Parameter tuning for document image binarization using a racing algorithm

Rafael G. Mesquita[a], Ricardo M. A. Silva[a], Carlos A. B. Mello[a,*], Péricles B. C. Miranda[a,b]

[a] *Centro de Informática, Universidade Federal de Pernambuco, Brazil*
[b] *Departamento de Estatística e Informática, Universidade Federal Rural de Pernambuco, Brazil*

Abstract

Binarizaton of images of old documents is considered a challenging task due to the wide diversity of degradation effects that can be found. To deal with this, many algorithms whose performance depends on an appropriate choice of their parameters have been proposed. In this work, it is investigated the application of a racing procedure based on a statistical approach, named I/F-Race, to suggest the parameters for two binarization algorithms reasoned (i) on the perception of objects by distance (POD) and (ii) on the POD combined with a laplacian energy-based technique. Our experiments show that both algorithms had their performance statistically improved outperforming other recent binarization techniques. The second proposal presented herein ranked first in H-DIBCO (Handwritten Document Image Binarization Contest) 2014.

Keywords: Parameter tuning, document image binarization, racing algorithms

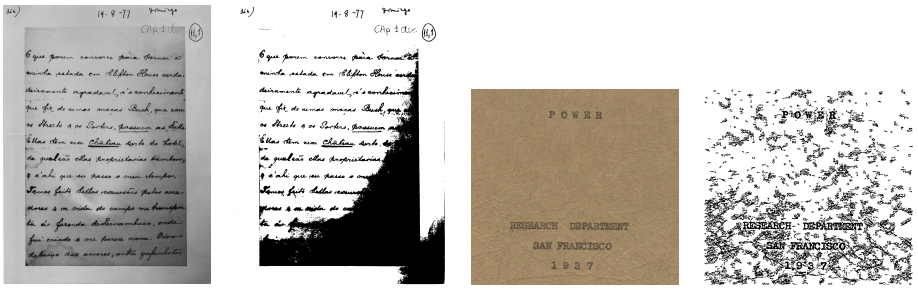
1. Introduction

Paper is still nowadays one of the most used medium to store and distribute information, even with the increase of the information technology that happened in the last century (Baird, 2003; de Mello et al., 2012; Snellen & Harper, 2002). This is true due to some of its properties, like (i) the possibility to easily read and write simultaneously, (ii) the independence of a power source, (iii) its portability and (iv) its low cost (Baird, 2003; Snellen & Harper, 2002). It is possible to find a large amount of documents (in format of paper) of high cultural or historical

*Corresponding author. Address: Centro de Informática (CIn), Universidade Federal de Pernambuco (UFPE), Av. Jornalista Anibal Fernandes, Cidade Universitária, 50740-560 Recife, PE, Brazil. Tel.: +55 8121268430; fax: +55 81 21268438

Email addresses: rgm@cin.ufpe.br (Rafael G. Mesquita), rmas@cin.ufpe.br (Ricardo M. A. Silva), cabm@cin.ufpe.br (Carlos A. B. Mello), pbcm@cin.ufpe.br (Péricles B. C. Miranda)

URL: http://www.cin.ufpe.br/~viisar (Carlos A. B. Mello)



(a) Original image (b) Otsu (c) Original image (d) *dibco*2011_8

Figure 1: Binarization algorithms applied to old document images

value in libraries, museums or government archives. However, in some situations, the use of paper to store information is not recommended as (i) it can suffer degradation due to man handling or by aging and (ii) it is hard to perform a keyword search. Nowadays, for preservation purposes, most of the document are being digitized which does not solve none of the problems previously mentioned. Usually, after digitization of the document, its image is converted into black and white in a process called Binarization or Thresholding (Gonzalez & Woods, 2010).

Binarization is usually performed for two main reasons: the first one is that it can reduce the space needed to store an image, which is particularly important when dealing with large data sets. The second one is that Optical Character Recognition (OCR) algorithms usually require binary images to proceed with the recognition of the characters. Binarization is an important step in the document image analysis pipeline (that usually includes digitization, binarization (Sezgin & Sankur, 2004), skew correction (Mascaro et al., 2010), text-line, word (Sanchez et al., 2011) and character segmentation (Lacerda & Mello, 2013) followed by character recognition (Cheriet et al., 2007; de Mello et al., 2012)) since its result affects further stages of the recognition. Thus, an unsuccessful binarization can also make it impossible to recognize characters, even for human beings (see Figure 1b). Nevertheless, binarization of document images is considered a challenging task, especially in the case of old documents, because in this kind of images it is possible to find different issues, like uneven illumination, faded ink, smudges and smears, bleed-through interference and shadows (Mello, 2010b; Mesquita et al., 2014; Ntirogiannis et al., 2013). Figure 1 presents some unsatisfactory results obtained by two binarization algorithms.

Many algorithms that aim to solve computationally complex problems have a number of parameters that need to be properly configured in order to achieve satisfactory results (Birattari et al., 2010). Those parameters are typically configured based on some executions with different candidate configurations chosen based on some personal experience. Usually this is a tedious and time consuming procedure that may not lead to satisfactory results. For example, in (Lin et al., 2014), a model to design a green transportation scheme based on Genetic Algorithm (GA) is proposed. Two possible values are considered for

each one of the four parameters of the GA and for each one of 24 different test scenarios the parameters of the GA are chosen based on at least five executions with different parametric configurations. Then, the best configuration is selected as the final solution for each scenario. Unfortunately, this scheme does not guarantee a good exploration of the search space (as only two possible values are considered for each parameter) and a considerable amount of time can be spent with executions using low performance settings. The process of selecting parameters is commonly treated as an optimization (Lin, 2010) problem in which a search technique is used to find the adequate ones for a given algorithm. In (Zhang et al., 2010), an Ant Colony Optimization-based algorithm used to optimize the parameters of a Support Vector Machine is presented. Furthermore, techniques as Particle Swarm Optimization (PSO) (Kennedy & Eberhart, 1995), Tabu Search (Wang et al., 2014) and Racing algorithms are other approaches commonly used for parameter selection by many authors. Among the most widely used techniques for tuning algorithms, it is highlighted the Racing Algorithm (Birattari et al., 2010). This algorithm aims to find a good configuration (model) from a given finite pool of alternatives through a sequence of steps. As the computation proceeds, if sufficient evidence is gathered that some candidate has lower performance than at least another one, such a candidate is dropped from the pool and the procedure is iterated over the remaining ones. The elimination of less promising candidates speeds up the procedure and allows a more reliable evaluation of the promising ones. Besides, other positive point of racing algorithms is that unlike certain meta-heuristics, as GA or PSO, which need to be tuned adequately to perform well, they are simple to be designed. Racing algorithms have been used for meta-heuristics tuning as it can be seen in (Pellegrini, 2005; Becker et al., 2006; Birattari et al., 2002), and to select parameters of learning algorithms (Maron & Moore, 1997; Maron, 1994). These applications have shown that algorithms can be tuned efficiently by racing procedures.

In this paper, it is introduced a document image binarization method that achieved very good results on four datasets containing images of historical documents affected by several kinds of degradations. Furthermore, it is investigated the application of a racing algorithm to tune the parameters of the proposed technique. The rest of this paper is organized as follows. In Section 2, some classical and recent binarization algorithms are reviewed. Section 3 reviews I/F-Race approach for parameter tuning. Section 4 presents the proposal of application of a racing algorithm to tune the parameters of a binarization technique, while in Section 5 the experiments performed are explained. Finally, Section 6 concludes the paper.

