Pattern Recognition 43 (2010) 3243–3254

Contents lists available at ScienceDirect

Pattern Recognition

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

Transition thresholds and transition operators for binarization and edge detection

Marte A. Ramírez-Ortegón [a,*], Ernesto Tapia [a], Raúl Rojas [a], Erik Cuevas [b]

[a] Institut für Informatik, Freie Universität Berlin, Takustr. 9, 14195 Berlin, Germany
[b] Department of Computer Science, University of Guadalajara, Av. Revolución 1500, Guadalajara, Jalisco, Mexico

ARTICLE INFO

Article history:
Received 12 November 2009
Received in revised form
26 February 2010
Accepted 10 April 2010

Keywords:
Transition function
Thresholding
Binarization
Edge detection
Image pre-processing

ABSTRACT

The transition method for image binarization is based on the concept of *t-transition pixels*, a generalization of *edge pixels*, and *t-transition sets*. We introduce a novel unsupervised thresholding for unimodal histograms to estimate the transition sets. We also present *dilation* and *incidence* transition operators to refine the transition set. Afterward, we propose the *simple edge* transition operator for detecting edges. Our experiments show that the new approach increases the effectiveness of OCR applications outperforming several top-ranked binarization algorithms.

1. Introduction

In binarization algorithms, the *foreground* $\mathcal{F}$ denotes the subset of pixel of interest, while the *background* $\mathcal{B}$ is the complement of $\mathcal{F}$.

Few binarization researchers work with color images directly [1]. The rest transforms an image from color to gray intensities before applying the binarization algorithm. For example: (1) Chou et al. [2] developed a binarization system for images produced by cameras, which deals with uneven illuminated images. They divide a gray-intensity image into several regions and decide how to binarize each region. (2) Caron et al. [3] detect region of interest characterizing each pixel with a gray-intensity template of 3×3, the frequency of which appears to obey a power law distribution. (3) Milewski et al. [4] presented a methodology for separating handwritten foreground pixels, from background in carbon copied medical forms. They compare the gray-intensity mean of small neighborhoods around the pixel of interest. (4) and (5) Both Chen et al. [5] and Mello et al. [6] binarize documents using gray-intensity images as input. Whereas Chen et al. generate the binary image from the edge image of the gray-intensity image, Mello et al. compute a threshold based on a weighted entropy equation.

We follow the approach of the previous examples. That is, our method takes a gray-intensity image I as input and returns a binary image B as output, where pixels in white represent the background and pixels in black represent the foreground.

We can identify three categories of binarization algorithms [7]. *Global algorithms* label a given pixel using information from the whole image, while *local algorithms* rely on information from the pixel neighborhood. *Hybrid algorithms* combine information from the whole image and from the pixel neighborhood. Note that global algorithms can be transformed into local versions by restricting the analysis to the pixel neighborhood. All algorithms considered in this article are local even though some of them were originally formulated for global analysis. Local versions usually perform better than global ones [7].

Thresholding algorithms classify a pixel as foreground if the gray intensity of the pixel is darker than a threshold. In this line, Sezgin and Sankur [8] present an exhaustive categorization of thresholding. They affirm that *Kittler's minimum error thresholding* [9] and *Sauvola's thresholding* [10] are the best-scored algorithms binarizing documents uniformly illuminated and degraded with noise and blur. In essence, Kittler's criterion maximizes the likelihood of the joint distribution of gray intensities assuming that foreground and background are normally distributed with different variances. *Otsu's algorithm* [11] minimizes the gray-intensity variances of both foreground and background. In contrast, *Portes et al.* [12] maximizes the *non-extensive entropy*, also called *Tsallis entropy* [13], of both foreground and background.

