A. B. Mello, and P. B. C. P. B. Miranda, “Parameter Tuning for Document Image Binarization Using a Racing Algorithm,” *Expert Systems with Applications*, vol. 42, no. 5, pp. 2593–2603, apr 2015.
- [9] Bolan Su, Shijian Lu, and Chew Lim Tan, “A Robust Document Image Binarization Technique for Degraded Document Images,” *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1408–1417, 2013.
- [10] K. Ntirogiannis, B. Gatos, and I. Pratikakis, “A Combined Approach for the Binarization of Handwritten Document Images,” *Pattern Recognition Letters*, vol. 35, no. 1, pp. 3–15, oct 2014.
- [11] P. Dollar and C. L. Zitnick, “Structured Forests for Fast Edge Detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013.
- [12] L. He, A. K. Nandi, X. Jia, Y. Zhang, H. Meng, and T. Lei, “Significantly Fast and Robust Fuzzy C-Means Clustering Algorithm Based on Morphological Reconstruction and Membership Filtering,” *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 5, pp. 3027–3041, 2018.