2. Document image binarization techniques

A survey of classical image thresholding algorithms is presented in (Sezgin & Sankur, 2004). In that work, the algorithms are categorized as (i) histogram shape-based, (ii) clustering-based, (iii) entropy-based, (iv) object attribute-based, (v) spatial or (vi) local methods. Thresholding algorithms can also be classified

as being parameter dependent or independent. For example, Otsu's (Otsu, 1979) clustering method, that defines an optimal threshold by minimizing the weighted sum of within-class variances and maximizing the between-class scatter, can be classified as parameter independent. On the other hand, as parameter dependent algorithms, there are classical local methods presented in (Niblack, 1986) and (Sauvola & Pietikainen, 2000), in which the image is divided into regions of size $b \times b$ and the threshold for each region is evaluated according to the local mean and standard deviation. In addition to the window size b, another parameter utilized by (Niblack, 1986) and (Sauvola & Pietikainen, 2000) is a bias k used in the calculation of the threshold.

Furthermore, it can be summarized more recent algorithms, like the ones proposed in (Howe, 2012; Su et al., 2013; Moghaddam et al., 2013). The method introduced in (Su et al., 2010) works by performing stroke width estimation, high contrast pixel detection and thresholding. It has two parameters that need to be properly set: (i) the minimum number of high contrast pixels that must be counted within a neighborhood window so that a given pixel can be classified as text or ink and (ii) the size of the respective neighborhood window. In (Su et al., 2013), the proposal in (Su et al., 2010) is extended by (i) combining local image contrast with the local image gradient, (ii) detecting text stroke edge pixel by combining Canny's algorithm with the image contrast binarized by Otsu's method and (iii) adding some post processing steps to achieve final binarization. In (Moghaddam et al., 2013), an unsupervised Ensemble of Experts (EoE) is introduced. Its main idea consists in selecting a set of appropriate binarization methods (experts) and combining their results. So, for each input, all methods are executed to generate a bi-level resultant image and a confidentness value for each classified pixel. Then, based on the confidentness maps, the set of experts that provides the better performance on the original image is identified and their binary results are combined into the final binary image. It is worth noting that EoE is also applicable to an ensemble of instances of the same technique with different parametric configurations (each parametric configuration for the given method is considered as a different expert); this is especially interesting for comparison reasons with the method proposed in this paper. In addition to the algorithms referred herein, in (Ntirogiannis et al., 2014), seven other binarization techniques, submitted to the Handwritten Document Image Binarization Contest 2014 (H-DIBCO 2014), are presented. The methods are based on a variety of concepts as: (i) stroke width and slant; (ii) function minimization and Canny edge detection; (iii) Fuzzy C-Means; (iv) Location Cluster Model; (v) phase analysis; and (vi) filtering and local statistics.

In the rest of this section, the approach presented in (Howe, 2012) is outlined. The reason for reviewing Howe's algorithm with more detail than the other recent binarization approaches is that it is used as part of our proposal and also used in a statistical comparison.

2.1. A laplacian energy based algorithm for document binarization

The work in (Howe, 2011) proposes an algorithm that uses a graph cut implementation (maximum flow) to find the minimum energy solution of an

objective function that combines the Laplacian operator (Gonzalez & Woods, 2010) and Canny edge detection (Canny, 1986). The objective function to be minimized is defined as

$$\varepsilon_I(B) = \sum_{i=0}^{m}\sum_{j=0}^{n}[L_{ij}^0(1-B_{ij}) + L_{ij}^1 B_{ij}] + \sum_{i=0}^{m-1}\sum_{j=0}^{n} C_{ij}^h(B_{ij} \neq B_{i+1,j}) + \sum_{i=0}^{m}\sum_{j=0}^{n-1} C_{ij}^v(B_{ij} \neq B_{i,j+1}), \tag{1}$$

where $B_{ij} \in 0, 1$ is the label value (background or foreground, respectively) of the pixel at position (i, j), L_{ij}^l is the cost to assign label l to the pixel at position (i, j), and C_{ij}^v and C_{ij}^h are the cost of a label mismatch between B_{ij} and its neighbour to the east or to the south, respectively (Howe, 2011).

The values of L_{ij}^0 and L_{ij}^1 are set according to the Laplacian of the image intensity, as follows:

$$L_{ij}^0 = \nabla^2 I_{ij} \tag{2}$$

$$L_{ij}^1 = \begin{cases} -\nabla^2 I_{ij}, & \textit{if } I_{ij} \leq \mu_{ij}^r + 2\sigma_{ij}^r \\ \phi, & \textit{if } I_{ij} > \mu_{ij}^r + 2\sigma_{ij}^r \end{cases}, \tag{3}$$

where $\nabla^2 I_{ij}$ represents the Laplacian operator applied to the intensity image I_{ij}. The use of the standard deviation σ_{ij}^r and of the local mean μ_{ij}^r computed, considering nearby pixels weighted by a Gaussian of radius r, ensures that only pixels most certain to be background are considered as such (Howe, 2012). ϕ is a sufficiently large negative value. According to the formulation in Eq. 1, C_{ij}^v and C_{ij}^h should be set in order that discontinuities are tolerated for edge pixels (that usually belongs to discontinuities between ink and background) and penalized for non-edge pixels. Thus, C_{ij}^v and C_{ij}^h are set according to the image edges E_{ij} detected by Canny's edge detector:

$$C_{ij}^h = \begin{cases} 0, & \textit{if } E_{ij} \bigwedge (I_{ij} < I_{i+1,j}) \\ 0, & \textit{if } E_{i+1,j} \bigwedge (I_{ij} \geq I_{i+1,j}) \\ c, & \textit{otherwise} \end{cases}, \tag{4}$$

$$C_{ij}^v = \begin{cases} 0, & \textit{if } E_{ij} \bigwedge (I_{ij} < I_{i,j+1}) \\ 0, & \textit{if } E_{i,j+1} \bigwedge (I_{ij} \geq I_{i,j+1}) \\ c, & \textit{otherwise} \end{cases} \tag{5}$$

In (Howe, 2012), an automatic method to set the two parameters (c, used to set the cost of label mismatches and t_{hi}, the higher threshold used in Canny's edge detector) that strongly influences the binarization result of the algorithm is proposed. The automatic choice of c and t_{hi} is based on a stability heuristic criterion under the hypothesis that good parameter values present low variability in the final binarization with respect to changes in parameters settings (Howe, 2012).

3. I/F-Race

In this Section, it is first introduced a definition of the problem of selecting a parametric configuration for a binarization algorithm based on the explanation in (Birattari et al., 2010). Then, in Sections 3.1 and 3.2, it is presented F-Race and I/F-Race, respectively.

The algorithm configuration (or tuning) problem is dealt as a generalization problem, in which it is aimed to determine promising parametric configurations based on a given set of instances during the training phase. Then, in the test phase, the algorithm is properly configured based on the results achieved in the training phase, and it is executed on a set of unseen instances. It is worth to mention that it is expected that the set of training instances is representative for the instances the algorithm faced in the test phase.

For the sake of clarity, it is assumed that the algorithm has N parameters, $X_d(d = 1, ..., N)$, that can have different values. The set of all possible configurations is denoted by Θ and each unique combination among all possible values for each parameter is called a configuration $\theta = (X_1, ..., X_N)$. Thus, our goal is to choose a configuration $\theta^* \in \Theta$ such that a given fitness function δ is minimized or maximized, considering the set of instances I.