Sauvola's algorithm is a modified version of *Niblack's algorithm* [14]. Both algorithms assume that the gray intensities of the background are approximately normally distributed and select a

* Corresponding author.
E-mail addresses: mars.sasha@gmail.com, mars_sasha@hotmail.com (M.A. Ramírez-Ortegón), tapia@inf.fu-berlin.de (E. Tapia), rojas@inf.fu-berlin.de (R. Rojas), erik.cuevas@cucei.udg.mx (E. Cuevas).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.04.028

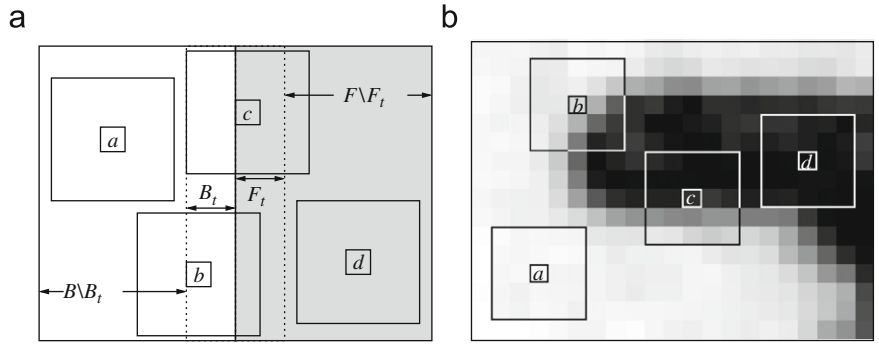

Fig. 1. The schemes exemplify two no-transition pixels $\boldsymbol{a}$ and $\boldsymbol{d}$, a negative transition pixel $\boldsymbol{b}$ and a positive transition pixel $\boldsymbol{c}$. We expect that $V(\boldsymbol{a}) \approx 0 \approx V(\boldsymbol{d})$, $V(\boldsymbol{b}) \leq -t_-$ and $V(\boldsymbol{c}) \geq t_+$, where t_- and t_+ are approximately equal to the expected gray-intensity contrast between foreground and background pixels.

threshold as the lower limit of an interval centered in the mean of the gray-intensity histogram. Wolf and Jolion [15] modified Sauvola's equation by adding two terms: the minimum gray intensity of the neighborhood and the maximum standard deviation of all neighborhoods. Both act like a dynamic variance-normalization factors.

The key ideas in the methods mentioned above are the following: Niblack, Sauvola and Wolf *use statistics from a neighborhood of the pixel of interest*, while Kittler, Portes and Otsu define *a criterion to optimize* and assume an *a priori statistical distribution for both foreground and background*. However, all six algorithms disregard edge information and this leads to false positive spots and diminished foreground boundaries.

We attempt to extend the transition method proposed in [16]. The concept of *t-transition pixel*, *t-transition set*, *transition value*, and *transition function* are briefly explained. Later on, we discuss the histogram behavior of transition values. Some mathematical arguments and empirical evidence lead us to propose the *double linear thresholding* to compute a threshold for transition values in order to approximate the transition set, which is refined by both *incidence* and *dilation* transition operators.

Although edge detection is not the main scope of this article, we introduce the *simple edge* transition operator that is capable to compute the edge image.

This paper considers only one input image without sudden illumination changes in small neighborhoods.

The rest of this paper is organized as follows. Section 2 introduces the main concepts and introduces the concept of transition. The transition method is developed in Section 3. Section 4 describes the design of our experiments. The results are shown in Section 5. Conclusions are presented in Section 6.

