Pattern Recognition 45 (2012) 2419–2431

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

AdOtsu: An adaptive and parameterless generalization of Otsu's method for document image binarization

Reza Farrahi Moghaddam*, Mohamed Cheriet

Synchromedia Laboratory for Multimedia Communication in Telepresence, École de Technologie Supérieure, Montreal, QC, Canada H3C 1K3

ARTICLE INFO

Article history:
Received 31 May 2010
Received in revised form
25 October 2011
Accepted 5 December 2011
Available online 23 December 2011

Keywords:
Document image processing
Binarization
Adaptive methods

ABSTRACT

Adaptive binarization methods play a central role in document image processing. In this work, an adaptive and parameterless generalization of Otsu's method is presented. The adaptiveness is obtained by combining grid-based modeling and the estimated background map. The parameterless behavior is achieved by automatically estimating the document parameters, such as the average stroke width and the average line height. The proposed method is extended using a multiscale framework, and has been applied on various datasets, including the DIBCO'09 dataset, with promising results.

1. Introduction

The binarization of document images is an old but challenging problem for the Document Image Analysis and Retrieval (DIAR) community [25,13,28]. Its goal is to segment the pixels on the document image into just two classes, regardless of the enormous number of possible text typefaces and the various types of degradation, which make it an ambitious process. Most of the possible approaches have been attempted to solve this problem, and state-of-the-art methods have been evaluated in several papers [35,13,41]. Although binarization-free document image processing has also been considered [23,44], the complexity reduction achieved by the binarization process is the main motivation for including this process in virtually all document processing work flows.

One of the major categories of binarization methods is the threshold-based method [30,34,10], which is very popular, since most types of images can be converted to gray-level images. Because of the high degree of variability on document images, especially on historical ones, adaptive threshold-based methods are required. Otsu's method is a powerful, parameterless method belonging to this category, but it is non-adaptive. Although there are many locally adaptive methods, such as Sauvola's method [34], their parameters are hard to estimate, which means that they do not perform well on historical documents with a high degree of variability. In [10], an adaptive form of Otsu's method is introduced that uses the global Otsu threshold to avoid non-text regions. Although the method is promising, it still depends on global threshold, which is not a good measure for differentiating between text and non-text regions for document images with very variable text and background. At the same time, the multi-level classifiers [9] have demonstrated their ability to improve the performance of the enhancement and restoration models and techniques. They have also contributed new terms and concepts to the modeling vocabulary, which simplifies the formulation of the models.

In this work, a novel, parameterless adaptive form of Otsu's method is introduced. The proposed binarization method, which we refer to as the AdOtsu method, is based on a multi-level classifier, the estimated background introduced in this work. The estimated background is calculated using a novel multiscale approach starting from a rough, binarized initialization. The estimated background is then improved in a bootstrap approach using the AdOtsu method itself. The AdOtsu method is then generalized to the multiscale form using the framework developed in [10]. The multiscale framework is also improved by removing the bias of the core text pixels. Grid-based modeling is used to reduce the computational cost, and a new skeleton-based postprocessing is considered to remove the remaining artifacts and sub-strokes.

The organization of the paper is as follows. In Section 2, the threshold-based methods are briefly discussed. In Section 3, the problem statement is presented. The proposed AdOtsu method is introduced in Section 4. The method is then generalized to the multiscale framework in Section 5. In Section 6, background estimation is discussed. Postprocessing steps are detailed in Section 7. Section 8 presents the subjective and objective experimental results. Finally, the conclusion and future prospects are provided in Section 9.

* Corresponding author. Tel.: +1 514 396 8972; fax: +1 514 396 8595.
E-mail addresses: rfarrahi@synchromedia.ca, imriss@yahoo.com (R. Farrahi Moghaddam), mohamed.cheriet@etsmtl.ca (M. Cheriet).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.12.013

2. Related work

The proposed method, which we call AdOtsu, is an Otsu-based adaptive method. In this section, we briefly review Otsu's method and some of the adaptive methods.

2.1. Otsu's method

Otsu's method is a parameterless global thresholding binarization method. It assumes the presence of two distributions (one for the text and another for the background), and calculates a threshold value in such a way as to minimize the variance between the two distributions [30,45]. The two-distribution limit of Otsu's method was removed in [45], where the degradation modes on the histogram of the image are discarded one by one by recursively applying Otsu's method until only one mode remains on the image. In another work, the global restriction of the method is removed [10] and an adaptive method is introduced which uses the same concept as Otsu's method, but on local patches. A measure based on the global Otsu threshold was used in that work to reveal the non-text regions that have only one class of pixels. As mentioned in Introduction, that method still requires with the global threshold implicitly (see Section 2.11 for the details). In this work, we propose a new adaptive method which uses the estimated background to identify the non-text regions. In this way, the method is well-adapted to the input image.

Below, two of the locally adaptive thresholding methods are discussed.

2.2. Sauvola's method

Among the adaptive threshold binarization approaches, Sauvola's method [34] is one of the best known. In this method, the threshold value, inspired by Niblack's method [29,41], has been modified in order to capture open non-text regions [34]. The threshold has two parameters to set and estimate.

2.3. Gatos' method

One of the state-of-the-art binarization methods is introduced in [15]. In this method, a rough binarization of the document image is obtained first (usually using Sauvola's method). Then, a rough background is estimated (see Section 2.12 below). In the next step, local threshold values are calculated, based on the estimated background, as well as some parameters. These threshold values are used to calculate the final binarization, which is postprocessed to remove noise.

2.4. Edge-based local thresholding method

This binarization method was proposed by Su, Lu, and Tan, and placed first in the DIBCO'09 binarization contest [14]. The method consists of four steps: (i) background extraction by polynomial fitting on the rows; (ii) stroke edge detection using Otsu's method on gradient information; (iii) local thresholding by averaging the detected edge pixels within a local neighborhood window; and, finally, (iv) postprocessing of the result.

2.5. Toggle mapping binarization method

The method that placed second at DIBCO'09 was proposed by Fabrizio and Marcotegui [14]. It is based on the toggle mapping morphological operator [7]. To avoid the salt-and-pepper noise associated with toggle mapping, they excluded from the analysis the pixels whose erosion and dilation are too close. Pixels are then classified as text, background, and uncertain. The uncertain pixels are assigned to text and background, according to their boundary class.

2.6. Level set binarization method

The method that placed third at DIBCO'09 was proposed by Rivest-Hénault, Farrahi Moghaddam, and Cheriet [33,14]. This method uses the level set framework to locate the boundaries of text strokes and binarize a document image [33]. Like the others, this algorithm consists of several steps: (i) initialization using a stroke map (SM) [9]; (ii) correction of the SM using the level set framework in erosion mode and local linear models; and finally, (iii) a second round of level set operations, this time with a stroke gray level force, which provides the final text regions as the interior regions of the level set function.

2.7. Learning-based clustering methods

Although document images may suffer from severe and variable degradation, it may be assumed that there are regions on them that could be labeled as true text or background. This hypothesis has been the foundation of many learning methods, which start from a rough estimation of the text and background regions, and then attempt to learn their behavior, in order to classify regions that are in the confusion interval [6,39,37,3,12,7,20]. For example, a simple thresholding has been used to identify the text and background classes in [6]. Then, a noise model is built and used to adjust the threshold value. In [39], a framework is presented which uses any binarization method to identify three classes; namely, text, background, and uncertain pixels. Then, it reclassifies the uncertain pixels using a classifier trained using the text and background classes. We refer to this framework as a self-learning framework in Table 3.

2.8. Local maximum and minimum method

In this method [38], image contrast, defined based on the local maxima and minima, is used to detect high contrast image pixels instead of the image gradient. Image contrast is less sensitive to uneven illumination. Then, the document is segmented using a local threshold estimated based on the image contrast.

2.9. Hypercomponent tree method

This method [32], which is an extension of the component tree based on flat zones to hyperconnections, defines the tree by a special order on the hyperconnections and allows non-flat nodes. The steps of the method are as follows: (i) removal of the background using a hypercomponent tree; (ii) adaptive thresholding based on the values of the image edges, which are detected using the Sobel operator with an Otsu thresholding; and, finally, (iii) postprocessing.

2.10. The multiscale grid-based Sauvola method [10]

Thanks to the grid-based modeling introduced in [10], the computational cost of Sauvola's method can be reduced significantly. This enables the introduction of the multiscale grid-based Sauvola method in [10], which is capable of capturing the text pixels on high scales and track them on the lower scales in order to avoid strongly interfering patterns. In this work, we use a similar multiscale approach combined with the AdOtsu method to improve its performance.