3.1. F-Race

Racing algorithms can select in a fully automatic way a configuration for an algorithm from a given set of candidate configurations Θ. A racing algorithm works by sequentially processing a given set of instances I. Let i_l denote the l^{th} instance and let Θ_k be the set of candidate configurations at iteration k. Initially $\Theta_0 = \Theta$, and then, at iteration k of the race, all sampled candidate configurations in Θ_k are run once on instance i_l. When all results are available, the candidates in Θ_k that achieved statistically inferior results are eliminated, resulting in a possibly smaller set Θ_{k+1}. Then, all configurations in Θ_{k+1} run again on instance i_{l+1} and the same evaluation is repeated. This procedure is iterated until only a minimum number N_min of candidate configurations remains.

The proposal of Maron and Moore (Maron & Moore, 1997) inspired a diverse number of new algorithms which improved the search process of racing algorithms. Birattari et al. (Birattari et al., 2002) proposed an automatic configuration approach, named F-Race, that uses Friedman's non-parametric two-way analysis of variance by ranks. For giving a description of the test, let us assume that F-Race has reached step k, and $n = |\Theta_{k-1}|$ configurations are still in the race. The Friedman test assumes that the observed fitness are k mutually independent $n - variate$ random variables called blocks, where each block corresponds to the computational results of a chosen objective function δ on instance i_l for each configuration in the race at step k. Within each block the quantities $\delta(\theta, i_l)$ are ranked. For each configuration $\theta_j \in \Theta_{k-1}$, R_{lj} is the rank of θ_j within block l, $R_j = \sum_{l=1}^{k} R_{lj}$ is the sum of the ranks over all instances i_l, with $1 \leq l \leq k$. The

Friedman's test considers the following statistic (Conover, 1999):

$$T = \frac{(n-1)\sum_{j=1}^{n}\left(R_j - \frac{k(n+1)}{2}\right)^2}{\sum_{l=1}^{k}\sum_{j=1}^{n}R_{lj}^2 - \frac{kn(n+1)^2}{4}}. \quad (6)$$

Under the null hypothesis that all possible rankings of the candidates within each block are equally likely, T is approximatively χ^2 distributed with $n-1$ degrees of freedom. If the observed T exceeds the $1-\alpha$ quantile of such a distribution, the null hypothesis is rejected, at the approximate level α, in favour of the hypothesis that at least one candidate tends to yield a better performance than at least one another. In this case, it is justified to perform pairwise comparisons between the best and all other candidates. θ_j and θ_h candidates are considered different if

$$\frac{|R_j - R_h|}{\sqrt{\frac{2k\left(1-\frac{T}{k(n-1)}\right)\left(\sum_{l=1}^{k}\sum_{j=1}^{n}R_{lj}^2 - \frac{kn(n+1)^2}{4}\right)}{(k-1)(n-1)}}} > t_{1-\alpha/2}, \quad (7)$$

where $t_{1-\alpha/2}$ is the $1-\alpha/2$ quantile of the Student's t distribution (Conover, 1999). All candidates that result significantly worse than the best are discarded and will not appear in Θ_{k+1}. On the other hand, if at step k the null of the aggregate comparison is not rejected, all candidates in Θ_k pass to Θ_{k+1}.

3.2. I/F-Race

The Iterated F-Race (I/F-Race) improves F-Race by sampling configurations from the parameter space and refining the sampling distribution by means of repeated applications of F-Race. Thus, at each iteration, the configuration of the candidates from the previous iteration bias the sampling of the configurations of new candidates. By doing that, it is expected to focus the sampling of new candidate configurations around the most promising ones. Therefore, I/F-Race follows three main steps: (i) sampling new candidate configurations according to a particular distribution, (ii) selecting the best configurations from the newly sampled ones by means of racing, and (iii) updating the sampling distribution in order to bias the sampling towards the best configurations. These three steps are repeated until a termination criterion is reached. An outline of the I/F-Race algorithm is given in Algorithm 1.

Initially, a sample from the complete set of configurations is selected. After that, the best configurations are selected by means of racing. At each step of F-Race, the candidate configurations are evaluated on a single instance. After each step, those candidate configurations that perform statistically worse than at least another one are discarded, and the race continues with the remaining surviving configurations. Once F-Race ends, the candidates within the Θ^{elite}, those who survived the race, are then weighted according to their ranks. It was used N^{elite} to denote the number of candidates in Θ^{elite}. The weight of an elite configuration with rank $r_z, z = 1, ..., N^{elite}$, is given by:

Algorithm 1 I/F-Race pseudocode.

Parameters:

- $I = i_1, i_2, ...$
- Parameter space: Θ
- Fitness function: $\delta(\theta, i) \in \mathbb{R}$
- F-Race stop criterion: N_{min}
- I/F-Race stop criterion: Max_iter
- Number of configurations per iteration: N_conf

Steps:

$\Theta_0 = Sample(\Theta, N\_conf)$
$\Theta^{elite} = FRace(\Theta_0, N_{min}, \delta, I)$
$k = 1$
while $k \leqslant Max\_iter$ **do**
  $\Theta^{new} = Sample(\Theta, N\_conf - |\Theta^{elite}|, \Theta^{elite})$
  $\Theta_k = \Theta^{new} \cup \Theta^{elite}$
  $\Theta^{elite} = FRace(\Theta_k, N_{min}, \delta, I)$
  $k = k + 1$
end while
return Θ^{elite}

$$w_z = \frac{N^{elite} - r_z + 1}{N^{elite}(N^{elite} + 1)/2}. \tag{8}$$

In the next iteration of I/F-Race, $N\_conf - N^{elite}$ new candidate configurations are iteratively sampled around Θ^{elite}, where N_conf is a fixed number of candidates per iteration. To do so, for each new candidate configuration, one elite solution $\theta_z \in \Theta^{elite}$ is chosen with a probability proportional to its weight w_z and a value is sampled for each parameter. Herein, it is selected the value of each new parameter according to a probability density function $f(x, \mu, \sigma^2)$, where μ is equal to the parameter value of θ_z and $\sigma^2 = 1 - 2 \times k/10$. By doing this, as the number of iterations k executed increases, the standard deviation decreases and the sampling process gets more focused around the best configurations. This procedure continues until reaching a maximum number of iterations Max_iter.

In this work, it was implemented the most recent version of I/F-Race developed by López-Ibanez et al. in 2011 (López-Ibáñez et al., 2011) and recently revised in (Birattari et al., 2010). This version of I/F-Race was already extensively tested in several research projects. Dubois-Lacoste et al. in (Dubois-Lacoste et al., 2011a,b) used I/FRace for tuning the parameters of several iterated greedy (IG) variants for various objectives in the permutation flowshop problem (PFSP), outperforming the state-of-the-art. López-Ibañez and Stutzle in (López-Ibáñez & Stützle, 2010) automatically configured a flexible ant colony optimization framework for the bi-objective travelling salesman problem (bTSP). Montes de

Oca et al., in (de Oca et al., 2011), designed an incremental particle swarm optimization algorithm for large-scale continuous optimization problems by means of automatic configuration.