2. Preliminary concepts

2.1. Notation

An image function F can be defined as the mapping $F : \mathcal{P} \to \mathcal{A}$ where $\mathcal{P} = \{\boldsymbol{p}_{i,j} | \boldsymbol{p}_{i,j} \in \mathbb{N} \times \mathbb{N}\}$ and $\mathcal{A} \subset \mathbb{Z}$. The pixel $\boldsymbol{p}_{i,j}$ is located in the i-th row and the j-th column. If the pixel position is irrelevant, we denote the pixel $\boldsymbol{p}_{i,j}$ as $\boldsymbol{p}$. $I : \mathcal{P} \to \mathcal{G} = \{0,1,\ldots,l\}$ is the gray image function. $B : \mathcal{P} \to \{0,1\}$ represents the binarization of I, where one is considered as foreground. Local binarization algorithms compute a threshold surface $T : \mathcal{P} \to \mathcal{G}$ over the whole image; $B(\boldsymbol{p}) = 1$ if $I(\boldsymbol{p})$ is lower than the threshold $T(\boldsymbol{p})$. The information to compute $T(\boldsymbol{p})$ is gathered from the pixels within a square $\mathcal{N}_r(\boldsymbol{p})$ centered at the pixel $\boldsymbol{p}$ of sides with length $2r+1$.

$P(\boldsymbol{p} \in \mathcal{A})$ denotes the probability that a pixel $\boldsymbol{p}$ belongs to $\mathcal{A} \subset \mathcal{P}$. The mean and variance of any function F in a set $\mathcal{A}$ are denoted as $\mu_{F,\mathcal{A}}$ and $\sigma^2_{F,\mathcal{A}}$.

The maximum $F(x)$ value in a set $\mathcal{A}$ is denoted as $\max_{x \in \mathcal{A}} F(x)$ while $\min_{x \in \mathcal{A}} F(x)$ denotes the minimum. Similarly, $\arg\max_{x \in \mathcal{A}} F(x)$ denotes the value of x that leads to the maximum value of $F(x)$ in $\mathcal{A}$ while $\arg\min_{x \in \mathcal{A}} F(x)$ the value of x that leads to the minimum.

We denote $H_{I,\mathcal{A}}$ the histogram of those pixels within $\mathcal{A}$. For instance, $H_{I,\mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p})}$ denotes the histogram of gray intensities foreground pixels within $\mathcal{N}_r(\boldsymbol{p})$.

2.2. Transition set

A pixel $\boldsymbol{p}$ is a *t-transition pixel* if the neighborhood $\mathcal{N}_t(\boldsymbol{p})$ contains foreground and background pixels. The set of these pixels is named $\mathcal{P}_t$. If $t=1$, then the t-transition pixel is an edge pixel.

A neighborhood that contains a large subset of $\mathcal{P}_t$ also contains a significant subset of the *foreground contour*. Furthermore, the statistical distribution of the *positive transition set* $\mathcal{F}_t = \mathcal{P}_t \cap \mathcal{F}$ approximates the distribution of $\mathcal{F}$ since it is a large foreground sample.[1] Analogously, the distribution of the *negative transition set* $\mathcal{B}_t = \mathcal{P}_t \cap \mathcal{B}$ approximates the distribution of $\mathcal{B}$, see Fig. 1.

A *transition function* F is a *discriminant function* that fulfills the following conditions:

1. Extreme positive values for pixels in $\mathcal{F}_t$.
2. Extreme negative values for pixels in $\mathcal{B}_t$.
3. F takes values close to zero in $\mathcal{P}_t^c$.

Formally, these conditions can be expressed in terms of conditional probabilities:

$$P(\boldsymbol{p} \in \mathcal{F}_t | F(\boldsymbol{p}) \geq t_+) > 1-\varepsilon_+, \tag{1}$$

$$P(\boldsymbol{p} \in \mathcal{B}_t | F(\boldsymbol{p}) \leq -t_-) > 1-\varepsilon_-, \tag{2}$$

$$P(\boldsymbol{p} \in (\mathcal{P}_t)^c | -t_- < F(\boldsymbol{p}) < t_+) \approx 1-\varepsilon, \tag{3}$$

where $\varepsilon_+ < 0.5$, $\varepsilon_- < 0.5$, and $\varepsilon < 0.5$, but the closer they are to zero, the better. Eqs. (1) and (2) mean $\boldsymbol{p}$ is pre-classified as foreground when $F(\boldsymbol{p})$ is greater than t_+. In contrast, $\boldsymbol{p}$ is pre-classified as background when $F(\boldsymbol{p})$ is lower than $-t_-$. Note that there is no information to pre-classify $\boldsymbol{p}$ if $-t_- < F(\boldsymbol{p}) < t_+$.

