Quality evaluation of degraded document images for binarization result prediction

Vincent Rabeux, Nicholas Journet, Anne Vialard, Jean-Philippe Domenger

Noname manuscript No.
(will be inserted by the editor)

Quality evaluation of degraded document images for binarization result prediction

V. Rabeux · N. Journet · A. Vialard · J.P. Domenger

Received: date / Accepted: date

Abstract This article proposes an approach to predict the result of binarization algorithms on a given document image according to its state of degradation. Indeed, historical documents suffer from different types of degradation which result in binarization errors. We intend to characterize the degradation of a document image by using different features based on the intensity, quantity and location of the degradation. These features allow us to build prediction models of binarization algorithms that are very accurate according to R^2 values and p-values. The prediction models are used to select the best binarization algorithm for a given document image. Obviously, this image-by-image strategy improves the binarization of the entire dataset.

Keywords Document image analysis · Quality evaluation · Binarization prediction

V. Rabeux
351, cours de la Libération F33405 Talence cedex
Tel.: +33(0)5 4000 69 00
Fax: +33(0)5 4000 66 69
E-mail: rabeux@labri.fr

N. Journet
351, cours de la Libération F-33405 Talence cedex
Tel.: +33(0)5 4000 69 00
Fax: +33(0)5 4000 66 69
E-mail: journet@labri.fr

J.P.Domenger
351, cours de la Libération F-33405 Talence cedex
Tel.: +33(0)5 4000 69 00
Fax: +33(0)5 4000 66 69
E-mail: domenger@labri.fr

A. Vialard
351, cours de la Libération F-33405 Talence cedex
Tel.: +33(0)5 4000 69 00
Fax: +33(0)5 4000 66 69
E-mail: vialard@labri.fr

1 Introduction

This paper involves quality evaluations of document images. Document quality evaluation is needed at every stage of the digitization workflow, for instance in the scanning stage to ensure that the scanner's settings are optimal, in the processing stage to apply the best algorithms (for example, restoration, binarization, OCR) and in the visualization stage to provide the best image quality.

To improve the results of the processing stage, it is necessary to take into account specific image defects. Document images may suffer from several types of degradation. According to [**?**], degradation can have different origins :

– wrong scanner settings: non-uniform illumination, focus, wrong white balance;
– the document itself: non-flat paper surface, spots, bleed-through; and,
– pre-processing algorithms, such as those involving high compression.

As ancient documents often present significant degradation, we focus this paper on their quality evaluation. However, the global methodology is suited for any damaged document images.

Essentially, most document analysis systems are created by sequentially applying algorithms (preprocessing, binarization, layout analysis, OCR, indexing). These chains of algorithms are *ad-hoc* workflows, built for a specific set of images. In such a workflow, the result of one algorithm can affect the result of all of the following ones. For example, Figure 1 illustrates the effect of a binarization algorithm on the result of a layout analysis. Thus, choosing the best algorithm available for one specific image is very important at each step of a workflow.

This choice needs to be automated to avoid tremendous manual work.

Currently, most proposals to automate the selection process of a document image workflow are based on performance evaluation. For example, the authors of [?] propose a software architecture for comparing algorithms and evaluating their performance on a complete workflow. Obtaining the best workflow implies an evaluation of a significant amount of algorithm combinations on a representative dataset. In [?] the authors propose an original method to improve the overall OCR process by combining several global thresholding binarisation results.

To automate the selection process, confidence rates can also be used. A confidence rate provides an estimate of how well the algorithm performed on an image. Confidence rates are well known in OCR systems, unfortunately most other processing algorithms do not associate confidence rates with their results. Moreover, even if available, confidence rates are not always relevant (see Figure 2).

We propose another approach, based on algorithm prediction models, to select the best algorithm for a specific task. Our approach is based on the following fact: the global quality of a document image directly affects the result of any processing algorithm (binarization, segmentation). Thus, we aim to predict the result of an algorithm according to the degradation type and quantity of the processed document. In this paper, we focus on binarization algorithm prediction.

For a given binarization algorithm and a set of images with their binarization ground truth, the significant correlation between algorithm performance and the quality of the images allows us to build a prediction function. The document image quality is characterized by new, dedicated features. This prediction function can forecast the binarization algorithm result for any new image on which quality features have been previously computed.

In the following sections, we first present the state of the art for image quality evaluation and for algorithm prediction in the context of document image analysis (Section 2). We then introduce different features that characterize ancient document degradation. These features rely on a decomposition of the document gray levels in three different classes: ink pixels, degradation pixels and background pixels (Section 3). We characterize the degradation layer by analyzing the distribution of its intensities, its quantity and its location within the image. The proposed features, dedicated to binarization evaluation, are presented in Section 4. Section 5 details the methodology used for creating algorithm prediction models. Prediction models of several binarization meth-