4. Proposed method

In (Mesquita et al., 2014), it is proposed an algorithm to binarize document images based on the perception of objects by distance. Its main idea, initially proposed in (Mello, 2010b), comes from the intuition that when one looks at document images that are far, the text tends to not be seen, while the background colors are still perceived. A simulation of this perception of objects by distance is created using morphological operations and image resizing. By the absolute difference between the original image and its simulated background it is possible to emphasize text regions and attenuate the background. In (Mesquita et al., 2014), the algorithm is improved by defining how far a document image should be from an observer so that text regions of the estimated stroke width could not be perceived anymore. This algorithm has two parameters that need to be properly configured. The first one is the minimum angle of resolution (MAR), referred as m_angle, that is the smallest target estimate in angular subtends that a person is capable of resolving. The MAR standard used for visual acuity test, that uses black characters in a white background, is 1' (one minute of arc). However, as the contrast between text and background tends to be less than it is in standard visual acuity tests, when ancient document images are dealt, the MAR value is usually higher (i.e, with a lower distance the target object is not perceived anymore by the human visual system). The second parameter is the difference between the radiuses of the disks used as structuring elements for the morphological operation of closing (used to simulate the visual system tendency to round corners when perceiving objects by distance), named as dif_rad. This process, referred as POD (Perception of Objects by Distance) is summarized by Algorithm 2. Further details can be found in (Mesquita et al., 2014).

The main objective of POD process is to increase the separation between ink and background pixels, making the classification into one class or the other easier. Thus, as im_eq is not a bi-level image, a final classification stage is required after applying POD. In (Mesquita et al., 2014) a combination between k-means and Otsu's thresholding algorithm is proposed as classification stage (this sequence is refereed as POD_KO). In the current work, it is proposed the application of Howe's algorithm as a classification stage after POD process, generating the algorithm named as POD_H. To both algorithms (POD_KO and POD_H), it is also proposed the use of I/F-Race to properly tune the parameters (named as m_angle and dif_rad) of POD. Thus, in Section 4.1 the parameter values used by I/F-Race are presented and in Sections 4.2 and 4.3 POD_KO and POD_H are detailed.

4.1. Parameter values and definitions used by I/F-Race

Parameter space Θ: The search space used by I/F-Race is defined as all possible configurations $\theta = (m\_angle, dif\_rad)$, where each parameter is

Algorithm 2 Perception of Objects by Distance (POD).

Parameters:

- Minimum Angle of Resolution (m_angle)
- Difference between radiuses structuring elements (dif_rad)

Steps:

1. Estimate the thickness ($thick$) of the strokes in the input image
2. Evaluate the distance (d_o) that will be simulated based on thick and on m_angle, so that the text is not perceived anymore
3. Determine the height (h_i) of the image formed on the retina based on the magnification equation
4. Apply closing morphological operation twice, achieving the image called im_mm
5. Downsize im_mm in order that its height equals to h_i, evaluating the width so that the image keeps its aspect ratio
6. Resize im_mm back to its original size, generating the image called $im\_background$
7. Evaluate the absolute difference between $im\_original$ and $im\_background$. The difference image is called im_diff
8. For every pixel i, if $im\_diff(i) = 0$, then $im\_diff(i)$ is converted into white (255)
9. For every pixel i, if $im\_diff(i) \neq 255$, then evaluate the complement of $im\_diff(i)$
10. Apply histogram equalization on im_diff, generating im_eq

restricted to the range $1, 2, ..., 30$. Thus, it has a number of $30 \times 30 = 900$ possible configurations.

N_min: F-Race has each iteration stopped when a minimum number $N\_min = 2 + round(log_2 d)$ of surviving candidate configurations is reached, where d is equal to the number of parameters to be optimized ($d = 2$, in our case) (Birattari et al., 2010).

δ: F-Measure (Ntirogiannis et al., 2013, 2014) is used as fitness function δ. It is defined as

$$FMeasure = \frac{2 \times Recall \times Precision}{Recall + Precision}, \tag{9}$$

where $Recall = \frac{TP}{TP + FN}$, $Precision = \frac{TP}{TP + FP}$ and TP, FP and FN denote the true positive, false positive and false negative values, respectively (Ntirogiannis et al., 2013). The reasons for choosing F-Measure as objective function are that (i) this measure has been widely used to evaluate binarization algorithms (as in (Gatos et al., 2009; Pratikakis et al., 2010, 2011, 2012, 2013)) and that (ii) in the experiments, it presented good results serving as a proxy for other measures (i.e., it was observed that maximizing F-Measure, the values of other measures were also improved, even though this was not guaranteed).

Max_iter: The number of iterations of I/F-Race is defined as $Max\_iter = 2 + round(log_2 d)$, where d is equal to the number of parameters, as suggested in (Birattari et al., 2010). It is important to highlight that the number of iterations to be executed depend on the number of parameters d to be optimized, as it is expected that, with more parameters involved, the problem to be optimized gets more difficult and, hence, more iterations are needed (Birattari et al., 2010).

N_conf: F-Race iteration is initialized with 30 candidate configurations. This value was defined experimentally in order to achieve satisfactory results in a considerable amount of time.

4.2. POD_KO

After the application of POD, it is still required a classification phase in which pixels are labeled as background or text. Thus, following the original approach proposed in (Mesquita et al., 2014), POD is followed by KO (POD_KO). It is important to mention that it was used the same sequence of POD followed by KO used in (Mesquita et al., 2014) so that it is possible to evaluate the parameter values suggested by I/F-Race comparing POD_KO with its original parameters suggested in (Mesquita et al., 2014) against the same algorithm with the parametric configurations suggested by I/F-Race.

The KO step works as follows: k-means algorithm with 3 classes ('text', 'unknown', and 'background') is applied to im_eq, generating the image $im\_k - means$, that contains only pixels that belong to 'text' class. Then, $im\_k - means$ is combined with the pixels classified as text by Otsu's method that are also located in text-line (Sanchez et al., 2011) regions of $im\_k - means$. The combination between two images 'A' and 'B' works by successively dilating 'A' conditioned to 'B'. This process is summarized by Algorithm 3.

Algorithm 3 Classification using a combination of K-means and Otsu algorithms (KO).

Steps:

1. Apply k-means clustering ($k = 3$) on im_eq, generating $im\_k - means$
2. Apply Otsu's algorithm on im_eq, generating im_otsu
3. Segment text lines in $im\_k - means$, generating im_lines
4. For each non text line pixel i in im_lines, $im\_otsu(i) = 255$
5. Final image = Combination between $im\_k - means$ and im_otsu

4.3. POD_H

To evaluate POD with the parameters suggested by I/F-Race as a pre-processing step to another binarization technique it is proposed the combination of POD followed by the algorithm presented in (Howe, 2012) (forming the sequence denominated as POD_H). The choice of Howe's algorithm has two main reasons: the first one is the good results achieved by this method in DIBCO'13 contest, and the second relies in the fact that the heuristic approach used by this algorithm to automatically choose the values of its parameters may fail in some unusual situations, as in the cases in which the choice made is optimal for only a part of the image (Howe, 2012). More specifically, this can happen, for example, in cases of smudges or uneven illumination. As the POD algorithm showed good performance with this cases (Mello, 2010b; Mesquita et al., 2014), it is expected that the combination POD_H can achieve promising results.

5. Experiments results and analysis

To perform experimental evaluation, the dataset from the document image binarization contest (DIBCO (Gatos et al., 2009; Pratikakis et al., 2010, 2011, 2012, 2013)) series is used, as it contains images with several kind of degradation, and the ground-truth and results obtained by the participant methods are publicly available. In addition, images from a set of manuscripts authored by the Brazilian politician Joaquim Nabuco from the end of the 19th Century, from ProHist project (Mello, 2010a) are also used. These images were digitized with 200 dpi resolution and stored in grayscale and in JPEG file format with 1% of loss for better preservation of the bequest.