The previous equations suggest the *max–min* function V to measure a transition value:

$$\begin{aligned} V(\boldsymbol{p}) &= \left(\max_{\boldsymbol{q} \in \mathcal{N}_t(\boldsymbol{p})} I(\boldsymbol{q}) - I(\boldsymbol{p})\right) - \left(I(\boldsymbol{p}) - \min_{\boldsymbol{q} \in \mathcal{N}_t(\boldsymbol{p})} I(\boldsymbol{q})\right) \\ &= \max_{\boldsymbol{q} \in \mathcal{N}_t(\boldsymbol{p})} I(\boldsymbol{q}) + \min_{\boldsymbol{q} \in \mathcal{N}_t(\boldsymbol{p})} I(\boldsymbol{q}) - 2I(\boldsymbol{p}). \end{aligned} \tag{4}$$

[1] The analysis of the sampling bias is beyond the scope of this paper. Readers interested in pursuing the topic further are encouraged to consult the books by [17,18] for a more thorough explanation. Useful discussion is also available in [19,20].

Fig. 1 exemplifies two transition pixels $\boldsymbol{b}$ and $\boldsymbol{c}$. Without considering outliers, we expect that $V(\boldsymbol{c}) \approx \max_{\boldsymbol{p} \in \mathcal{N}_t(\boldsymbol{c})} I(\boldsymbol{p}) - I(\boldsymbol{c})$ and $\min_{\boldsymbol{p} \in \mathcal{N}_t(\boldsymbol{c})} I(\boldsymbol{p}) - I(\boldsymbol{c}) \approx 0$ because the pixel with minimum gray intensity in $\mathcal{N}_t(\boldsymbol{c})$ is foreground and has to be similar to the gray intensity of $\boldsymbol{c}$ (foreground smoothness). Moreover, $\max_{\boldsymbol{p} \in \mathcal{N}_t(\boldsymbol{c})} I(\boldsymbol{p}) - I(\boldsymbol{c})$ has to be higher than the minimum contrast expected in the image because it is the gray-intensity difference between foreground and background pixels. On the contrary, $V(\boldsymbol{d}) \approx 0$ because both maximum and minimum gray intensity within $\mathcal{N}_t(\boldsymbol{d})$ have to be similar to $\boldsymbol{d}$.

The concept of transition pixel, transition set and transition function were proposed and widely explained by Ramírez-Ortegón et al. [16].

We denote $H_{V,\mathcal{A}}$ the transition-value histogram of the pixels within $\mathcal{A}$. For instance, $H_{V,\mathcal{F}}$ denotes the histogram of transition values of foreground pixels.

3. Description of the transition method

Since $\mathcal{F}_t$ and $\mathcal{B}_t$ are dual sets, we will explain only the method for $\mathcal{F}_t$ leaving out details for $\mathcal{B}_t$.

3.1. Overview of the transition method

Consider that Fig. 2(a) is a neighborhood $\mathcal{N}_r(\boldsymbol{p})$; the left peak of the gray-intensity histogram $H_{I,\mathcal{N}_r(\boldsymbol{p})}$ is mainly formed by foreground pixels, while the right peak is formed by background pixels.