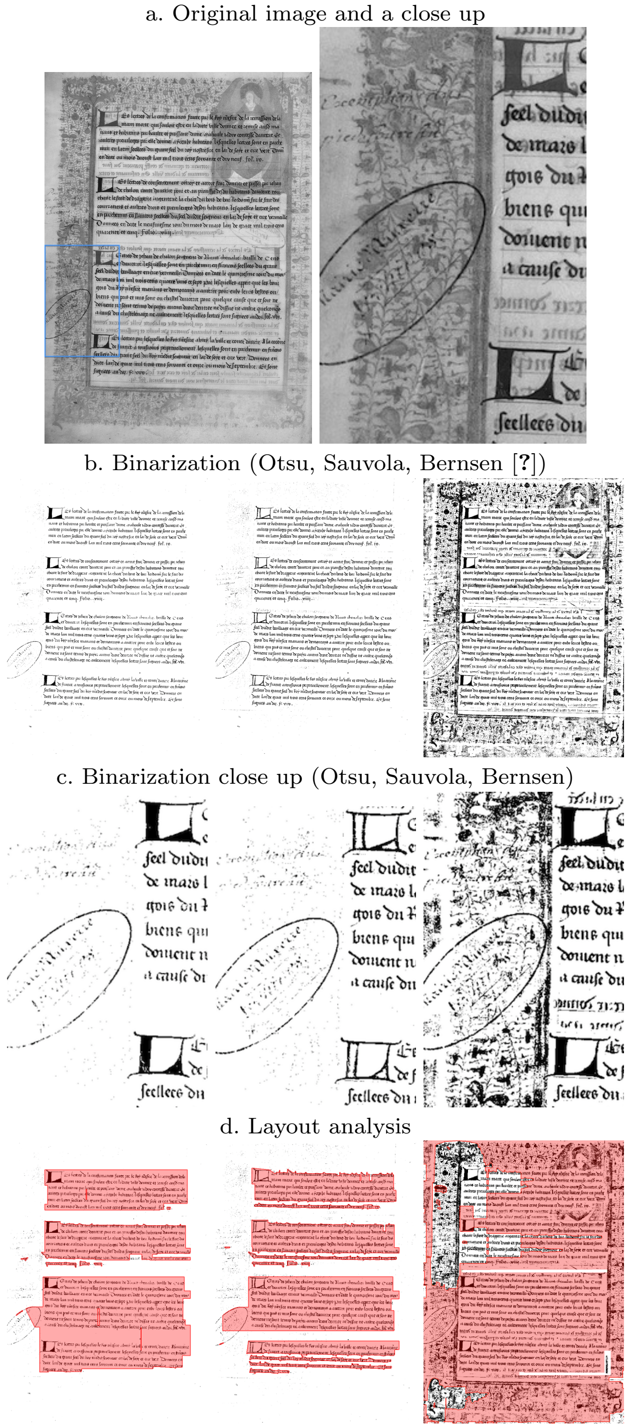


Fig. 1: Effect of binarization errors on a layout analysis algorithm. The processed document is composed of four text paragraphs. The background is severly degraded by a large amount of bleed-through particularly in the margins (a). Three binarization algorithms are applied to the document image. Depending on the chosen binarization method, the bleed though of the original document will induce more or fewer binarization errors. For example, the Bernsen's algorithm clearly fails where bleed-through is important (b-c). In the next step, a layout analysis algorithm is applied to the binarized document. The more binarization errors there are, the more inaccurate the layout analysis (in light red, text blocks are extracted with a white spaces segmentation algorithm).

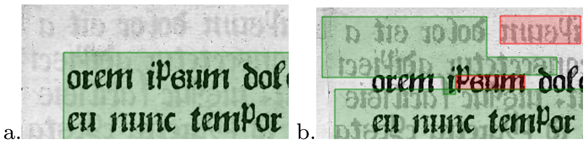


Fig. 2: OCR (ABBYY Finereader 9) errors from bleed-through: a document with a low level of bleed-through (a), and the same document with a higher level of bleed-through(b). The OCR fails when the document contains more bleed-through. A red zone corresponds to a low confidence measure, whereas a green zone corresponds to a high confidence measure: the OCR may be confident that some bleed-through regions are text regions (b).

ods are then presented, all of which present very high accuracy. Finally, Section 6 explains how to use the prediction models to select the best binarization algorithm for a specific image.

2 Related works

The first part of our work is to identify the degradation within document images. The related works are ancient document image enhancement methods with a first step that often consists of identifying and localizing specific degradations pixels.

Among the methods focusing on pixel degradation identification, the authors of [?] propose a directional wavelet transform to identify bleed-through pixels. The authors of [?] also localize document pixels suffering from bleed-through by a recto-verso registration: a parameter optimization method aims to find the appropriate transformation matrix that minimizes the difference between gray recto pixels and ink pixels from the verso. The recto pixels corresponding to the verso ones can then be labelled as bleed-through pixels. The problem addressed in [?] is the localization of pixels that suffer from illumination defects. This problem occurs when scanning documents with large bookbindings. The authors propose a line-by-line thresholding to localize the boundary of the dark area near the bookbinding.

The pixel identification methods previously mentioned are dedicated to the restoration of one specific defect (for example, bleed-through, illumination). However, typically, ancient documents suffer from a combination of defects. For example, the recto verso registration to localize bleed-through pixels may fail with a document suffering from geometrical distortions. A global approach has been chosen in recent restoration methods [?,?]. We also believe that a robust identification of defect pixels has to be performed globally.

The second part of our work involves predicting the result of a binarization algorithm. To our knowledge, there are no studies on binarization prediction. The existing work on algorithm prediction for document image analysis is only found in the OCR field, which typically use the quality features of characters to create prediction models.

The first features related to character quality were introduced in [?]. In this article the authors evaluate the quality of binary text documents by analyzing black and white connected components. The OCR result is predicted by simply thresholding the quality ratios (proportion of thick and broken characters). Each document image is finally labeled as good or poor. In [?], two new measures are introduced to account for speckles and connected characters. A linear regression is used to predict the OCR performance on handwritten black and white documents. The authors of [?] complete the set of features with new ones (Black Density Factor, Stroke Thickness Factor), which are used as inputs to a neural network to classify images into two classes (poor or good). By reusing a script identification engine, the method proposed in [?] can select the better of two OCRs according to a classification of the text image as *broken*, *clean* or *merged*. This classification is based on the computation of classical shape features of word images (compactness, Cartesian and centralized moments,...) and on a connected-component per word distribution.

Other works propose strategies to select the best restoration algorithm. As in OCR prediction methods, dedicated defect features are computed on a binary image. These values are then used as inputs for different types of semi-supervised classification algorithms. The authors of [?] use the features of [?] with three new ones from [?] (Small Speckles Factor, Font Size Factor and Broken Characters Factor) to select a restoration algorithm. The restoration algorithm selection is based on decision rules using thresholds. Another automatic restoration method selection is presented in [?]. In this latter article, the restoration algorithm selection is based on a linear classifier.

Previous methods suffer from two main drawbacks. First, most of them require a connected component extraction and, therefore, a binarization step. These methods strongly depend on the accuracy of this preprocessing step. We believe that a better approach consists of directly analyzing the defect pixels in the initial grayscale image.

Second, none of the presented articles dealing with prediction models analyze the significance of each feature. Only the authors of [?] propose an interesting correlation analysis between several quality metrics and the parameters of the degradation model used to produce the test images. This preliminary study shows that, even if most features are highly correlated with the defects, some are not. However, this study does not address the essential issue of the selection of relevant quality features to avoid overfitting of an algorithm prediction.

3 Degradation layer extraction

As in [?], we assume that an ancient document can be modeled as the combination of several information layers. Here, we consider three different layers: the text pixel layer, the background pixel layer and the degradation pixel layer. In ancient documents, most of the degradation (for example, bleed-through, spots, speckles, non-uniform illumination, ink loss) appears as connected components with grayscale values that differ from background and ink pixels. Figure 3.a shows several types of degradation in which the pixel gray intensities vary from low (ink spots) to high values (light bleed-through). We do not measure each type of degradation separately. On the contrary, we globally measure and characterize the document degradation by distinguishing three different layers of pixels according to the pixels' gray level. Let us denote the gray level of pixel p by $g(p)$. Let $\mathcal{I}$ be the set of ink pixels, $\mathcal{D}$ be the set of degradation pixels and $\mathcal{B}$ be the set of background pixels defined as follows:

1. $\mathcal{I} = \{p, g(p) \leq s_0\}$ ink layer
2. $\mathcal{D} = \{p, s_0 < g(p) < s_1\}$ degradation layer
3. $\mathcal{B} = \{p, g(p) \geq s_1\}$ background layer

Setting the two thresholds s_0 and s_1 can be determined using any classification algorithm. Our experiments used a 3-means clustering algorithm. Figure 3 shows that most degradation present in a document image can be extracted using these two thresholds. A few gray pixels (for example, from the background, or inside characters) are misclassified. Obviously, it is not possible to perfectly classify these pixels using only the gray-level histogram.