Three different experiments are performed in order (i) to evaluate the parametric configurations suggested by I/F-Race to POD_H algorithm against the methods submitted do DIBCO 2012 and 2013 contests, (ii) to statistically compare POD_H against Howe's original algorithm and (iii) to statistically compare the parametric configurations suggested by I/F-Race for POD_KO against the parametric configuration suggested for this algorithm in (Mesquita et al., 2014). Thus, in the first experiment, it is used the dataset of images and all algorithms submitted to DIBCO 2012 and 2013. In the second and third experiments, it is used a larger dataset containing 66 images in order to statistically evaluate

the parametric configurations achieved using I/F-Race. All experiments were executed using MATLAB in a Intel Core i7 computer, at 2.10 GHz and with 8GB of RAM memory.

5.1. Evaluation measures

Following DIBCO'13 contest, in this paper, four evaluation measures are used: F-Measure, Peak Signal-to-Noise Ratio (PSNR), Distance Reciprocal Distortion (DRD) and pseudo-FMeasure. PSNR, that is a measure of how close one image is to another, is evaluated as

$$PSNR = 10log(\frac{C^2}{MSE}), \tag{10}$$

where MSE is the Mean Square Error and C is equal to the difference between the foreground and the background values (Pratikakis et al., 2013).

DRD (Lu et al., 2004) is an objective distortion measure specific for binary document images. Its main idea is that the distance between two pixels strongly influences their mutual interference perceived by the human visual system. Thus, DRD is evaluated as

$$DRD = \frac{\sum_{k=1}^{S} DRD_k}{NUBN}, \tag{11}$$

where NUBN is the number of non-uniform (not all black or white pixels) 8×8 blocks in the ground truth image. DRD_k represents the distortion of the k^{th} flipped pixel and it is defined as

$$DRD_k = \sum_{i=-2}^{2} \sum_{j=-2}^{2} |GT_k(i,j) - B_k(i,j)| \times W_{Nm}(i,j). \tag{12}$$

As one can see, the difference between each pair of pixels in Eq. 12 is weighted by a normalized matrix $W_{Nm}(i,j)$ that herein is used as proposed in (Lu et al., 2004). The lower the DRD value, the better the result achieved.

Finally, Pseudo-FMeasure (Ntirogiannis et al., 2013) is a pixel-based binarization methodology that replaces *Recall* and *Precision* terms used in Eq. 9 by pseudo-Recall (R_{ps}) and pseudo-Precision (P_{ps}), respectively. The idea of pseudo-Recall is based on the fact that false negatives pixels near to the contour of text pixels should be less penalized than false negative pixels that are closer to the skeleton (Gonzalez & Woods, 2010) of the ground-truth text. Thus, pseudo-Recall is defined as

$$R_{ps} = \frac{\sum_{x=1,y=1}^{x=l_x,y=l_y} B(x,y) \times G_w(x,y)}{\sum_{x=1,y=1}^{x=l_x,y=l_y} G_w(x,y)}, \tag{13}$$

where $B(x,y)$ is the binarized image and $G_w(x,y)$ is the weighted ground truth image, as defined in (Ntirogiannis et al., 2013). Similarly, pseudo-Precision (P_{ps}) weights the background ground truth image based on the distance of each pixel to the contour of text regions and it is evaluated as

$$P_{ps} = \frac{\sum_{x=1,y=1}^{x=l_x,y=l_y} G(x,y) \times B(x,y)}{\sum_{x=1,y=1}^{x=l_x,y=l_y} B_w(x,y)}, \tag{14}$$

with $B_w(x,y) = B(x,y) \times P_w(x,y)$ as the binarized image after the application of a weighted map $P_w(x,y)$, as defined in (Ntirogiannis et al., 2013).

5.2. Evaluation of POD_H using DIBCO 2012 and 2013 datasets

In order to evaluate POD_H using the images and methods from DIBCO 2012 and 2013, it was applied I/F-Race to tune the parameters used in the Perception of Objects by Distance step. In the evaluation using the dataset from H-DIBCO'12, it was used the images from H-DIBCO'10 as training set, as both datasets contain only handwritten document images. Based on the same idea, DIBCO'11 images are used as training set for the test performed using DIBCO'13 images, as both datasets contain handwritten and typewritten document images. For each training set I/F-Race runs 10 times and, for each execution, a set of elite configurations is found. For each set of elite configurations the best one is selected to be used in the test phase. Thus, there are 10 configurations to evaluate each algorithm using DIBCO'12 dataset and 10 (not necessarily) different configurations to evaluate each algorithm using DIBCO'13 dataset.

DIBCO contest traditionally ranks the submitted algorithms by sorting the accumulated ranking value of all measures for each test image, as explained in (Pratikakis et al., 2012). However, accumulating individual ranking values for each measure in a given input image is not a good approach, as ranking values can only inform us that a given method A had a superior performance than another method B, but it does not reflect the quantitative difference between both performances. For example, a given method can occasionally be placed at the last rank position and obtain very similar result if compared to a method placed at first position, but it will also receive a high penalization. On the other hand, considering another test image, a method placed at second position receives a lower penalization even if it achieves much worse results when compared with the method placed at first position. This problem is also cited in (Howe, 2012), and explained in (Mesquita et al., 2014). Thus, it is used herein the ranking approach proposed in (Mesquita et al., 2014) that is based on the summation of the results of each method m achieved among all test images i, for each evaluation measure. Then, the results obtained are normalized by all methods for each measure to ensure equal weights among all evaluation measures:

$$fm\_norm(m) = \frac{(fmeasure(m) - min(fmeasure))}{max(fmeasure) - min(fmeasure)}, \tag{15}$$

$$pfm\_norm(m) = \frac{(pfmeasure(m)) - min(pfmeasure)}{max(pfmeasure) - min(pfmeasure)}, \tag{16}$$

$$psnr\_norm(m) = \frac{(psnr(m)) - min(psnr)}{max(psnr) - min(psnr)}, \tag{17}$$

$$drd\_norm(m) = 1 - \frac{(drd(m)) - min(drd)}{max(drd) - min(drd)}, \quad (18)$$

where $fmeasure(m)$, $pfmeasure(m)$, $psnr(m)$ and $drd(m)$ denote the sum of the results achieved by the method m considering all test images and $min(z)$ and $max(z)$ denote the minimum and maximum sum of results obtained among all methods for a given evaluation measure z. The subtraction by one is necessary in Eq. 18 as low scores obtained with DRD represent better results.

To evaluate our algorithms, the ranking approach explained above and the four evaluation measures described in Section 5.1 are used. Table 1 shows fm_norm, pfm_norm, $psnr\_norm$, and drd_norm values obtained by POD_H configured with $(m\_angle, dif\_rad) = (22, 12)$ and by some of the best methods submitted to DIBCO'13 using DIBCO'13 images as test set. This configuration achieved the best result among the 10 configurations tested. Table 2 shows the results of the worst configuration tested, that was the only one of the ten configurations that did not achieve the first place. Moreover, all configurations tested showed the best fm_norm values, even the worst one, as one can see in the second column of Table 2, indicating the effectiveness of I/F-Race procedure to find configurations that achieve good F-Measure rates. In addition, the other measures also have showed very good results. Its worth mentioning that the methods in (Moghaddam et al., 2013) and (Su et al., 2013) that were reviewed in Section 2 are represented in our experiments as *dibco5* and *dibco15b*, respectively, following the same notation used in (Pratikakis et al., 2013). It is also important to state that *dibco5* corresponds to the EoE framework (Moghaddam et al., 2013) applied to (Howe, 2012), in which each expert is an instance of the algorithm proposed in (Howe, 2012) configured with different parametric configurations.