2.11. The multiscale grid-based Adaptive Otsu method [10]

In [10], an adaptive and local version of Otsu's method has been introduced by combing Otsu's threshold and its local version in an intuitive way

$$T_{\text{Adaptive Otsu},u}(x)=\Theta\left(\frac{|T_{\text{Global Otsu},u}-T_{\text{Local Otsu},u}(x)|}{R}-1\right)\times\{T_{\text{Global Otsu},1-u}(x)-T_{\text{Local Otsu},u}(x)\}+T_{\text{Local Otsu},u}(x) \quad (1)$$

where Θ is the unit step function, and $T_{\text{Global Otsu},u}$ is Otsu's threshold [30]

$$T_{\text{Global Otsu},u}=\arg\max_{T}\left(\frac{\sigma_{bet}^2}{\sigma_{tot}^2}\right) \quad (2)$$

where u is the document image under study, and σ_{bet} and σ_{tot} are between-class and total variances respectively [30,4]. The local version of Otsu's threshold, $T_{\text{Local Otsu},u}(x)$, introduced in [10], applies the same formula as (2) to a local patch around pixel x. The parameter R in formula (1) is used to compensate the deviation of the local threshold from the global one.

The adaptive Otsu formula (1) was the first successful attempt to make Otsu's method adaptive. However, this method has some limitations. The main drawback of the method is the presence of the parameters R in the formula that push it far from Otsu's method toward other parameter-based methods such as Sauvola's method. A constant value, such as 0.1 can be used for R, but this will put an upper limit on the performance of the method. Also, learning of the parameters from the document image itself is a challenge in front of all adaptive methods that needs a thorough understanding of document images. The second limit of the method is the global Otsu threshold itself. Although the global Otsu threshold is used to stabilize the method and identify most probable background regions, it puts a limit on the performance of the method because the global threshold can be completely independent from the local behavior of text and background. These limits can be seen from the performance of the method which will be provided in Table 3 later in Experimental Results section. These limits are our main motivations toward a parameterless and generic adaptive Otsu method which will be discussed in Section 4. The difference of the proposed AdOtsu method and the method described in [10] will be discussed more in Section 4. It is worth noting that the other contributions of [10] are (i) introduction of a multiscale framework for adaptive binarization methods, (ii) introduction of the grid-based modeling, and (iii) introduction of recursive binarization methods in order to remove bleed-through interfering patterns.

2.12. Background estimation

The concept of background estimation has been used in many works [17,11,8,24]. For example, in [17], an approximate background is estimated using interpolation of the pixel values assigned to background according to a rough binarization on a patch of the size of two characters. The main difference with that method and ours, which will be discussed later in Section 6, is that we estimate the gray values of background and text using a rough binarization in a multiscale approach from a high scale of a factor of the stroke width (see Appendix A). In another work [27], an estimation of background is obtained using polynomial surface smoothing. It is worth noting that, in contrast to the other methods, it does not look for the accurate value of the background but rather an approximate of the average background.

In [19], a method similar to that of [17] has been used. In that work, they used window swell filter to recover disconnected weak strokes. It is worth noting that, in contrast, we will use multiscale approach to preserve weak strokes. The method is especially successful for high-intensity document images with degraded background.

Although the estimated background is an important element of many binarization methods, there is not an objective method to evaluate the performance of the background estimators to our best of knowledge. This is because of the complex nature of a background and non-uniqueness of its definition. Therefore, we evaluate only the binarization performance of the methods in Experimental Results section.

3. Problem statement

A degraded document image, u, is given, $u:\Omega\rightarrow\mathbb{R}$, where $\Omega\subset\mathbb{R}^2$ is an open rectangle. The value of u at pixel x is $u(x)$, where $x=(k,l)\in\Omega$. A clean and accurate binarization of u is required, which consists of a binary map on which the black (text) pixels have a value of 0 and the white pixels have a value of 1 (the BW01 representation [10]). The output is denoted u_{BW}. An estimation of the background of u is calculated and is denoted u_{EB}. In order to decrease the computational cost, grid-based modeling is used for almost all the functions. In this way, to compute a function on an image, the image under study is broken down into a set of local patches which have a high degree of overlap. Then, the function is calculated on each patch, and the resulting value is assigned to the corresponding node on the grid. The value of the function for each pixel on the image domain is then calculated using the interpolation of the surrounding nodes.

In the following section, this problem is addressed by introducing an adaptive version of Otsu's method which is based on an estimation of the background. The method is improved by generalizing it to the multiscale framework. The estimated background (EB) is calculated using a bootstrap approach which includes the binarization method.

4. AdOtsu method based on the estimated background

We call the adaptive versions of Otsu's method the AdOtsu methods. In this section, a new AdOtsu method providing superior performance is introduced, which is based on the estimated background (EB). In this way, the method is parameterless similar to Otsu's method itself, and therefore is less sensitive to variations on the input document image.

In the proposed AdOtsu method, we assign a threshold value to each pixel on the image domain. If the pixel value is less than that threshold, the pixel is considered a text pixel, otherwise it is labeled a background pixel. The following adaptive threshold value is used for each pixel x:

$$T_{\text{AdOtsu},u}(x)=\epsilon+\left(\arg\max_{T}\left(\frac{\sigma_{bet}^2(x)}{\sigma_{tot}^2(x)}\right)-\epsilon\right)\Theta(\sigma(x)-k_\sigma\sigma_{\text{EB}}(x)) \quad (3)$$

The first term in parentheses is Otsu's threshold calculated on a patch around x [10]. The small value ϵ is considered for the sake of numerical stability. The unit step function, denoted by Θ, controls the behavior of the method based on the variance of the input image and the estimated background (EB). The standard deviations of u and u_{EB}, denoted $\sigma(x)$ and $\sigma_{\text{EB}}(x)$ respectively are calculated on the same patch. Finally, k_σ is a factor between 1 and 2 used to add a hysteresis behavior in the switch. In brief, if the variation on the input image near x is less than a factor of that of the EB, ϵ will be used as the threshold and x will be assigned to the background. In contrast, if $\sigma(x)$ is high, Otsu's method is used, but on a local patch around x. In this way, the method benefits from the advantages of Otsu's method, while at the same time being completely adaptive. This is the main advantage of the proposed method compared

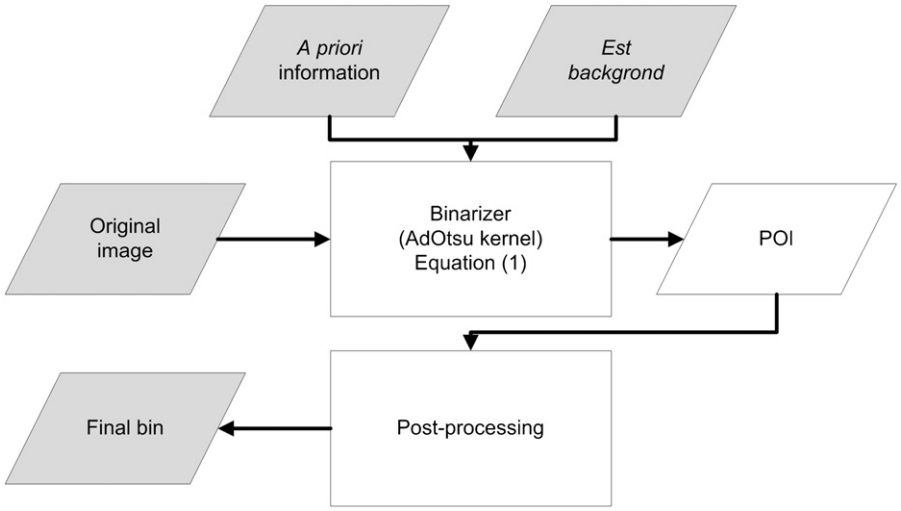

Fig. 1. The proposed AdOtsu binarization method using the estimated background (EB) to locate the non-text regions.