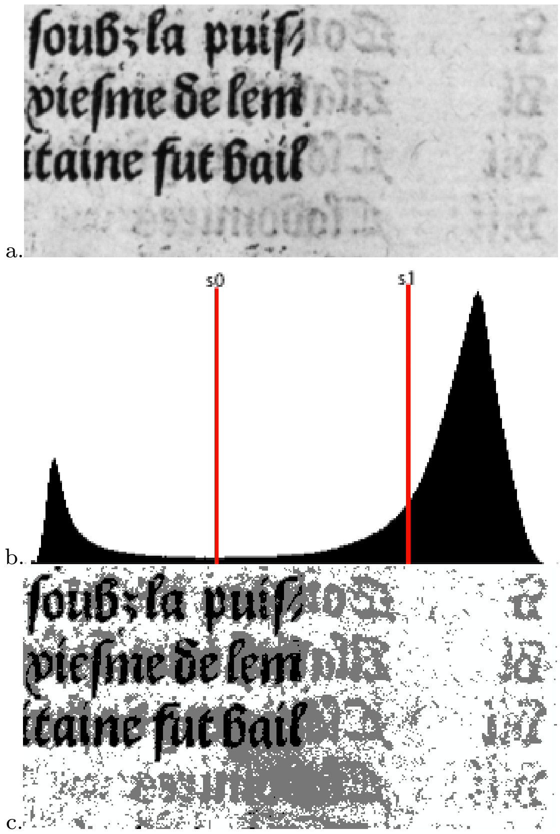


Fig. 3: The three classes of pixels. (a) the original grayscale document image. (b) its grayscale histogram with two thresholds s_0 and s_1 obtained by a 3-means algorithm. (c) classification result: pixels lower than the threshold s_0 in black, pixels between s_0 and s_1 in gray and pixels higher that s_1 in white. The gray set of pixels (between s_0 and s_1) contains most of the instances of document degradation, such as bleed through, speckles, spots and ink loss.

4 Quality features definition

This section details new features used to characterize document image degradation. All features are based on an analysis of the three layers previously extracted. A first set of global features is extracted directly from the three grayscale histograms without spatial consideration. A second set of spatial features is dedicated to the characterization of the localization of the degradation surrounding ink components.

4.1 Global Features

The global grayscale histogram contains information characterizing document quality. Figure 4 and Table 2 illustrate the differences between the histograms of a clean and a severely degraded document image.

We aim to compute the following global statistic features of the grayscale histogram: mean, variance and skewness. The skewness quantifies the asymmetry of the histogram. For example, a negative skewness indicates that the distribution of pixels gray-levels has relatively

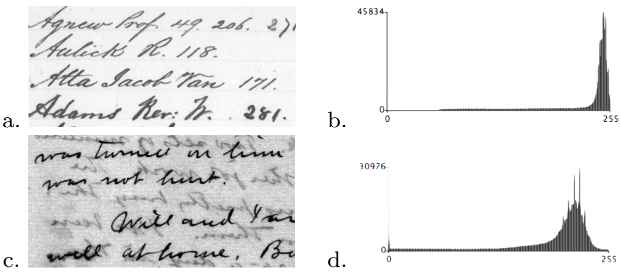


Fig. 4: Examples of global gray-level histograms: a relatively clean document (a), its corresponding gray-level histogram (b) , an ancient degraded document (c) and (d) its corresponding histogram (more scattered and irregular than in b). The gray-level histogram is used to provide a first indication of the quality of the document.

few low values. We denote the mean of the global histogram by μ, its variance by v, and its skewness by s. A good value for the skewness is a high negative value: the left tail of the histogram is longer, the intensities are concentrated on the right and the histogram has relatively few gray values. In that case, the image is likely easily binarized (see the images in Figure 4.a and Table 2 line 2) The mean, variance and skewness are also computed on the three *sub-histograms* to characterize each layer distribution (ink, background and degradation).

This step provides 12 features:

- μ, v, s (global histogram)
- $\mu_{\mathcal{I}}$, $v_{\mathcal{I}}$, $s_{\mathcal{I}}$ (ink histogram)
- $\mu_{\mathcal{D}}$, $v_{\mathcal{D}}$, $s_{\mathcal{D}}$ (degradation histogram)
- $\mu_{\mathcal{B}}$, $v_{\mathcal{B}}$, $s_{\mathcal{B}}$ (background histogram).

The previous global features characterizing the histograms cannot precisely represent the relationship between the ink layer, the degradation layer and the background layer. Therefore, we introduce two last global features extracted from the grayscale histogram to characterize the distance between the three layers. We assume that the mean intensity difference between the layers is directly correlated to the binarization result. For example, if the mean intensity value of the degradation layer is close to the ink intensity value, degradation pixels can be misclassified as ink pixels.

We define two features, $\mathcal{MI}_I$ and $\mathcal{MI}_B$, where $\mathcal{MI}_I$ corresponds to the distance between the average intensity of degradation pixels and the average intensity of ink pixels and, $\mathcal{MI}_B$ is the distance between the average intensity of degradation pixels and the average intensity of background pixels

$$\mathcal{MI}_I = \frac{\mu_{\mathcal{D}} - \mu_{\mathcal{I}}}{255}, \mathcal{MI}_B = \frac{\mu_{\mathcal{B}} - \mu_{\mathcal{D}}}{255}.$$

The gray-values of the three layers are not the only characteristics that could affect a binarization algorithm. The amount of degradation pixels is also directly correlated with the binarization performance. We aim to measure this performance as the relative quantity of ink and degradation pixels. We define $\mathcal{MQ}$ as the ratio:

$$\mathcal{MQ} = \frac{\|\mathcal{D}\|}{\|\mathcal{I}\|}.$$

This first family of features leads to the computation of a vector of dimension 15 for each document image.

4.2 Spatial features

Binarization is a segmentation task meant to extract objects of interest (for example, characters, drawings). A good binarization should preserve the shape of the objects and avoid the creation of unwanted black or white components. Obviously, the location of the degradation pixels is a significant characteristic that can influence the binarization result. Figure 5 illustrates the main situations observed in real documents in which the degradation pixels spatially interfere with ink pixels. For example, the binarization results worsen if dark spots overlap characters (Figures 5.b and c). In other words, an ink component may be even more deformed because it is connected with a degraded component. The following features are meant to capture, the possible creation of unwanted black components, and the possible deformation of the characters through the binarization process.

Let S be a set of pixels. We denote the set of the 4-connected components of S by $CC(S)$. In the rest of the section, we use the following notations: $\mathcal{C}_I = CC(\mathcal{I})$, $\mathcal{C}_D = CC(\mathcal{D})$ and $\mathcal{C}_B = CC(\mathcal{B})$.

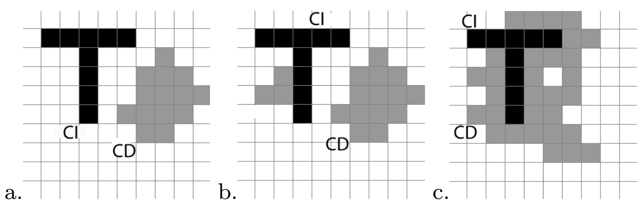


Fig. 5: The different locations of a degradation component on the page. The degradation component is not connected to an ink component (a), a small degradation component is adjacent to an ink component (b) and a large degradation component is adjacent to an ink component (c).