Table 1: Results obtained by the best POD_H configuration suggested by I/F-Race using DIBCO'11 as training set and DIBCO'13 as test set

Method	**fm_norm**	**pfm_norm**	**psnr_norm**	**drd_norm**	**sum**	**rank**
POD_H(22,12)	1	1	1	1	4	1
howe	0.9208	0.9126	0.9911	0.9945	3.8189	2
dibco15b	0.9133	0.9405	0.9136	0.9881	3.7555	3
dibco17	0.9439	0.9672	0.8461	0.9880	3.7452	4
dibco5	0.9018	0.9090	0.9267	0.9745	3.7120	6
dibco1	0	0	0	0	0	24

Similar experiments are performed using DIBCO'12 dataset. Tables 3 and 4 show the best and worst results (respectively) achieved among all 10 configurations tested. In this experiment, POD_H achieved first place in 5 configurations, while Howe's algorithm without POD step was placed in the first place in the other 5 times. However, similarly to what happened in the tests using DIBCO'13 dataset, all configurations of POD_H achieved the best F-Measures values.

Table 2: Results obtained by the worst POD_H configuration suggested by I/F-Race using DIBCO'11 as training set and DIBCO'13 as test set

Method	**fm_norm**	**pfm_norm**	**psnr_norm**	**drd_norm**	**sum**	**rank**
howe	0.9747	0.9435	1	1	3.9182	1
POD_H(25,24)	1	0.9586	0.9164	0.9879	3.8629	2
dibco15b	0.9667	0.9724	0.9219	0.9936	3.8545	3
dibco17	0.9991	1	0.8537	0.9935	3.8463	4
dibco5	0.9545	0.9398	0.9350	0.9799	3.8093	6
dibco1	0	0	0	0	0	24

Table 3: Results obtained by the best POD_H configuration suggested by I/F-Race using DIBCO'10 as training set and DIBCO'12 as test set

Method	**fm_norm**	**pfm_norm**	**psnr_norm**	**drd_norm**	**sum**	**rank**
POD_H(12,16)	1	1	1	1	4	1
howe	0.9026	0.9139	0.9700	0.9795	3.7659	2
dibco11	0.8776	0.8770	0.8245	0.9617	3.5409	3
dibco4a	0.8040	0.9005	0.7629	0.9445	3.4119	4
dibco12	0.8566	0.8319	0.7680	0.9627	2.8405	13
dibco17	0	0	0	0	0	26

Table 4: Results obtained by the worst POD_H configuration suggested by I/F-Race using DIBCO'10 as training set and DIBCO'12 as test set

Method	**fm_norm**	**pfm_norm**	**psnr_norm**	**drd_norm**	**sum**	**rank**
howe	0.9808	0.9393	1	1	3.9202	1
POD_H(26,24)	1	0.9402	0.9264	0.9868	3.8535	2
dibco11	0.9537	0.9015	0.8500	0.9819	3.6871	3
dibco4a	0.8737	0.9256	0.7865	0.9643	3.5501	4
dibco12	0.7216	0.8013	0.5029	0.9283	2.9540	13
dibco17	0	0	0	0	0	26

5.3. Statistical comparison using POD_H

In this experiment, it is compared the configurations suggested by I/F-Race to POD_H against Howe's algorithm. Thus, to perform a statistical evaluation, a dataset larger than the ones used in the previous experiments was created, containing 66 document images. This dataset is composed by 10 images from DIBCO'09, 10 images from DIBCO'10, 14 images from DIBCO'12, 16 images from DIBCO'13 and 16 images from ProHist project. It is important to mention that the images were selected from ProHist file in order to represent several kinds of degradation, like smudges, bleed-through and different illumination levels. As DIBCO'11 dataset demonstrated to be more appropriate to find promising parametric configurations than DIBCO'12 dataset, based on the results from the previous experiments and also based on the greater variability of degradation effects that this dataset contains, it was not included any image from DIBCO'11 in the test dataset. The worst configuration $(m\_angle, dif\_rad) = (25, 24)$ achieved from the training phase performed with DIBCO'11 data set was used

to perform the statistical evaluation described in this Section.

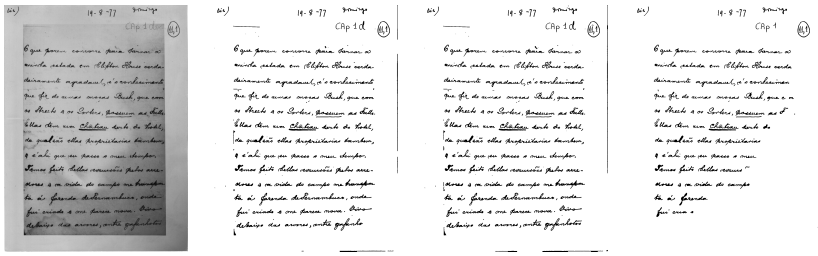
As the algorithms are deterministic, it was used a Monte Carlos's simulation to perform the comparison. Thereby, 100 samples of 30 images were randomly selected without reposition from the entire dataset and the F-Measure rates of both algorithms are evaluated on each sample. Then, as the null hypothesis that the F-Measure rates from both algorithms follow a normal distribution could not be rejected ($p - value > 0.5$) , it is executed a paired Student's t-test and concluded with 99% of confidence ($p - value = 3.14 \times 10^{-6}$) that POD_H algorithm achieves better results in terms of F-Measure than Howe's algorithm, considering the tested dataset. Figure 2 shows examples of results obtained by both algorithms.

It is worth reporting the performance evaluation of the application of I/F-race to tune the parameters of POD_H using DIBCO'11 as training set. The average number of configurations tested was 85.5 for each one of the 10 iterations. As there are an amount of 900 possible different configurations, it is needed to explore only 9.5% of the search space. The average time to execute each iteration of I/F-race is approximately 43 minutes (it is important to mention that it was stored the results of each configuration with each image so that none of them was executed more than once per image, considering all the 10 iterations). The average execution time of POD_H is 13 seconds considering an average image dimension of $574 \times 1,103$ pixels.

5.4. Statistical comparison using POD_KO

In this experiment, it is compared the configurations found by I/F-Race to tune the parameters $((m\_angle, dif\_rad))$ of POD_KO against the original parametric configuration $((m\_angle, dif\_rad) = (3, 2))$ proposed in (Mesquita et al., 2014).

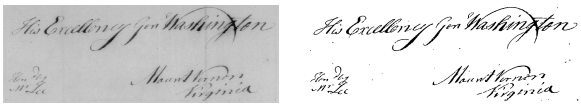
Initially, as in Section 5.2, 10 executions of I/F-Race were applied using DIBCO'11 as a training dataset and the best result achieved in each execution was used in DIBCO'13 images. Tables 5 and 6 illustrate the worst and best scenarios. As it can be seen, in both cases, the parametric configuration suggested by I/F-Race achieves better results for all evaluation measures. It is also important to mention that all the 8 other configurations achieved the 9^{th} place. It was also performed the same statistical evaluation executed in Section 5.3 to compare $POD\_KO(15, 6)$, that is the worst configuration achieved in terms of F-Measure, against $POD\_KO(3, 2)$. As the null hypothesis that the F-Measure rates from both configurations follow a normal distribution could not be rejected ($p - value > 0.5$), it was executed a paired Student's t-test and concluded with 99% of confidence ($p - value = 1.5 \times 10^{-7}$) that POD_KO algorithm with the worst configuration suggested by I/F-Race achieves better results in terms of F-Measure if compared to the parametric configuration suggested in (Mesquita et al., 2014).