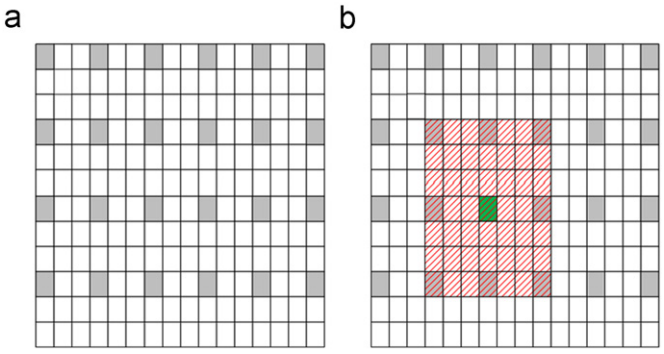

Fig. 2. (a) A typical grid network of nodes with $S_G = 3$. (b) The corresponding patch of the green node, shown in red, overlaps the other patches by a significant amount. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

to that in [10], which is restricted by its high dependency on the global threshold value. This can be seen from Table 3 in Experimental Results section, which shows that the grid-based Sauvola method performs better.

Fig. 1 shows the proposed AdOtsu method schematically. It is worth noting that the pixels of interest (POI) shown in the figure represent the result obtained using the threshold formula (3). Postprocessing techniques, discussed in Section 7, are used to correct the POI. The binary image obtained after the postprocessing step is the final binarization of the input image. Also, because of the computational cost associated with the adaptive methods, we use grid-based modeling [10] in the next subsection to reduce the computational complexity of our method.

4.1. The grid-based AdOtsu method

We adopt grid-based modeling for the sake of computational cost. In this approach, a grid G is assigned to the image, which consists of a set of nodes and associated overlapping patches [10] (see Fig. 2, for an example of grid patches). The nodes are distributed evenly over the image domain with a grid step distance of s_G. All functions and the threshold value are calculated for each node on the grid based on the data of its corresponding patch. Then, for each pixel on the image domain, the values are obtained by interpolating the threshold values of its neighboring nodes on the grid. The interpolated values are assigned a G subscript, which is also used to label the method. In terms of the grid-based notation, the threshold formula (3) can be rewritten as follows:

$$T_{\mathrm{G,AdOtsu},u}(x) = \epsilon + \left(\underset{T,\mathrm{on\ G}}{\arg\max} \left(\frac{\sigma^2_{bet}(x)}{\sigma^2_{tot}(x)} \right) - \epsilon \right) \Theta(\sigma_{\mathrm{G}} - k_\sigma \sigma_{\mathrm{G,EB}}) \tag{4}$$

5. The multiscale AdOtsu method

It has been proved that the multiscale approaches are capable of extracting the correct text pixels while avoiding degradation and interfering patterns [10,5,40]. In this section, the proposed AdOtsu method is generalized using the multiscale framework (see Algorithm 1 for the details). It is worth noting that w_s and h_l in the algorithm are the average stroke width and the average line height respectively. The average stroke width is defined as a constant value on each image (see Appendix A), and a method to estimate this characteristic length is presented in Appendix B. The average line height h_l is defined as the average value of distances between adjoining baselines of text lines (see Appendix A). This characteristic length is estimated using spectral analysis of a rough binarization of the document image. The details are provided in Appendix C. At each scale, the core pixels of the text on the binarized map of the previous scales are excluded from the calculations (step 5 in the algorithm), and the AdOtsu method is applied at that scale. Then, the result is combined with the result of previous scales, and that scale is divided by 2. The process continues until the lowest scale, which is a factor of w_s is reached. The exclusion of the core text pixels in the proposed multiscale framework makes it possible for the proposed method to reach very weak text regions by following the smooth gradient in the stroke intensity. The flow diagram of the multiscale AdOtsu method is provided in Fig. 3. The performance of the proposed method is evaluated in the experimental section. As is obvious from the discussions, the key element in methods is the estimated background, which is discussed in the next section.

Algorithm 1. The flow diagram of the multiscale model T_{mG}

1. Get the input image u, *a priori* information w_s^a and h_l^a, and the adaptive method T_G;
2. Compute s_{high} based on the T_G; Set $s = s_{high}$;
3. $n=0$; Compute the binarized map at s using T_G: $u_{BW,s_{high}}$; Set the multiscale binarized map u_{BW} to $u_{BW,s_{high}}$; Compute the binarized mask based on erosion of $u_{BW,s_{high}}$ with the

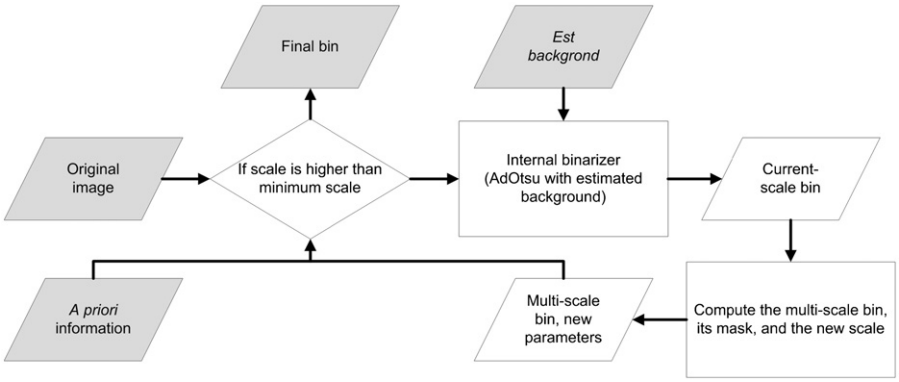

Fig. 3. The multiscale AdOtsu method.

amount of $s/4$; Relax the parameters of the adaptive method;
repeat
  $n=n+1$, Set s to $s/2$;
  Remove the core text pixels from u_{BW} by removing the skeleton pixels;
  Compute the binarized map at s using $T_G : u_{BW,s}$;
  Transfer the $u_{BW,s}$ on u_{BW} based on connectivity criteria and binarized mask;
until $s < s_{low}$;

[a] See the text and Appendix for the definition.

6. The estimated background (EB)

The estimated background (EB) concept has been used in many document image processing techniques [17,11,8]. Usually, the EB is used as an approximation along with other variables to calculate the probability or likelihood that a pixel belongs to the background or text. The concept of the "true" EB has been first introduced in [9] based on a combination of several multi-level classifiers. The main difference between these two concepts is that in the latter the actual characteristics of the background, such as variance, can be calculated from the EB. In this work, we use an iterative approach, in which a binarization method is used to capture the pixels-of-interest (POI) and calculate the EB. Before discussing the whole process of background estimation, the core process that calculates an estimation of background gray level based on a rough, *a priori* binary map is discussed in the next subsection.

6.1. Estimation of the background gray level (BGL) based on a rough binarization map

Assume that u is the input document image, and u'_{BW} is a rough binarization of that image. For each pixel on Ω, a background value and a text value are calculated using masked averaging on a patch around the pixel (see Algorithms 2 and 3 for details). We call the calculated maps the background gray level (BGL) and the stroke gray level (SGL) respectively. The patches have two different scales, depending on whether the target pixel belongs to the background or to the text based on the rough binarization u'_{BW}. For text pixels, a higher scale is used. Also, the patches are masked according to u'_{BW}, so that only the background pixels and text pixels participate in the averaging process for background value and text value respectively. Using the calculated values for text and background, the u'_{BW} is corrected [1], and then the process is repeated. It is possible that, at the end of the process, some pixels would not have an assigned background value. Using an inpainting process, the estimated background is extended to cover these pixels. Fig. 1 shows a schematic diagram of the process. In the following subsection, the complete process of background estimation, which uses the estimator in this subsection iteratively, will be discussed.

Algorithm 2. Estimation of a background gray level (BGL) based on a rough binarization map:

Get the input image and its rough binarization: u and u'_{BW};
Set the parameters: $s_{\text{high}} = 4w_s$, $s_{\text{low}} = 1$;
Mask u using u'_{BW} to keep just the background pixels;
Initialization of the BGL: Calculate the mean of the masked u as the stroke gray level of the scale of infinity;
Set the current scale s to s_{high};
while *The current scale is higher than* s_{low} **do**
  Compute the background gray level of the current scale;
    Calculate the averaged masked u on the current scale using the grid − based modeling [10];
    Correct the void spaces on the BGL of the current scale, using the values of the BGL;
  Correct the BGL on pixels that have higher (darker) value on the BGL of the current scale;
  $s = s/2$;
end
For pixels which are labeled as background in u'_{BW}, replace their BGL with their u;

Algorithm 3. Estimation of the SGL map based on a rough binarization map:

Get the input image and its rough binarization: u and u_{BW};
Set the parameters: $s_{\text{high}} = 4h_l$, $s_{\text{low}} = 1.5w_s$;
Mask u using u_{BW} to keep just the text pixels;
Calculate the mean of the masked u as the stroke gray level of the scale of infinity, and use it as the initialization of the SGL;
Set the current scale s to s_{high};
while The current scale is higher than s_{low} **do**
  Compute the stroke gray level of the current scale;
    Calculatetheaveragedmaskeduonthecurrentscale usingthegrid − basedmodeling [10];
    Correct the void spaces on the stroke gray level of the current scale, using the values of the SGL;
  Correct the SGL on pixels that have higher (darker) value on the SGL of the current scale;
  $s = s/2$;
end