If we knew the *class-conditional density* $P(I(\boldsymbol{q})|\boldsymbol{q} \in \mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p}))$, we could consider the *maximum likelihood estimation* or *Bayesian estimation* approach to solve the binarization problem. Unfortunately, we rarely know the class-conditional densities. However, we can reasonably assume that the gray intensities of the foreground are approximately normally distributed with mean μ_+ and variance σ_+^2 in small neighborhood $\mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p})$. That is

$$H_{I,\mathcal{F}\cap\mathcal{N}_r(\boldsymbol{p})}(i) \approx c_+ \phi(i; \mu_+, \sigma_+^2), \tag{5}$$

where $c_+ = |\mathcal{F}_t \cap \mathcal{N}_r(\boldsymbol{p})|$ and $\phi(x; \mu, \sigma)$ denotes the normal probability density function with mean μ and variance σ^2. $T(\boldsymbol{p})$ is quickly computed when there is an analytic intersection between $c_+ \phi(i; \mu_+, \sigma_+^2)$ and the correspondent background function $c_- \phi(i; \mu_-, \sigma_-^2)$, see Fig. 2(b).

We can approximate $P(I(\boldsymbol{q})|\boldsymbol{q} \in \mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p}))$ by drawing a representative sample of $\mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p})$. The positive transition set $\mathcal{F}_t(\boldsymbol{p})$ satisfies

$$P(I(\boldsymbol{q})|\boldsymbol{q} \in \mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p})) \approx P(I(\boldsymbol{q})|\boldsymbol{q} \in \hat{\mathcal{F}}_t \cap \mathcal{N}_r(\boldsymbol{p})). \tag{6}$$

Hence, $\mathcal{F}_t \cap \mathcal{N}_r(\boldsymbol{p})$ is a representative sample of $\mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p})$, see Fig. 2(c).

Although the transition sets are also unknown, our method provides $\hat{\mathcal{F}}_t$ which is an accurate estimate of $\mathcal{F}_t$, see Fig. 2(d). Thus, (6) changes to

$$P(I(\boldsymbol{q}) \mid \boldsymbol{q} \in \mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p})) \approx P(I(\boldsymbol{q}) \mid \boldsymbol{q} \in \hat{\mathcal{F}}_t \cap \mathcal{N}_r(\boldsymbol{p})). \tag{7}$$

We are now able to compute the gray threshold with usual classification procedures. In Table 1, for instance, we computed the *minimum symmetric value* as

$$s_{\mathcal{A}} = \underset{i \in [k, l-k], H_{I,\mathcal{A}}(i) > 0}{\arg\min} \frac{1}{H_{I,\mathcal{A}}(i)} \sum_{j=1}^{k} |H_{I,\mathcal{A}}(i+j) - H_{I,\mathcal{A}}(i-j)|,$$

where $\mathcal{A}$ is a set of pixels and $k \le l/2$ is an integer.

Table 1
Estimated threshold by minimum symmetric value with $k = 10$.

	a	b	$T(\boldsymbol{p}) = \frac{a+b}{2}$
Ground truth	$s_{\mathcal{F}\cap\mathcal{N}_r(\boldsymbol{p})} = 98$	$s_{\mathcal{B}\cap\mathcal{N}_r(\boldsymbol{p})} = 207$	152.5
Transition set	$s_{\mathcal{F}_t\cap\mathcal{N}_r(\boldsymbol{p})} = 99$	$s_{\mathcal{B}_t\cap\mathcal{N}_r(\boldsymbol{p})} = 205$	152
Transition set approximation	$s_{\hat{\mathcal{F}}_t\cap\mathcal{N}_r(\boldsymbol{p})} = 98$	$s_{\hat{\mathcal{B}}_t\cap\mathcal{N}_r(\boldsymbol{p})} = 207$	152.5