Let $c_{\mathcal{I}} \in \mathcal{C}_I$ be an ink component and $c_{\mathcal{D}} \in \mathcal{C}_D$ be a degradation component. We denote the predicate returning true by $SG(c_{\mathcal{I}}, c_{\mathcal{D}})$ if $c_{\mathcal{I}}$ and $c_{\mathcal{D}}$ are connected:

$SG(c_{\mathcal{I}}, c_{\mathcal{D}}) = \exists(p_{\mathcal{I}}, p_{\mathcal{D}})_{\in c_{\mathcal{I}} \times c_{\mathcal{D}}} \mid p_{\mathcal{I}}, p_{\mathcal{D}}$ are 4-connected.

We distinguish three different cases that can produce different types of binarization errors:

1. If $c_{\mathcal{I}}$ and $c_{\mathcal{D}}$ are not connected (Figure 5.a), the original character will not be altered by the binarization process. If this configuration occurs numerous times, the binarization can lead to a document image highly degraded by many small black spots between characters. Let $C_{\mathcal{MA}}$ be the set of degradation components that are not connected to any ink component:

$$C_{\mathcal{MA}} = \{c_{\mathcal{D}} \in \mathcal{C}_D \mid \forall c_{\mathcal{I}} \in \mathcal{C}_I, SG(c_{\mathcal{I}}, c_{\mathcal{D}}) = false\}.$$

The relative quantity of non-connected ink and degradation components is measured by $\mathcal{MA}$:

$$\mathcal{MA} = \frac{\|C_{\mathcal{MA}}\|}{\|\mathcal{C}_I\|}.$$

2. If $c_{\mathcal{I}}$ and $c_{\mathcal{D}}$ are connected (Figure 5.b), the original character may be altered by the binarization: degraded pixels may be misclassified as ink pixels. Let $C_{\mathcal{MS}}$ be the set of all ink components that are connected to at least one degradation component:

$$C_{\mathcal{MS}} = \{c_{\mathcal{I}} \in \mathcal{C}_I \mid \exists c_{\mathcal{D}} \in \mathcal{C}_D, SG(c_{\mathcal{I}}, c_{\mathcal{D}})\}.$$

The feature $\mathcal{MS}$ is defined as the ratio between the number of ink components that may be expended by at least one degradation component and the total number of ink components:

$$\mathcal{MS} = \frac{\|C_{\mathcal{MS}}\|}{\|\mathcal{C}_I\|}.$$

3. $\mathcal{MSG}$ measures the possible extent of ink component deformation using the number of known ink components that may be modified by the binarization process. It is defined as the mean area of the pairs of components that satisfy SG over the mean area of all ink components:

$$\mathcal{MSG} = \frac{Average_{\{(c_{\mathcal{I}}, c_{\mathcal{D}}) \mid SG(c_{\mathcal{I}}, c_{\mathcal{D}})\}}(\|c_{\mathcal{I}}\| + \|c_{\mathcal{D}}\|)}{Average_{c_{\mathcal{I}} \in \mathcal{C}_I}(\|c_{\mathcal{I}}\|)}.$$

The higher $\mathcal{MSG}$ is, the more likely it is that the document has large spots around ink components. Combined with other features (for example, $\mathcal{MI}_I$), $\mathcal{MSG}$ helps predict whether the spots lead to binarization errors.

Table 1: Comparison of spatial features in Figure 5.

	$\mathcal{MA}$	$\mathcal{MS}$	$\mathcal{MSG}$
Figure 5.a	1	0	0
Figure 5.b	1	1	1.3
Figure 5.c	0	1	4.3

Given all of the previously defined features, each document image is characterized by a vector of dimension 18. The table 1 shows the values of the three spatial deformation features on the examples in Figure 5.

The feature $\mathcal{MA}$ is equal to 1 in Figure 5.a and Figure 5.b because one degradation component is connected to one ink component. The feature $\mathcal{MSG}$ equals to 0 in Figure 5.a because no component are connected. The feature $\mathcal{MSG}$ has a lower value in Figure 5.b than in Figure 5.c because the union area size between the ink component and the degradation component is smaller.

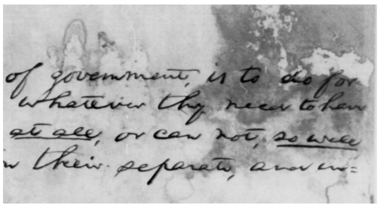
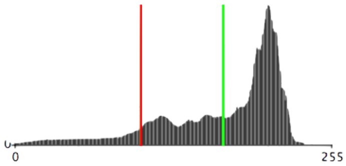
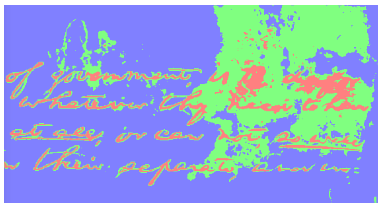
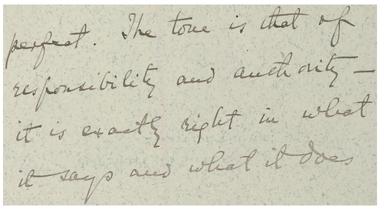
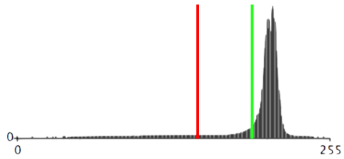
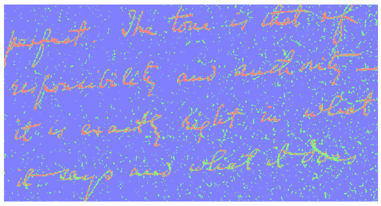
4.3 Case study

This section analyzes the 18 features computed on two different document images containing several defects (Table 2). These two examples emphasize the link between the 18 features and the f-score obtained after having binarized the images with Otsu's and Sauvola's methods (a global *versus* a local thresholding method). There are multiple ways to measure binarization accuracy. In this paper, we used the f-score.

The first document image (Table 2, line 1) is damaged by a large spot that overlaps text lines. The gray-levels of the spot are close to the gray-level of the text pixels. Because the Otsu method is based on a global threshold, the spot pixels tend to be misclassified as ink. On the contrary, the local method is more likely to achieve a correct separation of ink and background on the defective area, which explains why the respective f-scores of the Otsu and Sauvola methods are 0.4 and 0.7 on this image. The second document image (Table 2, line 2) presents a non-even background with speckles. Moreover the ink color is light relative to the background color. On this image, the respective f-scores of Otsu and Sauvola are 0.8 and 0.4. The Sauvola method is not robust to the background speckles, which are classified as ink. The faded ink defect is a drawback for a global method and lowers the performance of Otsu's method.