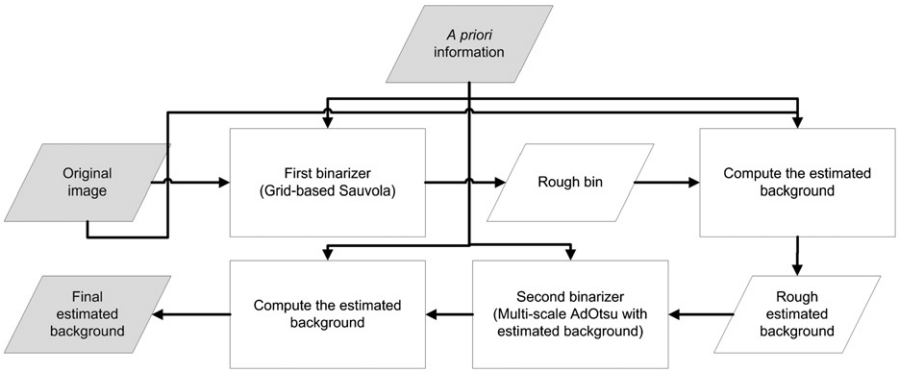

Fig. 4. The process for calculating the estimated background (EB).

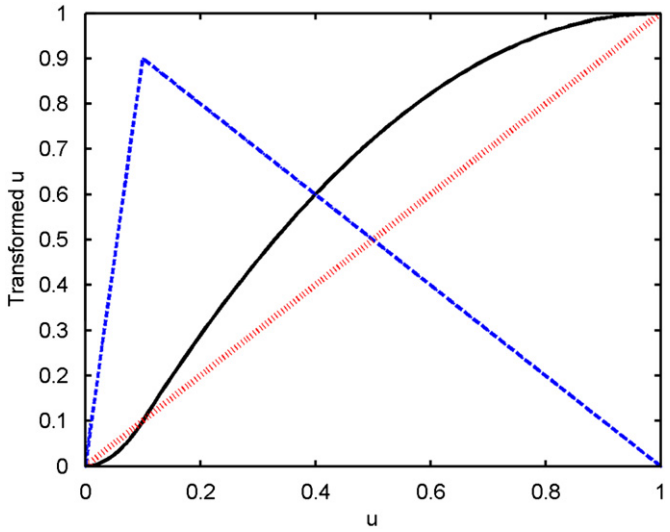

Fig. 5. A typical plot of Eq. (5) for constant value $T=0.1$ is shown in black. The piecewise linear distribution used for this transform and the linear distribution of the unitary transform are also shown in blue and red respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6.2. Estimation of the background

Using the process to calculate the BGL in the previous subsection, we use a bootstrap strategy to converge to an accurate EB. The whole process is shown in Fig. 4. The process consists of binarization performed twice and calculation of the estimated background. For the initialization, a parameterless double-scale binarization method, which is based on the grid-based Sauvola method [10] and *a priori* information, is used. The two scales are selected based on the *a priori* information, the line height. This binarization is applied to a transformed version of the input image in which the gray values of pixels are normalized after being subject to transform according the following equation:

$$u=\begin{cases}(1-T)u^2, & \text{For pixels with } u\le T\\ (1-T)T^2+T((1-T)^2-(1-u)^2), & \text{For pixels with } u>T\end{cases} \quad (5)$$

where, in this equation, u is the gray value of a pixel and T is its local Otsu threshold value [10]. The transform tries to push the data far from the overlapping area between text and background in a smooth way. A typical plot of the transforming equation is presented in Fig. 5. Also, the original distribution used to build the transform and the unitary transform are shown for the sake of comparison.

After inserting this process of the EB calculation into the multiscale AdOtsu method (shown in Fig. 3), the complete process of the binarization can be visualized (see Fig. 6). A typical example of how the proposed multiscale AdOtsu method works is presented in Fig. 7. The original image from the DIBCO'09 dataset is shown in Fig. 7(a). The rough binarization obtained using the grid-based Sauvola method [10] is shown in Fig. 7(b). As can be seen from the figure, there are many discontinuities on the regions of the weak ink. In the first run of background estimation, missed strokes appear on the BGL (shown in circles in Fig. 7(c)). Based on the BGL, an internal binarization is obtained using the multiscale AdOtsu method (see Fig. 7(d)). Thanks to recovery of the weak strokes in the multiscale framework, the second BGL, which is the EB, does not contain any part of the text. The EB is shown in Fig. 7(e), and a recovered stroke is circled on that image. The final binarization obtained using the multiscale AdOtsu method and the EB are shown in Fig. 7(f). The continuity of the strokes is highly preserved on the final image. The full images are available on Internet.[1]

7. Postprocessing

In the postprocessing step, the artifacts of the POI are removed. This consists of two subsequent sub-steps (see Fig. 8). In the first sub-step, the blobs or connected components (CCs) of the POI are clustered in two classes based on a feature vector, which is composed of the average color values of the blob and the area ratio of the blob with respect to the whole image size. Also, the background region of the binarized image is considered as a blob. All blobs of the same class as the background region are dropped from the POI. Difference of the average color of each blob to the color SGL, its color variation, the average direction of its edges, and its normalized area are considered as features. The color SGL is calculated similar to the SGL except in line 10 in Algorithm 3 gray version of the color SGL is used to make the decision.

In the second sub-step, sub-strokes on the POI are analyzed in a similar way. For this purpose, the blobs or CCs are first broken down into their sub-strokes. This is achieved using the corrected skeleton of the blobs. The corrected skeleton is obtained by removing the bridges and short branches from the skeleton. A bridge is an artificial connection on the output image which appears during the binarization process. The correction process works based on w_s. Fig. 9 shows a typical example. The skeletons are obtained using the thinning technique, the corrected skeletons are broken down into their branches (Fig. 9(b)), and then the

[1] 〈http://www.synchromedia.ca/web/reza/expres/adotsueb10〉

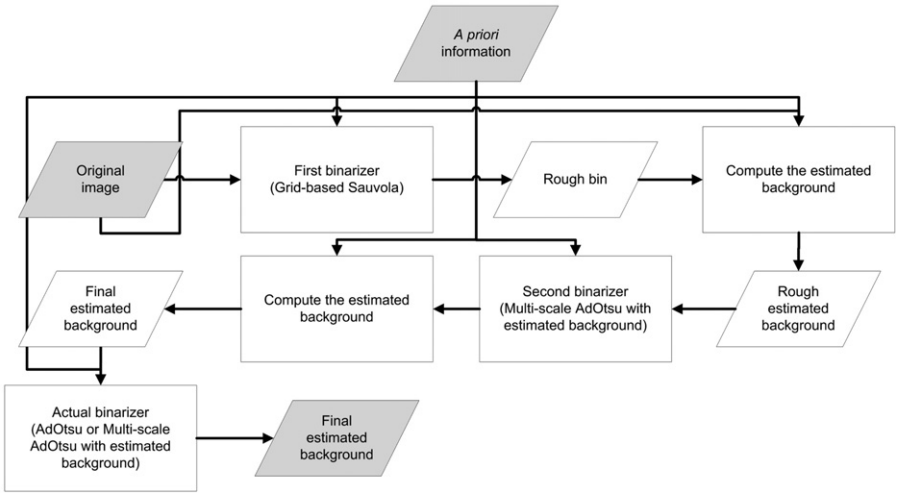

Fig. 6. The total flow diagram of the multiscale AdOtsu method including the steps for estimating the background.