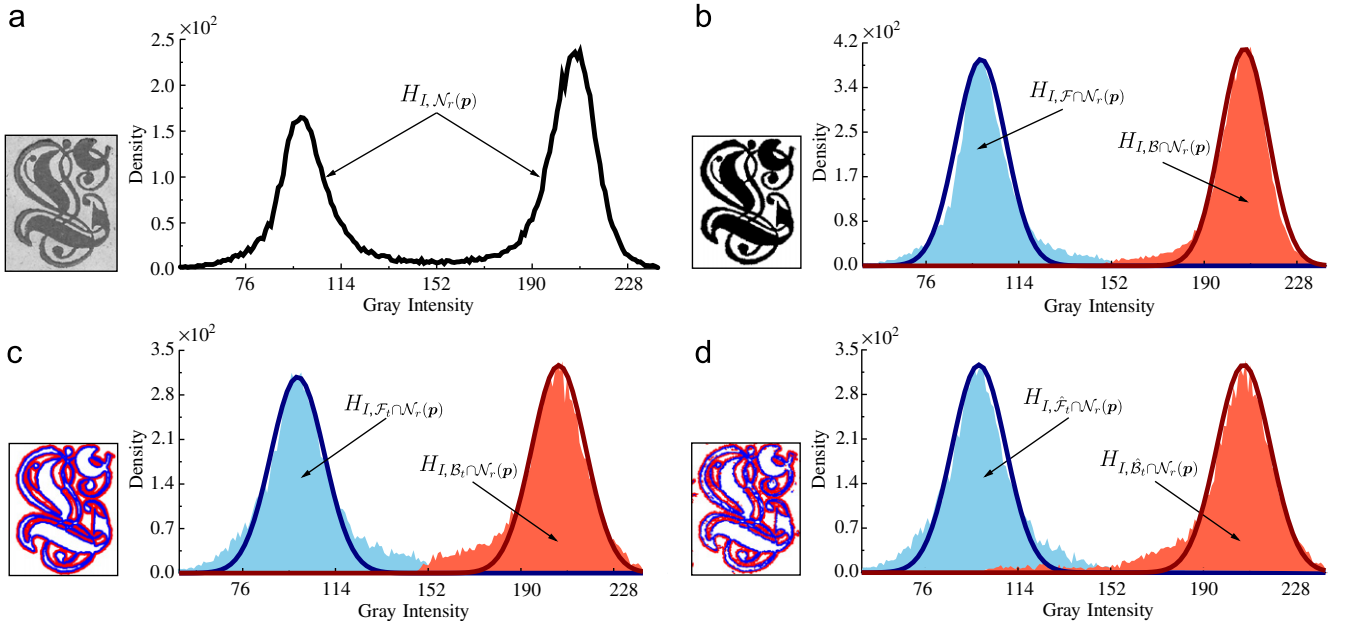

Fig. 2. (a) Gray-intensity image of the neighborhood $\mathcal{N}_r(\boldsymbol{p})$ and gray-intensity histogram $H_{I,\mathcal{N}_r(\boldsymbol{p})}$ of pixels within $\mathcal{N}_r(\boldsymbol{p})$. (b) Binary ground truth and histogram of gray intensities of both foreground and background. We have manually fitted a normal probability density distribution function to each histogram. (c) Transition sets. In blue, pixels within the positive transition set $\mathcal{F}_t$, while pixels within the negative transition set $\mathcal{B}_t$ are shown in red. In $\mathcal{N}_r(\boldsymbol{p})$, the density of gray intensities of $\mathcal{B}_t$ and $\mathcal{F}_t$ approximate the gray-intensity densities of $\mathcal{B}$ and $\mathcal{F}$, respectively. (d) Approximation of the transition sets. We use the approximation of positive and negative transition sets as foreground and background samples, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