Table 2 shows that specific defects that reduce binarization performance are captured by the proposed features. Even if the global features based on histogram

Table 2: Two document image examples from the DIBCO dataset and their feature vectors. The proposed features capture different degradation types (for example, ink spots, faded ink, background speckles)

Image					GrayScale Histogram						3-mean clusters						
																	
$\mathcal{MI}_I$	$\mathcal{MI}_B$	$\mathcal{MQ}$	$\mathcal{MA}$	$\mathcal{MS}$	$\mathcal{MSG}$	s_I	s_D	s_B	v_I	v_D	v_B	μ_I	μ_D	μ_B	s	v	μ
0.27	0.25	3.26	0.05	0.2	3.6	-0.4	-0.05	-0.5	741	392	161	66	135	199	-1.25	2065	171
Image					GrayScale Histogram						3-mean clusters						
																	
$\mathcal{MI}_I$	$\mathcal{MI}_B$	$\mathcal{MQ}$	$\mathcal{MA}$	$\mathcal{MS}$	$\mathcal{MSG}$	s_I	s_D	s_B	v_I	v_D	v_B	μ_I	μ_D	μ_B	s	v	μ
0.19	0.17	1.23	0.3	0.2	1.4	-0.6	-0.02	-0.5	257	206	30	98	146	189	-3	356	185

analysis are meaningful, they are not sufficient in that case to choose the best binarization method. The ink pixels' mean value μ_I of the first image is lower than that of the second one, indicating that the ink layer seems easier to identify using a global thresholding method. However, the skewness of the ink s_I is negative, indicating that most pixels are concentrated on the right part of the distribution: there are more gray pixels than really dark pixels. The skewness of the second global histogram s is much higher than that of the the first image, indicating that the background of the second image is easy to separate using a global thresholding method. This separation is confirmed by the global variance v. Without additional information, the global thresholding method seems adapted to the second image but we cannot draw a similar conclusion for the first image.

In the first image, the values of $\mathcal{MI}_I$ and $\mathcal{MI}_B$ are low, indicating that a global thresholding method is likely to fail to correctly classify the pixels. The value of $\mathcal{MSG}$ is also high, indicating that there are large spots around the characters. Generally, window-based methods have better results on this type of document.

On the second image, the values of $\mathcal{MI}_I$ and $\mathcal{MI}_B$ are even lower: Otsu's method will also yield a bad result for the second image, but other features such as s or the relatively low value of v indicate that failure may be relative. Moreover, the value of $\mathcal{MA}$ is high, meaning that many components do not touch text pixels. This type of degradation is likely to produce binarization errors with windows based methods such as Sauvola's method.

According to the computed features, it is preferable to use Sauvola's method for the first image and Otsu's for the second. Doing so is consistent with the f-scores of the two binarization methods.

The proposed features characterize three different aspects of degradation: intensity, quantity and location. The next section details a methodology that uses these features to predict the result of a binarization algorithm, which is applied to the prediction of 12 binarization algorithms on the DIBCO dataset.

5 Predicting binarization method accuracy

This section presents a unified methodology that is able to predict the result of most binarization methods (for example, adaptive thresholding, clustering, entropic, document dedicated). Our methodology is evaluated on 12 binarization methods used in document analysis. The methods are referenced in the text by their author's names.

1. Bernsen [?] is a local adaptive thresholding technique.
2. Kapur [?] is an entropy-based thresholding method.
3. Kittler [?] is a clustering-based thresholding algorithm.
4. Li [?] is a cross-entropic thresholding method based on the minimization of an information theoretic distance (Kullback-Leibler).
5. Niblack [?]: is a locally adaptive thresholding method using pixels intensity variance.

6. Ridler [?] is an iterative thresholding method based on two-class Gaussian mixture models.
7. Sahoo [?] is an entropy-based thresholding method.
8. Shanbag [?] is a fuzzy entropic thresholding technique that considers fuzzy memberships as an indication of how strongly a gray value belongs to the background or to the foreground.
9. Sauvola [?] is a locally adaptive thresholding algorithm using pixel intensity variance.
10. Otsu [?] is a two-class global thresholding method.
11. White [?] is a locally adaptive thresholding method using local contrast.
12. Shijian Lu [?] is a recent method based on an *ad-hoc* combination of existing techniques. [?] has proven to have very good accuracy on the ICDAR 2009 Binarization Contest.

Some binarization methods rely on parameters. In this article, we do not focus on parameter optimization. Therefore, we chose to use the parameters given by the authors of each method in their corresponding original articles. Table 3 summarizes the values of these parameters. Importantly, note that the prediction models created are only able to predict the performance of a binarization method with a specific set of parameters. However, a binarization method can have several prediction models, one for each set of parameters. To illustrate the difference between two sets of parameters, we will create two different prediction models for Sauvola's method. The second set of parameters was manually chosen (Table 3).

Table 3: Method parameters: we chose to use the parameters given by each author in their original articles.

Method	Parameters	
Bernsen	window size	31
White	window size	15
	bias	2
Niblack	window size	15
	K	-0.2
Sauvola (original parameters)	window size	15
	R	128
	K	0.5
Sauvola (manual parameters)	window size	51
	R	128
	K	0.4

In order to assess the accuracy of a binarization method, several measures can be used. Most of them are presented in [?,?]. The authors of [?] propose an interesting study of these measures regarding to the human perception of image quality. Their main conclusion is that the human perception is consistent with the classical measures for the ranking of bests and worsts binarization methods. In our experiments, we choose to use the f-score measure.

To predict the accuracy of the binarization method, we follow a methodology based on a step-wise linear regression. Section 5.1 presents the dataset we used to train and validate our prediction models. This predictive methodology can be applied to all types of binarization methods and is presented in a general way in Section 5.2. We then detail the prediction models corresponding to three popular binarization methods for document images: Otsu's, Sauvola's and Shijian Lu's (Section 5.3). The prediction model accuracy for the other methods is presented in Section 5.4 to highlight the generality of the presented methodology.

5.1 The dataset

To create a precise and usable prediction model, we need a dataset of images with their binarization ground truth. This dataset needs to be heterogeneous. In our case, a well distributed dataset should contain images with various levels of defects leading to various f-scores for the different binarization methods.

We use a dataset obtained by merging the DIBCO[1] and H-DIBCO[2] datasets [?,?,?]. These datasets are primarily used as data for binarization contests and contain a heterogeneous set of images from difficult to easy to binarize. Table 4 summarizes some statistical results of the F-score measures for 12 binarization algorithms applied to all DIBCO images (36 images).

The DIBCO datasets are currently the reference used for binarization contests and in other scientific articles dealing with binarization problems. These images were selected by the DIBCO team on the fact that they have different caracteristics and degradation amounts inducing different effects on binarization methods. The DIBCO datasets are used for performance evaluation [?,?] or for learning and trainning steps [?].

5.2 Using step wise multivariate linear regression to predict the result of the binarization algorithm

To estimate the f-score of a binarization algorithm, we automatically build a prediction model based on the most significant features among the 18. More precisely, the prediction model is computed using multivariate step wise linear regression [?,?,?,?], followed by repeated random sub-sampling validation (cross validation).