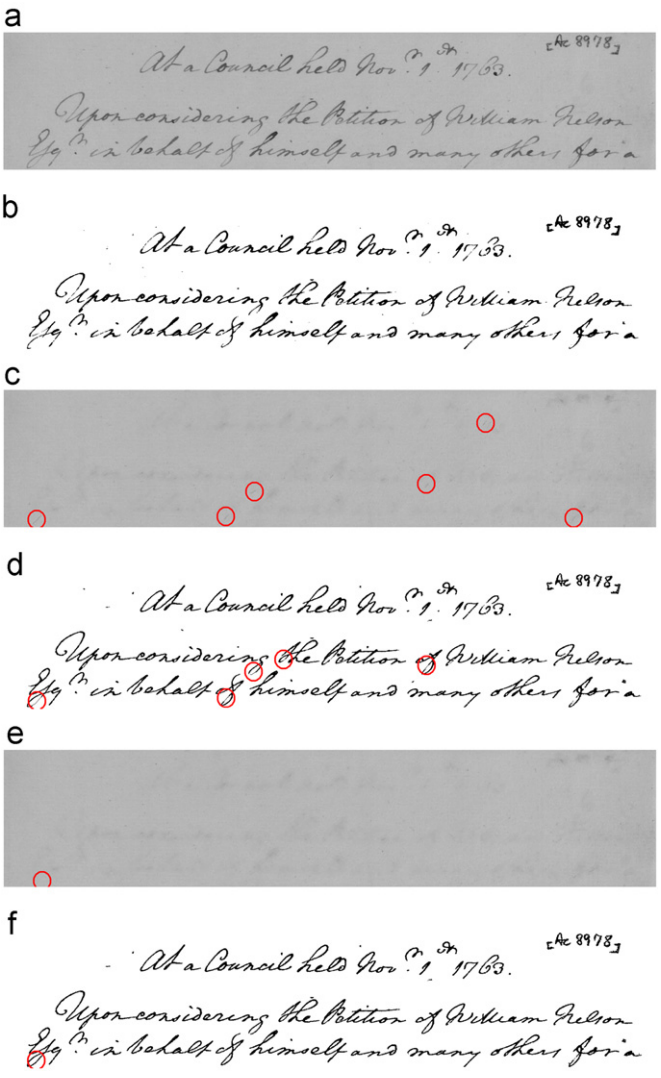

Fig. 7. An example of the internal steps of the proposed multiscale method. (a) The original image. (b) The rough binarization map, u'_{BW}. (c) The first BGL. (d) The internal binarization result. (e) The EB. (f) The final result of multiscale AdOtsu method using the EB.

blobs are partitioned based on those branches (Fig. 9(c)). Each partition represents a sub-stroke. Similar to the first sub-step, a feature vector is assigned to each sub-stroke, without considering its normalized area this time. Finally, the sub-strokes are divided into two classes. The class with the darkest average gray value is kept, and the blobs of the other class are deleted from the POI. Again similar to the first sub-step, the background region is also considered in the clustering process. Fig. 9(d) shows the final output. It is worth noting that the feature vector used for classification is not ideal, and this can be seen from remaining some interfering patterns on the output. The author is working on improvement of the feature vector for a better performance. It is worth noting that, in the case of severe bleed-through, the background can be excluded from the clustering. In this case, the bleed-through signatures and interfering patterns will compose the second class. The presence of the bleed-through signatures is automatically analyzed using a method presented in Appendix.

8. Experimental results

In this section, the proposed methods are evaluated both subjectively and objectively. Several datasets have been used including: (i) the Google book search database [18]; (ii) manuscripts courtesy of the Juma Al Majid Center for Cultural Heritage (Dubai);[2] and (iii) the DIBCO'09 dataset [13]. The authors acknowledge that the DIBCO'09 participants did not have access to the dataset, and therefore were not able to adapt the parameters of their methods. The datasets comprise images in different scripts and languages suffering from various types of degradation. Below, the subjective and objective evaluations are presented. In all experiments, $k_\sigma = 1.6$.

8.1. Subjective evaluation

Two samples from the datasets are shown in Fig. 10. The input images are shown in the left-hand column, and the outputs of the proposed multiscale grid-based AdOtsu method are shown in the right-hand column. The results are promising. The method is able to extract the text while the image is suffering from the degraded background or has large, empty background regions. For the sake

[2] http://www.almajidcenter.org/English/Pages/default.aspx

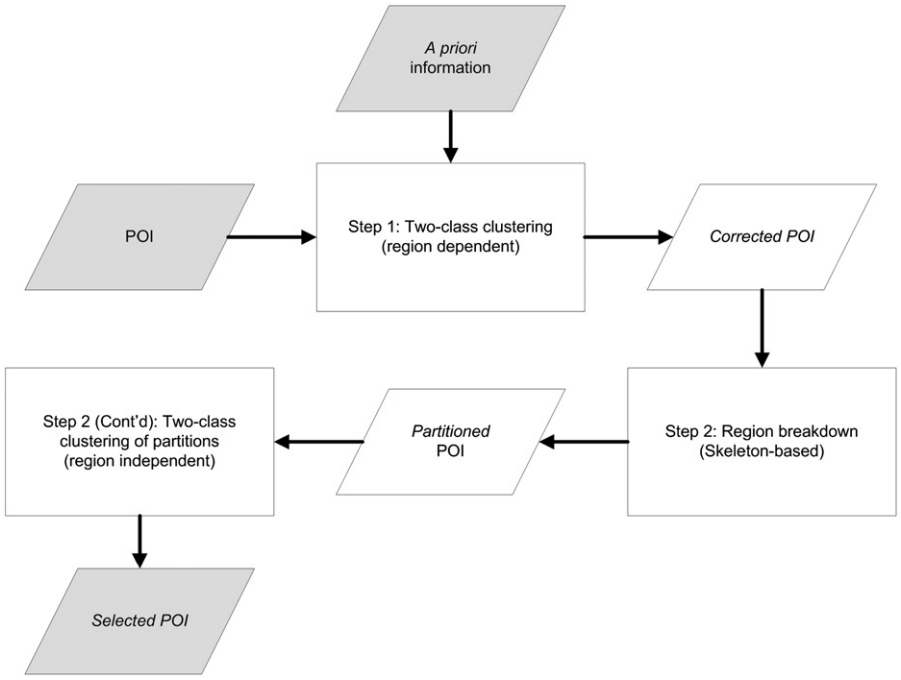

Fig. 8. The postprocessing flow diagram.

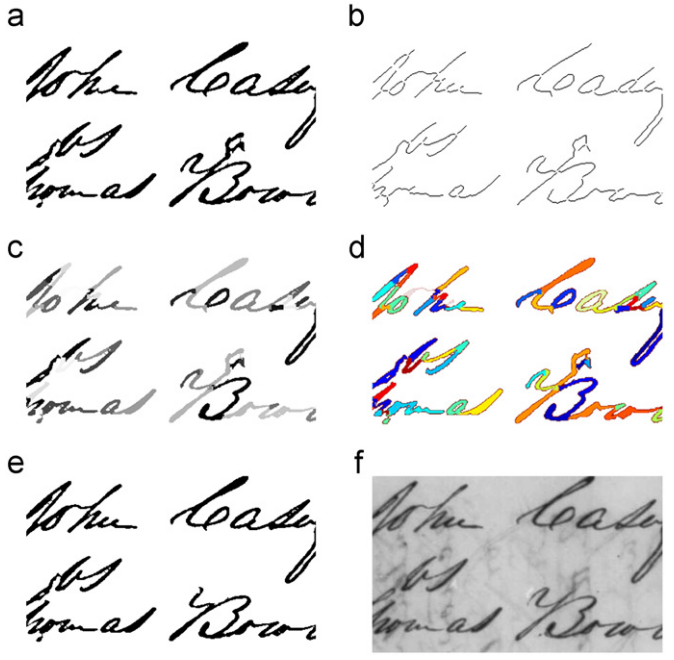

Fig. 9. An example of skeleton-based postprocessing on a part of image H03 from DIBCO'09 dataset. (a) The POI. (b) The skeleton image with its branches broken down. (c) The sub-strokes obtained using (b). (d) The same as (c) but in color. (e) The final result after clustering. (f) The original image. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

of comparison, the available results of the other methods are presented in Fig. 11. It is worth noting that Fig. 11(c) is obtained using another adaptive modification of Otsu's method presented in [10]. As discussed in Introduction, that method is obtained by combining the global and local thresholds calculated using Otsu's method. The limited performance of this combination can be seen

Fig. 10. The performance of the proposed multiscale grid-based AdOtsu method for two historical document images. (a) and (c) the input images from dataset (ii) and dataset (iii) respectively. (b) and (d) the outputs of the proposed method.

Fig. 11. The performance of various methods.The output of: (a) the PDE-based enhancement method [9], (b) the multiscale grid-based Sauvola method [10], (c) the multiscale grid-based adaptive Otsu method [10], (d) the proposed multiscale grid-based AdOtsu, repeated from Fig. 10(b).

Table 1
The performance of the proposed multiscale AdOtsu method against the DIBCO'09 dataset when the *a priori* parameters h_l and w_s are provided by a human expert. The values of these parameters are also shown in the table.