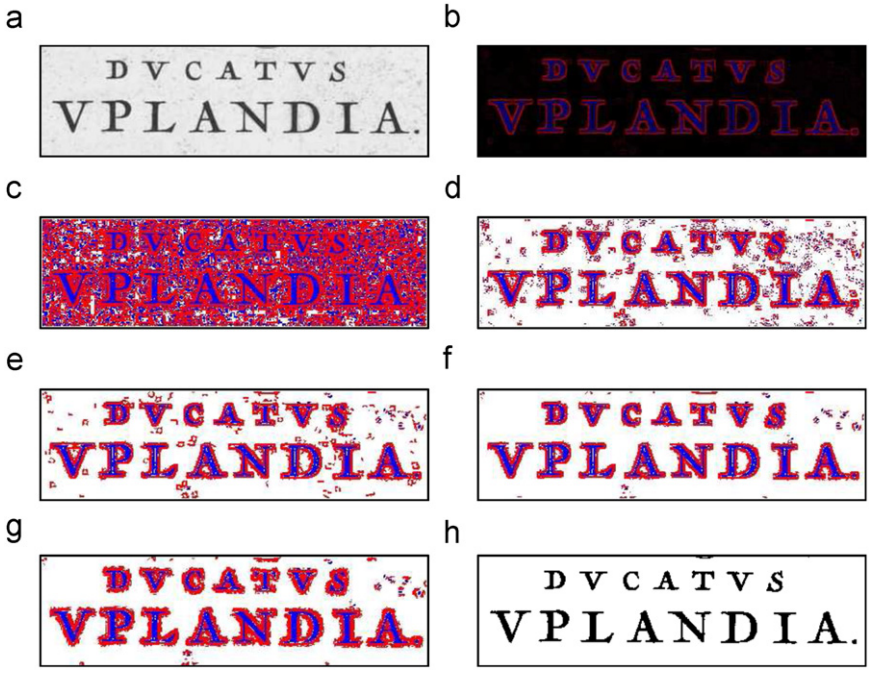

Fig. 3. (a) Original image. (b) Transition image by function max–min using $\mathcal{N}_2$. (c) Transition image. In blue, pixels with transition value higher than zero; in red, pixels with transition value lower than zero. (d) The transition image after filtering by $t_+=14$ and $t_-=15$. (e) Transition image after removing isolated pixels. (f) Transition image after incidence transition operators. (g) Transition image after dilation transition operators. (h) Binary image. Modeling the gray intensities as lognormally distributed. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

We abbreviate $\mathcal{P}_{t,r}(\boldsymbol{p})=\mathcal{P}_t\cap\mathcal{N}_r(\boldsymbol{p})$ and $\hat{\mathcal{P}}_{t,r}(\boldsymbol{p})=\hat{\mathcal{P}}_t\cap\mathcal{N}_r(\boldsymbol{p})$. We define $\mathcal{F}_{t,r}(\boldsymbol{p})$, $\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})$, $\mathcal{B}_{t,r}(\boldsymbol{p})$, and $\hat{\mathcal{B}}_{t,r}(\boldsymbol{p})$ in a similar manner.

The complete method consists of the following steps:

1. Compute the transition values for each pixel with a transition function. We suggest the max–min function using neighborhoods of radius 2, see Fig. 3(b).
2. Calculate the thresholds t_+ and t_-. Take $\hat{\mathcal{F}}_t=\{\boldsymbol{p}\mid V(\boldsymbol{p})\geq t_+\}$ and $\hat{\mathcal{B}}_t=\{\boldsymbol{p}\mid V(\boldsymbol{p})\leq -t_-\}$ (Section 3.2), see Fig. 3(b).
3. Restore $\hat{\mathcal{F}}_t$ and $\hat{\mathcal{B}}_t$ (Section 3.3), see Fig. 3(f, g).
4. Label $\boldsymbol{p}$ using information from $\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})$ and $\hat{\mathcal{B}}_{t,r}(\boldsymbol{p})$ (Section 3.4).
5. - If binarization: Compute $T(\boldsymbol{p})$ (Section 3.4.1), Fig. 3(h).
   - If edge detection: Compute simple edge transition operator (Section 3.5).
6. Remove noise from B with standard algorithms.