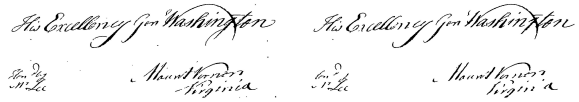
(a) original (b) pod_h(25,24) (c) pod_h(27,11) (d) Howe

(e) original (f) pod_h(25,24)

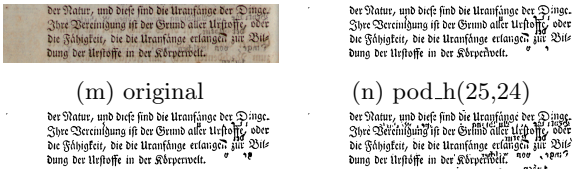
(g) pod_h(27,11) (h) Howe



(i) original (j) pod_h(25,24)

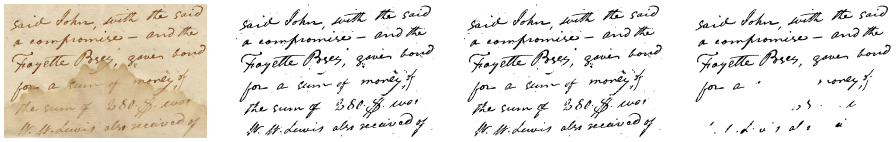


(k) pod_h(27,11) (l) Howe



(m) original (n) pod_h(25,24)

(o) pod_h(27,11) (p) Howe



(q) original (r) pod_h(25,24) (s) pod_h(27,11) (t) Howe

Figure 2: Example of results

Table 5: Results obtained by the worst POD_KO configuration suggested by I/F-Race using DIBCO'11 as training set and DIBCO'13 as test set

Method	**fm_norm**	**pfm_norm**	**psnr_norm**	**drd_norm**	**sum**	**rank**
dibco3	0.9755	0.9435	1	1	3.9190	1
POD_KO(7,6)	0.8657	0.9125	0.7230	0.9586	3.4599	10
POD_KO(3,2)	0.8455	0.8984	0.6994	0.9506	3.3939	14
dibco14	0.6040	0.7503	0.4689	0.8336	2.6568	24
dibco1	0	0	0	0	0	25

Table 6: Results obtained by the best POD_KO configuration suggested by I/F-Race using DIBCO'11 as training set and DIBCO'13 as test set

Method	**fm_norm**	**pfm_norm**	**psnr_norm**	**drd_norm**	**sum**	**rank**
dibco3	0.9755	0.9435	1	1	3.9190	1
POD_KO(29,6)	0.8908	0.9430	0.7488	0.9715	3.5540	9
POD_KO(3,2)	0.8455	0.8984	0.6994	0.9506	3.3939	14
dibco14	0.6040	0.7503	0.4689	0.8336	2.6568	24
dibco1	0	0	0	0	0	25

6. Conclusions

In this work, it was proposed the application of I/F-Race to tune the parameters of a binarization algorithm. Our experiments showed that using the parametric configurations suggested by I/F-Race the performance of POD_KO was increased for each one of the four usual evaluation measures considered. For the best of the knowledge of the authors, there is no other method that applies a racing algorithm for thresholding purposes.

Another contribution of this work is the application of POD (with the parameters suggested by I/F-Race) as a pre-processing step to Howe's algorithm. This process, named POD_H, outperformed the participants of DIBCO'13 and DIBCO'12 in terms of F-Measure with all configurations suggested by I/F-Race. Furthermore, POD_H was tried in the H-DIBCO 2014 (Ntirogiannis et al., 2014) and it ranked the first place. The practical advantage of using POD as a pre-processing step is the fact that it simplifies the classification phase and the benefit of applying it before Howe's algorithm can be seen especially in the case of images affected by uneven illumination or smudges, as illustrated in Figure 2. This improvement was statistically demonstrated by a comparison between Howe's algorithm and POD_H using a larger dataset than the ones used in DIBCO 2012 and 2013.

Among the reasons of the good results obtained it can be highlighted the efficient exploration of the search space realized by I/F-Race. Besides considering all possible parameter combinations, the mechanism employed by I/F-Race also makes possible the refinement of the search among the most promising configurations while the non-promising candidates are discarded. The benefits of this approach are clear when it is compared with the strategy employed in the *dibco5* entry, that uses the framework proposed in (Moghaddam et al.,

2013) with different parametric configurations of Howe's algorithm. Despite the fact that, in (Moghaddam et al., 2013), different configurations can be used for different inputs, there is not a guarantee that the best ones are considered, as the configurations are selected manually. Another strength of our approach is that in the test phase just one configuration (the one selected as the most promising) needs to be executed while in (Moghaddam et al., 2013) it is required to run all considered configurations for each input image.

The disadvantage of the proposed method is the need of a training phase, in which more than one configuration has to be executed for each image in the dataset (this problem is mitigated by the fact that, following the search mechanism adopted, just a small part of the total configurations are executed). Another point related to the training phase is that the dataset used in this step needs to be sufficiently representative in relation to the problem in question. In the case of binarization, this means that the dataset used in the test phase should contain images with different kinds and levels of degradations. If compared to the framework proposed in (Moghaddam et al., 2013), the method proposed herein also has the disadvantage of just selecting one possible configuration instead of maintaining the possibility of using different configurations for different problems.

As future research possibilities, it is suggested the adaptation of I/F-Race to select parameters considering the binarization result as a multi-objective function, so that more evaluation measures, such as DRD, PSNR and pseudo-FMeasure, are considered in addition to FMeasure. Other suggestions are (i) the application of I/F-Race to tune binarization algorithms considering only a specific kind of degradation, (ii) the classification of each input image according to the kind and level of degradation, so that it would be possible to have different parametric configurations for each input image, and (iii) the investigation of the use of POD as preprocessing stage for other binarization algorithms.

Acknowledgments:

This work was partially sponsored by the Brazilian National Council for Scientific and Technological Development (CNPq) under grants 141190/2013-2 and 141000/2013-9. The research of R.M.A Silva was also partially supported by CNPq.

References

Baird, H. (2003). Digital libraries and document image analysis. In *Proceedings of the Second International Conference on Document Analysis and Recognition.* (pp. 2–14). IEEE Comput. Soc volume 1.

Becker, S., Gottlieb, J., & Stützle, T. (2006). Applications of racing algorithms: An industrial perspective. In *Proceedings of the 7th International Conference on Artificial Evolution* EA'05 (pp. 271–283). Berlin, Heidelberg: Springer-Verlag.

Birattari, M., Stutzle, T., Paquete, L., & Varrentrapp, K. (2002). A Racing Algorithm for Configuring Metaheuristics. In *GECCO '02 Proceedings of the Genetic and Evolutionary Computation Conference2* (pp. 11–18).

Birattari, M., Yuan, Z., Balaprakash, P., & Stutzle, T. (2010). F-Race and Iterated F-Race: An Overview. In *Experimental Methods for the Analysis of Optimization Algorithms* June (pp. 311–336).

Canny, J. (1986). A computational approach to edge detection. *IEEE transactions on pattern analysis and machine intelligence*, *8*, 679–698.

Cheriet, M., Kharma, N., Liu, C.-L., & Suen, C. (2007). *Character Recognition Systems: A Guide for Students and Practitioners*.

Conover, W. (1999). *Practical nonparametric statistics*. Wiley series in probability and statistics (3rd ed.). New York, NY [u.a.]: Wiley.