Image name	w_s	h_l	*P*	*R*	*F*-measure
H01	3	140	0.95539	0.92593	94.04268
H02	5	60	0.94518	0.88432	91.37345
H03	5	140	0.89509	0.93364	91.39586
H04	7	80	0.89360	0.88309	88.83168
H05	7	100	0.86648	0.90506	88.53497
P01	5	60	0.93375	0.90976	92.15973
P02	7	70	0.96903	0.95551	96.22192
P03	15	100	0.99256	0.94192	96.65742
P04	7	60	0.94895	0.90124	92.44783
P05	5	60	0.95745	0.83108	88.98042
Average	–	–	**0.93575**	**0.90715**	**92.06460**

from the figure for images suffering from a degraded background. Thanks to the EB, the proposed method provides an accurate and clean binarization of the input image.

8.2. Objective evaluation

Dataset (iii) contains the ground truth images that enable us to perform the objective evaluation. Table 1 provides the performance of the proposed multiscale grid-based AdOtsu method, in terms of Precision, Recall, and *F*-measure [43]. The average stroke width w_s and the average line height h_l are the two main parameters making up the *a priori* information [10], and are provided by a human expert. The values are also reported in the table.

The same experiment is repeated, but this time using the estimated values of these parameters. The methods used for estimation are provided in Appendix. The results, along with the estimated values of the average line height, are reported in Table 2. We call this mode of operation the *fully automated* mode. Despite a small decrease in the performance of the method, the results are promising. The major advantage of this mode is its parameterless nature. There is a big difference between the estimated value and the value assigned by the human expert in the case of P03, and this is because of the presence of two different texts on the image. It shows that there is a need to adapt the parameter values to the variations on the image domain. We will consider this requirement in future work.

Finally, in Table 3, the proposed methods are compared to results of state-of-the-art methods on the DIBCO'09 database. The first three methods are the standard Sauvola, Otsu, and Gatos methods. As expected, the Gatos method outperforms the other two, and the Recall performance of Otsu's method is the highest. The second set of methods is made up of modified versions of the Sauvola and Otsu methods published previously [10,39]. These methods achieve higher performance, because of the use of optimization, learning techniques, and multiscale analysis. All these methods have two or more parameters to set, which makes optimization more difficult. As seen from the table, the performance of the adaptive Otsu method from [10] is good, but limited because of its dependence on the global Otsu threshold, and also because of an unoptimized parameter. The next set consists of the three top methods in the DIBCO'09 contest. These methods use various approaches, from edge detection to the

Table 2
The performance of the proposed multiscale AdOtsu method in all automatic mode. The parameters h_l and w_s are determined using the procedures described in Appendix.

Image name	H01	H02	H03	H04	H05	
The estimated h_l	124.00	63.78	167.67	95.25	104.20	
P	0.95782	0.93694	0.88199	0.89923	0.88443	
R	0.92553	0.89290	0.95182	0.87836	0.80803	
F-measure	94.13973	91.43925	91.55734	88.86713	84.45075	
Image name	P01	P02	P03	P04	P05	Average
The estimated h_l	62.20	62.00	236.50	61.25	62.80	–
P	0.92716	0.96751	0.99256	0.91837	0.95659	**0.93226**
R	0.91112	0.95721	0.94192	0.90318	0.83902	**0.90091**
F-measure	91.90714	96.23331	96.65742	91.07109	89.39511	**91.57183**

Table 3
Comparison of the performances of various methods against the DIBCO'09 dataset.

Set	Method name	*P*	*R*	*F*-measure	Time (a.u.)
1	Sauvola's [34][a]	–	–	85.41	124.0
	Otsu's [30]	0.72956	0.94555	78.39	0.05
	Gatos' [15][a]	–	–	85.25	133.1
2	Optimized Sauvola method[b]	0.88253	0.87481	87.27	204.9
	Sauvola's method in a self-training framework [39]	–	–	89.12	–
	The multiscale grid-based Sauvola [10]	0.85828	0.93808	89.26	13.1
	The multiscale grid-based adaptive Otsu [10]	0.85433	0.86628	85.77	12.4
3	Lu and Tan algorithm [26,27,13]	–	–	91.24	–
	Fabrizio and Marcotegui algorithm [7,13]	–	–	90.06	–
	Rivest-Hénault, Farrahi Moghaddam, and Cheriet algorithm [33,13]	–	–	89.34	–
4	Su, Lu and Tan algorithm [38][a]	–	–	91.06	–
	Perret, Lefèvre, Collet and Slezak algorithm [32]	–	–	91.24	–
5	The proposed grid-based AdOtsu	0.94522	0.89739	**92.01**	40.86
	Proposed multiscale grid-based AdOtsu; see Table 1	0.93575	0.90715	**92.06**	55.8
	ut supra, in all automatic mode; see Table 2	0.93226	0.90091	**91.57**	58.86

[a] Using the recommended parameters [26].
[b] The performance of Sauvola's method is taken from [10].

level set framework. (The dataset was not available to the authors of these methods prior to the contest.) The fourth set considers two other state-of-the-art methods published after the contest. Both methods achieve a high score on the dataset. The first method uses local maxima and minima to identify edge pixels, and the second is based on a hypercomponent tree. Brief descriptions of these state-of-the-art methods are provided in Section 2. Finally, the results of three proposed methods are presented: (i) the grid-based AdOtsu method; (ii) the multiscale grid-based AdOtsu method; and (iii) the fully automated version of the latter. It is worth noting that the high performance of the grid-based AdOtsu method, which is not explicitly multiscale, can be explained by the fact that it uses an accurate EB, and, in calculating that EB, we use the multiscale version of the method. Therefore, it implicitly uses the multiscale framework. The reason for the lower Recall value of the proposed method compared to that of the multiscale Sauvola method is the imperfection of the postprocessing step, especially in its clustering process. The authors believe that a better feature vector would help increase the Recall value. They are currently working on this problem, and will present their results in future work.

As can be seen from the table, all the methods converge to an imaginary upper bound, which is characteristic of the state-of-the-art methods. In other words, they reach the capacity limit of their modeling category, which is the adaptive binarization category in our case. In contrast, the lack of a performance upper bound would be a sign of the incompleteness of the dataset used for evaluation [22]. Being within the marginal border of the performance upper bound, each method may benefit from its advantages, one of them for our proposed method being fewer free parameters. A chart is presented in Table 4, in which several factors, such as number of free parameters, complexity, and adaptivity, are compared for clarification. Seven parameters are considered for the Gatos method: three for its Sauvola component, one for background estimation, and three for the threshold formula [15]. The complexity of the proposed method is lower by a factor of s^2, thanks to grid modeling [10]. The multiscale nature of the method adds a factor of $\log_2(h_l)$ to this complexity. The factor 5 in the complexity of the

Table 4
Comparison of the characteristics of various methods. n and n_g are the image size and number of gray levels respectively.

Method name	# Free parameters	Complexity	Complexity (a.u.)[a]	Locally adaptive
Otsu's [30][b]	0	$O(n^2+7n_g^2)$	72.08	×
Sauvola's [34][b]	3	$O(s^2n^2)$	21 233.66	✓
Gatos' [15]	7	$O(s^2n^2+s_b^2n^2)$	42 467.32	✓
Proposed grid-based AdOtsu	0[c]	$O(5\log_2(h_l)n^2)$	8708.29	✓

[a] Where $n=512$, $n_g=256$, $h_l=100$, $s=10$ and $s_b=10$, where $2s+1$ and $2s_b+1$ are the window sizes used to calculate the mean and the estimated background respectively.
[b] Taken from [31].
[c] If we ignore the grid scale, which is set automatically to $2h_l$.

proposed method represents multiple calculations, including background estimation. As can be seen from the table, the proposed method not only has the advantage of Otsu's method in terms of fewer parameters, but it is capable of adapting to the input image while its complexity is within the range of that of the adaptive methods. Achieving higher performances is possible by considering higher levels of modeling; for example, by combining different paradigms [16] or considering OCR output or language-level feedback to build high-level models. This, however, is outside the scope of this work.

9. Conclusion and future prospects

A novel adaptive binarization method inspired by Otsu's method is introduced. The method, called AdOtsu, uses the estimated background (EB) as *a priori* information to differentiate between text and non-text regions. The estimated background values are calculated in a bootstrap process implicitly incorporating the proposed binarization method. Also, *a priori* structural information, including the average stroke width and the average text height, is used to adapt the method on the input document image, and to make it parameterless. As was done in the previous work, the method is generalized to a multiscale binarization, which enables it to separate interfering patterns from the true text using higher scales. Postprocessing corrections, both topological and clustering, are considered to improve the final output. The method is applied to several datasets, including DIBCO'09, with promising results.