[1] http://users.iit.demokritos.gr/∼bgat/DIBCO2009/
[2] http://users.iit.demokritos.gr/∼bgat/H-DIBCO2010/

Table 4: Statistical results of the F-score measures for 12 binarization algorithms applied to all DIBCO images. Except for the Sahoo algorithm, all binarization methods have a significant min/max f-score gap and standard deviation between 0.1 and 0.3, indicating that the dataset is heterogeneous and well suited for the learning step of our prediction model.

F-Score	Mean	Std. Dev.	Min	Max
Bernsen	0.48	0.47	0.10	0.85
Kapur	0.84	0.07	0.63	0.94
Kittler	0.77	0.16	0.24	0.95
Li	0.69	0.20	0.10	0.96
Niblack	0.35	0.14	0.1	0.6
Ridler	0.82	0.14	0.28	0.96
Sahoo	0.82	0.01	0.50	0.96
Sauvola (original)	0.72	0.22	0.00	0.95
Sauvola (manual)	0.55	0.28	0.10	0.91
Shanbag	0.81	0.11	0.49	0.93
Shijian Lu	0.89	0.12	0.21	0.95
Otsu	0.81	0.14	0.28	0.96
White	0.40	0.22	0.00	0.83

The linear regression models, as an hyperplane, the relationship between the features and the ground truth f-scores. This result can then be used to predict a f-score according to a set of computed features. The prediction can be improved by using only a pertinent subset of features among the 18 independent computed features. There are three main ways to carry out a selection. First, the forward strategy consists in computing a criteria (linked to the R^2 value) by adding one feature at a time. On the contrary, a second approach (backward strategy) consists in starting with all the features and deleting them one at a time. After each deletion the criteria is computed. The last strategy consists in testing all the possible combinations. As we have only 18 features, we decided to use the exhaustive strategy.

This overall process which is presented on Figure 6 can be divided into five steps[3]:

1. **Feature computation:** The 18 proposed features are computed for each image.
2. **F-score computation:** We run the binarization algorithm and compute the f-score for each image by comparing the binarization result and the ground truth. In the following, these ground truth f-scores and denoted by $f_i, i \in [0, N]$ (with N the total number of images).
3. **Generation of the predictive model**: This step consists of applying a step wise multivariate linear regression to the overall dataset, allowing us to select the most significant features for predicting the given binarization algorithm. Keeping all features in each prediction model would lead to overfitt models. Indeed, some features may not be significant for predicting a specific binarization method. Moreover, even if a feature is highly correlated to the accuracy of an algorithm, it may have a weak contribution to the final prediction model. The output of this step is a linear function that gives a predicted f-score value for any image, for one binarization algorithm, knowing the selected features. We denote by $\hat{f}_i, i \in [0, N]$ the predicted f-scores.
4. **Evaluation of model accuracy**: The R^2 value
$$1 - \frac{\sum_i (\hat{f}_i - f_i)^2}{\sum_i (f_i - \bar{f})^2},$$
with $\bar{f}$ the mean of ground truth f-scores, measures the quality of the prediction model. It can be interpreted as a correlation between the ground truth and the prediction.
The best theoretical value for R^2 is 1. Moreover, a p-value is computed for each selected feature indicating its significance: a low p-value leads to reject the hypothesis that the selected feature is not significant (null hypothesis). At this step, there is no automatic rule to decide whether a model is valid or not. However, in our experiments, we choose to keep the model only if the R^2 value is higher than 0.7 and if a majority of p-values are lower than 0.1.
5. **Model validation**: Because of the relatively few images in the dataset, we use cross validation to estimate the performance of the predictive function generated in step 2. We randomly split the overall set of images into two different subsets: the training set and the validation set. In our experiments, the training set is composed of 90% of the dataset images and the validation set is composed of the remaining 10%.
By applying linear regression to the training set, we compute a new prediction function with its associated R^2. The features used here are those selected at step 2. We apply this new predictive function to the validation set. The obtained predicted f-scores are compared with the ground truth f-scores using a linear regression that provides the slope coefficient α. If α is close to 1, then the f-scores of the binarization are well predicted on the validation set.
This step combines training and validation and is repeated 100 times to ensure that the prediction model accuracy is independent of the random learning step (also known as "statistical type III error"). The results (R^2 and α) are then averaged over the 100 random splits. The predictive model of step 2

[3] The overall R project script and our evaluation data can be downloaded from the following website `https://bitbucket.org/vrabeux/qualityevaluation`

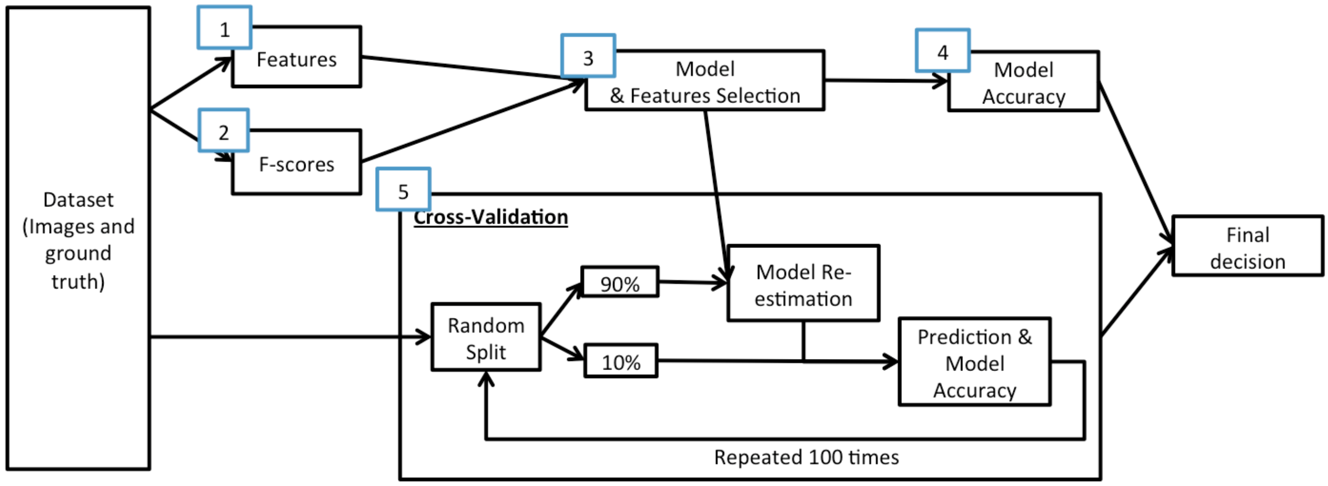


Fig. 6: Overall process to create a prediction model for a specific binarization algorithm.

is finally validated only if $\bar{R}^2$ and $\bar{\alpha}$ are close to 1 (>0.7 in our experiments).