Dubois-Lacoste, J., López-Ibáñez, M., & Stützle, T. (2011a). A hybrid tp+pls algorithm for bi-objective flow-shop scheduling problems. *Comput. Oper. Res.*, *38*, 1219–1236.

Dubois-Lacoste, J., López-Ibáñez, M., & Stützle, T. (2011b). Improving the anytime behavior of two-phase local search. *Ann. Math. Artif. Intell.*, *61*, 125–154.

Gatos, B., Ntirogiannis, K., & Pratikakis, I. (2009). ICDAR 2009 Document Image Binarization Contest (DIBCO 2009). In *2009 10th International Conference on Document Analysis and Recognition* (pp. 1375–1382).

Gonzalez, R. C., & Woods, R. E. (2010). *Digital Image Processing*. (3rd ed.).

Howe, N. R. (2011). A Laplacian Energy for Document Binarization. In *2011 International Conference on Document Analysis and Recognition* (pp. 6–10).

Howe, N. R. (2012). Document binarization with automatic parameter tuning. *International Journal on Document Analysis and Recognition (IJDAR)*, *16*, 247–258.

Kennedy, J., & Eberhart, R. C. (1995). Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks* (pp. 1942–1948).

Lacerda, E. B., & Mello, C. A. B. (2013). Segmentation of connected handwritten digits using Self-Organizing Maps. *Expert Systems with Applications*, (pp. 5867–5877).

Lin, C., Choy, K. L., Ho, G. T. S., & Ng, T. W. (2014). A Genetic Algorithm-based optimization model for supporting green transportation operations. *Expert Systems with Applications*, *41*, 3284–3296.

Lin, C.-j. (2010). Computing shadow prices/costs of degenerate LP problems with reduced simplex tables. *Expert Systems With Applications*, *37*, 5848–5855.

López-Ibáñez, M., & Stützle, T. (2010). Automatic configuration of multi-objective aco algorithms. In *Proceedings of the 7th International Conference on Swarm Intelligence* ANTS'10 (pp. 95–106). Berlin, Heidelberg: Springer-Verlag.

López-Ibáñez, M., Dubois-Lacoste, J., Stützle, T., & Birattari, M. (2011). *The irace package: Iterated Racing for Automatic Algorithm Configuration*. Technical Report TR/IRIDIA/2011-004 IRIDIA, Université Libre de Bruxelles, Belgium.

Lu, H., Kot, A. C., & Shi, Y. Q. (2004). Distance-Reciprocal Distortion Measure for Binary Document Images. *IEEE Signal Processing Letters*, *11*, 228–231.

Maron, O. (1994). *Hoeffding Races: model selection for MRI classification*. Master's thesis The Massachusetts Institute of Technology.

Maron, O., & Moore, A. W. (1997). The racing algorithm: Model selection for lazy learners. *Artificial Intelligence Review*, *11*, 193–225.

Mascaro, A. A., Cavalcanti, G. D., & A.B. Mello, C. (2010). Fast and robust skew estimation of scanned documents through background area information. *Pattern Recognition Letters*, *31*, 1403–1411.

Mello, C. A. B. (2010a). Prohist project. Accessed on 18th March 2014.

Mello, C. A. B. (2010b). Segmentation of Images of Stained Papers Based on Distance Perception. In *IEEE International Conference on Systems Man and Cybernetics* (pp. 1636 – 1642). Istanbul, Turkey.

de Mello, C. A. B., de Oliveira, A. L. I., & dos Santos, W. P. (2012). *Digital Document Analysis and Processing*. (1st ed.).

Mesquita, R. G., Mello, C. A. B., & Almeida, L. H. E. V. (2014). A new thresholding algorithm for document images based on the perception of objects by distance. *Integrated Computer-Aided Engineering*, *21*, 133–146.

Moghaddam, R. F., Moghaddam, F. F., & Cheriet, M. (2013). Unsupervised ensemble of experts (EoE) framework for automatic binarization of document images. In *2013 12th International Conference on Document Analysis and Recognition* (pp. 703–707).

Niblack, W. (1986). *An Introduction to Image Processing*. Pretince Hall, Englewood Cliffs, NJ.

Ntirogiannis, K., Gatos, B., & Pratikakis, I. (2013). Performance evaluation methodology for historical document image binarization. *IEEE transactions on image processing*, *22*, 595–609.

Ntirogiannis, K., Gatos, B., & Pratikakis, I. (2014). ICFHR 2014 Competition on Handwritten Document Image Binarization (H-DIBCO 2014). In *2014 14th International Conference on Frontiers in Handwriting Recognition* (pp. 809–813).

de Oca, M. A. M., Aydin, D., & Stützle, T. (2011). An incremental particle swarm for large-scale continuous optimization problems: an example of tuning-in-the-loop (re)design of optimization algorithms. *Soft Comput.*, *15*, 2233–2255.

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. In *IEEE Transactions on Systems Man and Cybernetics* (pp. 62–66). volume 20.

Pellegrini, P. (2005). *Application of Two Nearest Neighbor Approaches to a Rich Vehicle Routing Problem*. Technical Report TR/IRIDIA/2005-015 IRIDIA, Université Libre de Bruxelles.

Pratikakis, I., Gatos, B., & Ntirogiannis, K. (2010). H-DIBCO 2010 - Handwritten Document Image Binarization Competition. In *2010 12th International Conference on Frontiers in Handwriting Recognition* (pp. 727–732).

Pratikakis, I., Gatos, B., & Ntirogiannis, K. (2011). ICDAR 2011 Document Image Binarization Contest (DIBCO 2011). In *2011 International Conference on Document Analysis and Recognition* (pp. 1506–1510). Beijing.

Pratikakis, I., Gatos, B., & Ntirogiannis, K. (2012). ICFHR 2012 Competition on Handwritten Document Image Binarization (H-DIBCO 2012). In *2012 International Conference on Frontiers in Handwriting Recognition* (pp. 817–822).

Pratikakis, I., Gatos, B., & Ntirogiannis, K. (2013). ICDAR 2013 Document Image Binarization Contest (DIBCO 2013). In *2013 International Conference on Document Analysis and Recognition* (pp. 1102–1107).

Sanchez, A., Mello, C. A. B., Suarez, P., & Lopes, A. (2011). Automatic line and word segmentation applied to densely line-skewed historical handwritten document images. *Integrated Computer-Aided Engineering*, *18*, 125–142.

Sauvola, J., & Pietikainen, M. (2000). Adaptive document image binarization. *Pattern Recognition*, *33*, 225–236.

Sezgin, M., & Sankur, B. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, *13*, 146–168.

Snellen, A. J., & Harper, R. H. R. (2002). *The Myth of the Paperless Office*. Cambridge, MA.

Su, B., Lu, S., & Tan, C. L. (2010). Binarization of Historical Document Images Using the Local Maximum and Minimum. In *Workshop on Document Analysis Systems (DAS)* (pp. 159–166). Boston, USA.

Su, B., Lu, S., & Tan, C. L. (2013). Robust document image binarization technique for degraded document images. *IEEE Transactions on Image Processing*, *22*, 1408–1417.

Wang, Y., Hao, J.-K., Glover, F., & Lü, Z. (2014). A tabu search based memetic algorithm for the maximum diversity problem. *Eng. Appl. of AI*, *27*, 103–114.

Zhang, X., Chen, X., & He, Z. (2010). An ACO-based algorithm for parameter optimization of support vector machines. *Expert Systems with Applications*, *37*, 6618–6628.