As a possible venue for future study, application of the Modified Markovian Clustering (MMCL) method [21] to improve the classification of the POI blobs into text and background/degradation blobs in the postprocessing step will be considered. The MMCL method is a modified version of Markovian Clustering (MCL) [42], and is especially designed for fast classification of objects into two classes. The MMCL method has successfully been used for selection of candidate pixels in the non-local means (NLM) denoising of images [21]. Also, parameters such as average stroke width will be made adaptive to the local variation of text on the document image. Applying the same concept to estimate the parameters of Sauvola's method is another direction for future work.

Acknowledgments

The authors thank the NSERC of Canada for their financial support.

Appendix A. Characteristic lengths [10]

Characteristic lengths are defined based on the range of interactions on the document images. They can be extracted using various tools, such as wavelet transform or kernel-based analysis. Although these parameters may vary drastically even on a single image from one site (paragraph) to another, their behavior is usually very robust and almost constant over a whole dataset. Therefore, many learning and data mining methods can also be used to obtain robust values, or at least robust candidate values, for the characteristic lengths. We assume that the values of these parameters are known *a priori* and are constant on each image. Below, a few characteristic lengths are defined that will be used throughout this work.

- *Stroke width*: The most obvious characteristic length on a document image is the *stroke width*. The stroke width is an important parameter for distinguishing true text from possible high intensity degradation. It is worth noting that because of fixed imaging resolution at the beginning of the digital era (mostly 150 DPI), this parameter was actually hidden within many enhancement and binarization methods. However, using high and variable resolutions in imaging systems, especially in camera-based systems, the stroke width is no longer a constant and must be considered explicitly in modeling. Also, in the case of historical documents, and even in some forms of modern documents, such as newspapers and magazines, several different stroke widths can present even on a single page. In this work, we use the *average stroke width*, w_s, as a constant, *a priori* parameter, which is estimated using a kernel-based algorithm (see Appendix B). The stroke width is schematically shown in Fig. A1(a).
- *Line height*: The second most important characteristic length on document images is the shortest distance between text sites, which are usually text lines. We call this parameter the *line height*. By definition, line height is the distance between two adjoining baselines. Again, in this work, only the *average line height*, h_l, is used. Fig. A1(b) shows the basic definition of the average line height as the distance between two adjoining baselines. In Appendix C, a method to estimate the average line height is presented.

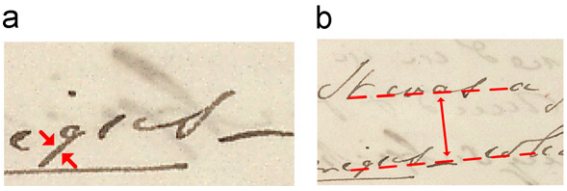

Fig. A1. Typical average stroke width (a) and average line height (b) on actual text samples.

In the subsequent sections, our methods used to estimate these parameters are presented.

Appendix B. Estimation of the average stroke width

In order to estimate the average stroke width, we first compute the *stroke width spectrum* (SWS), which is a measure of the frequency of possible stroke width values on the document image. In order to determine the value of this spectrum, a kernel-based approach on a binarized version of the image is used for each stroke width candidate $\overline{w}_s$. For each pixel on the image domain, if the ratio of text pixels in a patch of size $\overline{w}_s$ around that pixel is higher than a threshold (say 0.9) that pixel is counted as a stroke pixel. In order to reduce the computational cost, integral image representation [36,2] is used to obtain the SWS, a sample of which is shown in Fig. B1 as blue bars. Usually, $\overline{w}_s$ is tested from 3 to 21 pixels. We use a model-based approach. In this model, we assume that the ideal input images contain strips of a constant width w. For these images, it is easy to show that SWS is equal to $1/w$ for $\overline{w}_s = w$. This model is shown in Fig. B1 as dashed line. By dividing the actual SWS of a document by this model, a characteristic curve for the document is obtained which reaches its highest value at the average stroke width. The curve is shown in the same figure as a continuous line and is normalized to one. The average stroke width of 11 can be estimated from the figure.

Appendix C. Estimation of the average line height

Using a rough binarization of the input image, the average line height h_l is calculated by obtaining the first peak frequency of the vertical distribution of the black pixels. Several enhancement

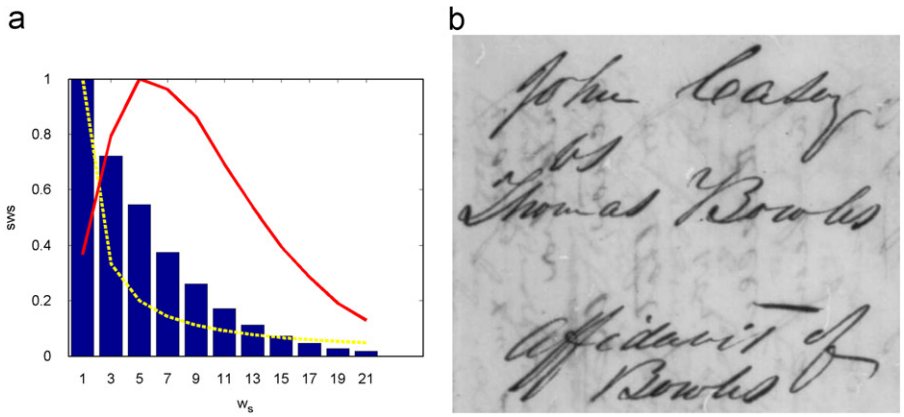

Fig. B1. (a) An example of the stroke width spectrum. (b) The input image. The grid-based Sauvola method is used to obtain a rough binarization. As seen from the figure, the maximum is at $\overline{w}_s = 5$, and therefore $w_s = 5$ is considered the average stroke width.

steps are used to improve accuracy (see Algorithm 4 for the details).

Algorithm 4. Estimation of h_l based on a rough binarization map:

1. Get the input image u and an initial value for h_l, $h_{l,ini}$;
2. Get a rough binarization map of u in BW10 protocol;
3. Apply one-dimensional smoothing to the binarized map along the vertical lines of pixels;
4. Calculate the distribution of the pixel values along the vertical axis;
5. Remove the small peaks which are less than half the value of the max peak in the distribution;
6. Substitute the silent interval at the end of the distribution with an interval of the length $h_{l,ini}$;
7. Calculate the FFT transformation of the distribution;
8. Find the peak frequency in the FFT result;
9. Calculate the h_l as the associated interval to that peak frequency;

Appendix D. Presence of the bleed-through degradation

The bleed-through degradation is an important but serve problem for document images. The model parameters usually vary drastically whether this degradation is presented or not. Here, a method to identify the presence of bleed-through is developed. First, recursive Otsu method [4] is used to isolate the text and possible bleed-through pixels. Then, if the distance between the peaks of these data is larger than a threshold value, the document image is labeled as it suffers from the bleed-through degradation.