Some binarization methods described in this article require parameter settings. Note that our methodology involves the creation of different predictive models, one for each parameter set.

5.3 Prediction models of commonly used binarization methods in document analysis systems

The prediction model for Otsu's, Sauvola's and Shijian Lu's binarization algorithms were generated with the methodology described in Section 5.2. The coefficients associated with the most significant selected features, their p-values and the intercept of the linear predictive function are detailed in Tables 5, 6 and 7. If a feature is not present in a table, then it was not selected by the step wise algorithm. As mentioned in the previous section, the cross validation for each model gives the pair $(\bar{\alpha}, \bar{R}^2)$.

Otsu's binarization method The most significant selected features for Otsu's prediction model are $\mathcal{MI}_I$, v_I, v_B, μ_B, μ and v (see Table 5 for the coefficients of the predictive function). For Otsu's prediction model, we can explain the feature selection by the fact that Otsu's binarization method is based on global thresholding. That is why features such as $\mathcal{MI}_I$, μ and v are significant and have such low p-values. The model's R^2 equals 0.93, which is considered very good [?].

The cross-validation gives a $\bar{\alpha}$ coefficient of 0.989 and $\bar{R}^2$ of 0.987. These results indicate that our model does not depend on the chosen training data.

Table 5: Otsu's prediction model: all selected features are significant (p-value < 0.1), and the model is likely to correctly predict future unknown images given that the R^2 value is higher than 0.9. The mean percentage error is denoted by *mpe*.

	Feature coef.	p-value
Intercept	$1.19e+00$	$< 10^{-4}$
$\mathcal{MI}_I$	$1.24e+00$	$< 10^{-4}$
v_I	$2.42e-02$	$< 10^{-1}$
v_B	$-4.34e-02$	$< 10^{-2}$
μ_B	$-2.66e-02$	$< 10^{-4}$
μ	$2.44e-02$	$< 10^{-4}$
v	$3.26e-04$	$< 10^{-4}$
$R^2 = 0.93, mpe = 5\%$		

Sauvola's binarization method For Sauvola's binarization method, we created two different models (Table 6). The first one corresponds to the set of parameters proposed by the authors in their original article, and the second one corresponds to the set of parameters that were manually chosen (see Table 3).

For the first model (original parameters), the selected features for Sauvola are: $\mathcal{MI}_B$, $\mathcal{MQ}$, $\mathcal{MA}$, μ, s, s_I, v_I. The estimated coefficients are presented in Table 6. The resulting prediction model also seems accurate with an R^2 value of 0.8372. Note that $\mathcal{MQ}$ and $\mathcal{MA}$ are selected for this binarization method. Indeed, window-based methods are sensitive to noise components near ink components. The cross validation gives a mean slope coefficient $\bar{\alpha}$ of 0.85 and an $\bar{R}^2$ of 0.8.

The results are similar using the second model that predicts Sauvola's binarization method f-scores with our manually chosen parameters. However, the feature $\mathcal{MSG}$ is introduced in this model, which can be explained by the fact that we changed the *window size* parameter (51 pixels instead of 15). Indeed, using this window size, the results of Sauvola are sensitive to large gray components surrounding characters. The cross-validation step gives a slope coefficient $\bar{\alpha}$ equal to 1.114 and an R^2 of 0.94.

These results allow us to conclude that these models are accurate and can be used in practice.

Table 6: Sauvola prediction models.

Sauvola (original parameters) prediction model		
	Feature coef.	p-value
Intercept	$1.54e+00$	$< 10^{-4}$
$\mathcal{MI}_B$	$1.09e+00$	$< 10^{-2}$
$\mathcal{MQ}$	$-1.33e+00$	$< 10^{-4}$
$\mathcal{MA}$	$2.68e-02$	$< 10^{-1}$
$\mathcal{MSG}$	$1.08e-02$	$< 10^{-1}$
μ	$-3.91e-03$	$< 10^{-2}$
s	$8.89e-02$	$< 10^{-4}$
s_I	$1.34e-01$	$< 10^{-4}$
v_I	$4.41e-04$	$< 10^{-4}$
$R^2 = 0.83, mpe = 10\%$		
Sauvola (manually chosen parameters) prediction model		
	Feature coef.	p-value
Intercept	$1.61e+00$	$< 10^{-4}$
$\mathcal{MI}_B$	$1.19e+00$	$< 10^{-2}$
$\mathcal{MQ}$	$-1.10e+00$	$< 10^{-3}$
$\mathcal{MA}$	$2.30e-01$	$< 10^{-1}$
μ	$-4.56e-03$	$< 10^{-4}$
s	$7.71e-02$	$< 10^{-4}$
s_I	$1.43e-01$	$< 10^{-4}$
v_I	$4.26e-04$	$< 10^{-4}$
$R^2 = 0.84, mpe = 7\%$		

Shijian Lu's binarization method The selected features and their estimated coefficients for Shijian Lu's prediction model are presented in Table 7. The step wise linear regression selects two spatial deformation features, $\mathcal{MA}$ and $\mathcal{MSG}$, and a global feature, $\mathcal{MI}_B$. This choice is not surprising because this method is a combination of several global and window based techniques. The prediction model is also very accurate (0.86). The cross validation gives a $\bar{R}^2$ of 0.99 and a mean slope $\bar{\alpha}$ of 1.06.

5.4 Accuracy of other prediction models

The same experiment was conducted on the other binarization methods. Table 8 sums up the selected features and the significant information to validate or not a binarization prediction model.

Table 7: Shijian Lu's prediction model. The mean percentage error is denoted by *mpe*.

	Feature coef.	p-value
Intercept	$1.07e+00$	$< 10^{-4}$
$\mathcal{MI}_B$	$-7.97e-01$	$< 10^{-1}$
$\mathcal{MA}$	$3.16e-02$	$< 10^{-4}$
$\mathcal{MSG}$	$-3.28e-02$	$< 10^{-4}$
var	$-1.39e-04$	$< 10^{-4}$
s_I	$3.88e-02$	$< 10^{-4}$
s_D	$1.33e-01$	$< 10^{-3}$
μ_I	$-4.00e-04$	< 0.5
$R^2 = 0.86, mpe = 5\%$		

Among the 18 features, most models embed about 7 features. Globally the selected features are consistent with the binarization algorithm: the step wise selection process tends to keep global (resp. local) features for global (resp. local) binarization algorithms. We also note that $\mathcal{MS}$ is never selected by any prediction model. Indeed, the binarization accuracy is measured at the pixel level (f-score). With this accuracy measure, the feature $\mathcal{MSG}$ becomes more significant than $\mathcal{MS}$, which may not have been the case with another evaluation measure.

The R^2 values show the quality of each prediction model. The prediction models of Sahoo and Niblack binarization methods were not kept for the statistical validation step since the R^2 values were below 0.7. For these two binarization models new features have to be created in order to obtain more accurate prediction models.