References

[1] J. Bernsen, Dynamic thresholding of grey-level image, in: Eighth International Conference on Pattern Recognition, 1986.
[2] D. Bradley, G. Roth, Adaptive thresholding using the integral image, Journal of Graphics, GPU, and Game Tools 12 (2) (2007) 13–21.
[3] Y. Chen, G. Leedham, Decompose algorithm for thresholding degraded historical document images, IEE Proceedings Vision, Image and Signal Processing 152 (6) (2005) 702–714.
[4] M. Cheriet, J.N. Said, C.Y. Suen, A recursive thresholding technique for image segmentation, IEEE Transactions on Image Processing 7 (6) (1998) 918–921.
[5] M. Cheriet, Extraction of handwritten data from noisy gray-level images using a multiscale approach, International Journal of Pattern Recognition and Artificial Intelligence 13 (5) (1999) 665–684.
[6] H.-S. Don, A noise attribute thresholding method for document image binarization, International Journal on Document Analysis and Recognition 4 (2) (2001) 131–138.
[7] J. Fabrizio, B. Marcotegui, M. Cord, Text segmentation in natural scenes using toggle-mapping, in: ICIP'09, 2009, pp. 2373–2376.
[8] R. Farrahi Moghaddam, M. Cheriet, EFDM: restoration of single-sided low-quality document images, in: ICFHR'08, Montreal, Quebec, Canada, August 19–21, 2008, pp. 204–209.
[9] R. Farrahi Moghaddam, M. Cheriet, RSLDI: restoration of single-sided low-quality document images, Pattern Recognition 42 (12) (2009) 3355–3364.
[10] R. Farrahi Moghaddam, M. Cheriet, A multi-scale framework for adaptive binarization of degraded document images, Pattern Recognition 43 (6) (2010) 2186–2198.
[11] R. Farrahi Moghaddam, M. Cheriet, A variational approach to degraded document enhancement, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (8) (2010) 1347–1361.
[12] M.-L. Feng, Y.-P. Tan, Contrast adaptive binarization of low quality document images, IEICE Electronics Express 1 (16) (2004) 501–506.
[13] B. Gatos, K. Ntirogiannis, I. Pratikakis, ICDAR 2009 document image binarization contest (DIBCO 2009), in: ICDAR'09, 2009, pp. 1375–1382.
[14] B. Gatos, K. Ntirogiannis, I. Pratikakis, Dibco 2009: document image binarization contest, International Journal on Document Analysis and Recognition (2010) 1–10.
[15] B. Gatos, I. Pratikakis, S.J. Perantonis, Adaptive degraded document image binarization, Pattern Recognition 39 (3) (2006) 317–327.
[16] B. Gatos, I. Pratikakis, S.J. Perantonis, Improved document image binarization by using a combination of multiple binarization techniques and adapted edge information, in: ICPR'08, 2008, pp. 1–4.
[17] B. Gatos, I. Pratikakis, S.J. Perantonis, An adaptive binarization technique for low quality historical documents. in: Lecture Notes in Computer Science: Document Analysis Systems VI (DAS'04), vol. 3163, Springer, 2004, pp. 102–113.
[18] Google, Book Search Dataset, Version V edition, 2007.
[19] Y.S. Halabi, Z. SA"SA, F. Hamdan, K.H. Yousef, Modeling adaptive degraded document image binarization and optical character system, European Journal of Scientific Research 28 (2009) 14–32.
[20] R. Hedjam, R. Farrahi Moghaddam, M. Cheriet, A spatially adaptive statistical method for the binarization of historical manuscripts and degraded document images, Pattern Recognition 44 (9) (2011) 2184–2196.
[21] R. Hedjam, R. Farrahi Moghaddam, M. Cheriet, Markovian clustering for the non-local means image denoising, in: ICIP'09, Cairo, Egypt, November 7–10, 2009, pp. 3877–3880.
[22] R.C. Holte, Very simple classification rules perform well on most commonly used datasets, Machine Learning 11 (1) (1993) 63–90.
[23] S.-W. Lee, Y.J. Kim, Direct extraction of topographic features for gray scale character recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (7) (1995) 724–728. doi:10.1109/34.391416.
[24] M. Lettner, R. Sablatnig, Higher order mrf for foreground–background separation in multi-spectral images of historical manuscripts, in: DAS'10, ACM, Boston, MA, 2010, pp. 317–324.
[25] R.D. Lins, J.M.M. da Silva, F.M.J. Martins, Detailing a quantitative method for assessing algorithms to remove back-to-front interference in documents, Journal of Universal Computer Science 14 (2) (2008) 266–283.
[26] S. Lu, B. Su, C. Tan, Document image binarization using background estimation and stroke edges, International Journal on Document Analysis and Recognition 13 (4) (2010) 303–314.
[27] S.J. Lu, C.L. Tan, Binarization of badly illuminated document images through shading estimation and compensation, in: C.L. Tan (Ed.), Ninth International Conference on Document Analysis and Recognition 2007, ICDAR 2007, vol. 1, 2007, pp. 312–316.
[28] R. Milewski, V. Govindaraju, Binarization and cleanup of handwritten text from carbon copy medical form images, Pattern Recognition 41 (4) (2008) 1308–1315.
[29] W. Niblack, An Introduction to Digital Image Processing, Strandberg Publishing Company, Birkeroed, Denmark, Denmark, 1985.
[30] N. Otsu, A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man and Cybernetics 9 (1979) 62–66.

[31] Y.-T. Pai, Y.-F. Chang, S.-J. Ruan, Adaptive thresholding algorithm: efficient computation technique based on intelligent block detection for degraded document images, Pattern Recognition 43 (9) (2010) 3177–3187.
[32] B. Perret, S. Lefèvre, C. Collet, E. Slezak, From hyperconnections to hypercomponent tree: application to document image binarization, in: WADGMM'10, Istanbul, Turkey, August 22, 2010.
[33] D. Rivest-Hénault, R. Farrahi Moghaddam, M. Cheriet, A local linear level set method for the binarization of degraded historical document images, International Journal on Document Analysis and Recognition, Online First. doi:10.1007/s10032-011-0157-5, 2011.
[34] J. Sauvola, M. Pietikainen, Adaptive document image binarization, Pattern Recognition 33 (2) (2000) 225–236.
[35] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic Imaging 13 (1) (2004) 146–168.
[36] F. Shafait, D. Keysers, T.M. Breuel, Efficient implementation of local adaptive thresholding techniques using integral images, in: Document Recognition and Retrieval XV, San Jose, CA, January 2008.
[37] M. Shokri, H.R. Tizhoosh, Q(λ)-based image thresholding, in: CVR'04, 2004, pp. 504–508.
[38] B. Su, S. Lu, C.L. Tan, Binarization of historical document images using the local maximum and minimum, in: DAS'10, Boston, Massachusetts, June 9–11, 2010, pp. 159–166.
[39] B. Su, S. Lu, C.L. Tan, A self-training learning document binarization framework, in: ICPR'10, 2010, pp. 3187–3190.
[40] S. Tabbone, L. Wendling, Multi-scale binarization of images, Pattern Recognition Letters 24 (1–3) (2003) 403–411.
[41] O.D. Trier, A.K. Jain, Goal-directed evaluation of binarization methods, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (12) (1995) 1191–1201.
[42] S. van Dongen, Graph clustering by flow simulation, Ph.D. Thesis, University of Utrecht, May 2000.
[43] C.J. van Rijsbergen, Information Retrieval, 2nd ed., Butterworths, London, 1979.
[44] X. Wang, X. Ding, C. Liu, Gray-scale-image-based character recognition algorithm for low-quality and low-resolution images. in: Proceedings of SPIE, vol. 4307, SPIE, San Jose, CA, USA, 2000, pp. 315–322.
[45] X. Ye, M. Cheriet, C.Y. Suen, Stroke-model-based character extraction from gray-level document images, IEEE Transactions on Image Processing 10 (8) (2001) 1152–1161.

Reza Farrahi Moghaddam received his B.Sc. degree in Electrical Engineering and his Ph.D. degree in Physics from the Shahid Bahonar University of Kerman, Iran, in 1995 and 2003, respectively. Since 2003, he has been with Vali-Asr University of Rafsandjan, Iran. Since 2007, he has been a Postdoctoral Fellow with the Synchromedia Laboratory for Multimedia Communication in Telepresence, École de Technologie Supérieure (University of Quebec), Montreal, QC, Canada. Dr. Farrahi has published more than 45 technical papers. His research interests include mathematical modeling for image processing and document image processing and understanding.

Mohamed Cheriet was born in Algiers (Algeria) in 1960. He received his B.Eng. from USTHB University (Algiers) in 1984 and his M.Sc. and Ph.D. degrees in Computer Science from the University of Pierre et Marie Curie (Paris VI) in 1985 and 1988 respectively. Since 1992, he has been a professor in the Automation Engineering department at the École de Technologie Supérieure (University of Quebec), Montreal, and was appointed full professor there in 1998. He co-founded the Laboratory for Imagery, Vision and Artificial Intelligence (LIVIA) at the University of Quebec, and was its director from 2000 to 2006. He also founded the SYNCHROMEDIA Consortium (Multimedia Communication in Telepresence) there, and has been its director since 1998. His interests include document image analysis, OCR, mathematical models for image processing, pattern classification models and learning algorithms, as well as perception in computer vision. Dr. Cheriet has published more than 250 technical papers in the field, and has served as chair or co-chair of the following international conferences: VI'1998, VI'2000, IWFHR'2002, and ICFHR'2008. He currently serves on the editorial board and is associate editor of several international journals: IJPRAI, IJDAR, and Pattern Recognition. He co-authored a book entitled, "Character Recognition Systems: A guide for Students and Practitioners," John Wiley and Sons, Spring 2007. Dr. Cheriet is a senior member of the IEEE and the chapter chair of IEEE Montreal Computational Intelligent Systems (CIS).