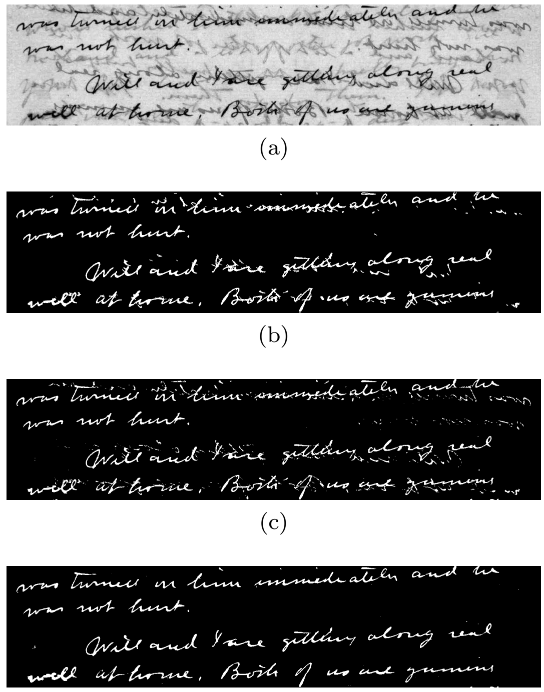
The two values $\bar{R}^2$ and *mpe* show the accuracy of each prediction model on the validation step. A $\bar{R}^2$ value higher than 0.7 indicates that it is possible to predict the results of a binarization method [**?**]. As a result, 12 binarization methods can be well predicted. The mean percentage error (*mpe*) is the average difference between predicted f-scores and real f-scores. This value is around 5%.

6 Automatic and optimal selection of binarization methods

The methodology previously explained allows the creation of an accurate prediction model for any binarization method. Given a document image, a binarization method and its prediction model, we can compute all of the features required by the model and use them as inputs. The result is the predicted accuracy of this specific binarization method for this specific image. Given

Table 8: Accuracy of the prediction model for the other binarization methods. The selected features are different from one method to another. Two models where not kept for the cross-validation step due to their low R^2 value (< 0.7). The accuracy and robustness of the other prediction models are good (cross validation $\bar{R}^2 > 0.7$). The mean percentage error of each model is denoted by *mpe*.

Method	Selected Features	R^2	$\bar{R}^2$	*mpe*
Bernsen	$\mathcal{MI}_I$; $\mathcal{MA}$; $\mathcal{MSG}$; v; v_D; v_I	0.83	0.96	6%
Kapur	$\mathcal{MI}_I$; $\mathcal{MA}$; μ; v; s_D; v_I; μ_D; μ_I	0.78	0.99	2%
Kittler	$\mathcal{MI}_I$; $\mathcal{MQ}$; s; v_I; μ_B; v_B	0.84	0.98	5%
Li	$\mathcal{MI}_I$; $\mathcal{MA}$; $\mathcal{MSG}$; μ; v; v_I; μ_D; μ_I	0.81	0.95	11%
Niblack	μ; v; s_G; v_B; μ_B	0.59	-	-
Riddler	$\mathcal{MI}_I$; v; v_D; v_I	0.75	0.98	5%
Sahoo	$\mathcal{MI}_I$; μ; s_B; v_I; μ_D; μ_I	0.68	-	-
Shanbag	$\mathcal{MI}_I$; s; v; s_D; s_I; v_D; v_I	0.73	0.98	6%
White	$\mathcal{MI}_I$; $\mathcal{MSG}$; s; v; μ_D; μ_I; v_D	0.92	0.99	7%



(a) (b) (c) (d)

Fig. 7: Sophisticated binarization algorithms do not always give the best output: original image (a), Shijian Lu's binarization output (b), Sauvola's binarization output (c) and Otsu's binarization output (d). Ostu's algorithm has the best performance on this specific image.

several binarization prediction models, we can create a binarization process that uses these prediction models to select the optimal binarization method for each image of a dataset.

For instance, Shijian Lu's method is the binarization method which gives the best results on average. However, in some borderline cases Shijian Lu's significantly fails while other methods perform better. This is illustrated in Figure 7 where the bleed-through defect disrupts methods which use a local analysis of the image.

More generally, Table 9 presents some f-score statistics obtained from binarizing the DIBCO dataset. The first line corresponds to the best theoretical f-scores (having the ground truth, we know for each image the binarization method that will provide the best f-score). The second line corresponds to the f-scores obtained using only Shijian Lu's method. The last line corresponds to the f-scores obtained using our automatic binarization selection.

We analyse the accuracy of our binarization method selection algorithms in several ways. As expected, the method has only a slightly better (2%) mean accuracy than using only Shijian Lu's method. What is significant is that the standard deviation lowers from 0.12 to 0.04. It means that the worst binarization result of our method is much higher than Shijian Lu's (56%). We also compared our method with the optimal selection that we can compute from the ground truth. The results are very similar, indicating that the prediction models are accurate enough to select the best binarization method for each image (70% perfect match). The mean error of our method is 0.009 (standard deviation equals 0.02), and, the worst error equals 0.06.

Table 9: Binarization of the DIBCO dataset. Comparison between the best theoretical f-score (computed from the ground truth), f-scores obtained using only Shijian Lu's method and f-scores obtained from our automatic selection.

F-Score	Mean	Std. Dev.	Min	Max
Optimal selection	0.913	0.04	0.77	0.96
Shijian Lu	0.891	0.12	0.21	0.95
Automatic selection	0.906	**0.04**	**0.77**	0.96

These results are very encouraging and show that this naive selection technique can be used to improve the binarization of a document. Moreover, the selection errors can be minimized by promoting models with the highest R^2 and with the lowest possible p-values. In other words, a model with a high confidence rate (good R^2 and p-values) may be selected even if its predicted f-score is not the highest one.

7 Conclusion and research perspectives

This paper presented 18 features that characterize the quality of a document image. These features are used in step-wise multivariate linear regression to create prediction models for 12 binarization methods. Repeated random sub-sampling cross-validation shows that the models are accurate (max percentage error equals 11%). Moreover, given the step-wise approach of the linear regression, these models are not overfit. As a result, 10 models out of 12 are validated and show sufficient accuracy to be used in an automated selection method of the optimal binarization method for each image.

One of our future research goals is to apply the same methodology to predict OCR error rates. However, OCRs today are very complex engines that are able to restore documents and perform layout analysis. Therefore, OCR failure cases are not only the result of a document's quality but also of its complexity (font, tables, figures, mathematical formulas). This complexity has to be evaluated with new OCR dedicated features.

Our second research goal is to improve the binarization algorithm selection method. We believe that the method can be tuned by studying different strategies. One notion is to take into account R^2 and p-values measures in the automatic selection of a method. Another idea is to weight predicted f-scores with computational costs: with similar accuracy, choosing the quickest one may be preferable.

At last, 2 prediction models are rejected due to their lack of accuracy. New dedicated features have to be created and used in the presented methodology to circumvent this issue.

Acknowledgements We would like to thanks the DIBCO team for providing datasets. This work was completed within the DIGIDOC project financed by the ANR (*Agence Nationale de la Recherche*).