3.2. Transition threshold

We find a suitable transition threshold by analyzing functions of the transition values:

Empirical scaled density function:

$$u_i=\frac{1}{k}H_{V,\mathcal{P}}(i)\quad\text{where } k=\max_{i\in[1,l]}H_{V,\mathcal{P}}(i). \tag{8}$$

Empirical complementary cumulative distribution function:

$$v_i=\frac{1}{k}\sum_{j=i}^{l}H_{V,\mathcal{P}}(j)\quad\text{where } k=\sum_{j=1}^{l}H_{V,\mathcal{P}}(j). \tag{9}$$

The behavior of the positive transition values (Fig. 4(a)) will appear to have a heavy right tail. The *power-law distribution* has been discarded because the *log–log plot* (Fig. 4(b)) of the empirical complementary cumulative distribution (CCD) function does not follow the characteristic straight-line form of the power-law distribution [21].

Fig. 4(a) shows two linear zones. The first linear relation mostly corresponds to non-transition set $\mathcal{P}_t^c$ having positive transition value. The second linear part is mainly formed by transition pixels. Indeed, the histogram of positive transition values is a combination of three histogram as is shown in Fig. 4(b).[2] Thus, a criterion to select the transition threshold t_+ is to take the value t that divide the graph, using linear–linear or linear–log scales, into approximately two lines.

3.2.1. Double-linear threshold for transition values

The double-linear method approximates the positive side of the transition graphs (i,w_i) by joining two linear functions, see Fig. 5(a), where w_i is computed as (8) or (9). However, the transition graph is truncated between the bounds x_{min} and x_{max} in order to reduce the noise in the first and last values of the graph, where x_{min} is the minimum index i that satisfies

$$w_i>w_{i+1}\quad\text{and}\quad w_i>w_{i+2}\quad\text{and}\quad w_{i+1}>w_{i+2} \tag{10}$$

and x_{max} is the maximum index i that satisfies

$$\frac{w_i}{w_{x_{min}}}>\delta, \tag{11}$$

such that $\delta>0$ is small (we suggest $\delta=0.01$).

For mathematical convenience, we re-label our values w_i as

$$y_i=w_{i+x_{min}}\quad\text{for } i=0,1,\ldots,x_{max}-x_{min}=n \tag{12}$$

and postulate that y_i satisfies (13) and (14):

$$y_i\approx m_1\cdot i+b_1\quad\text{if } i=0,1,2,\ldots,t, \tag{13}$$

$$y_i\approx m_2\cdot i+b_2\quad\text{if } i=t,t+1,\ldots,n. \tag{14}$$

[2] We have constructed a formal treatment of this argument using some probability assumptions over the gray-intensity differences. This work has been submitted for publication.

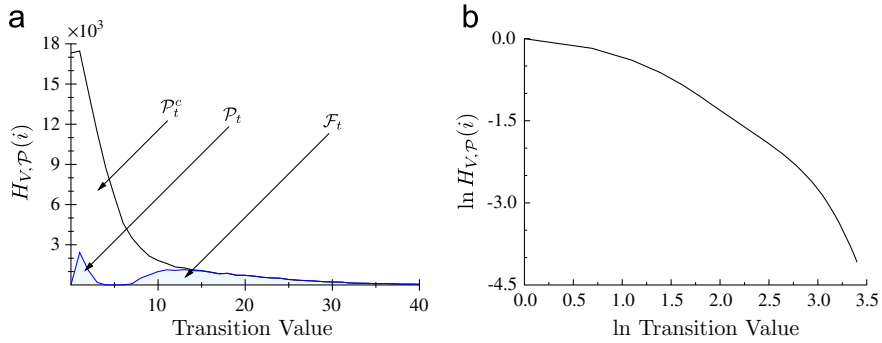

Fig. 4. (a) Estimated histogram of the positive transition set $\mathcal{F}_t$ given the histogram of positive transition values. (b) The log–log plot of the empirical complementary cumulative distribution functions of the positive transition pixels does not follow the characteristic straight-line form of the power-law distribution.

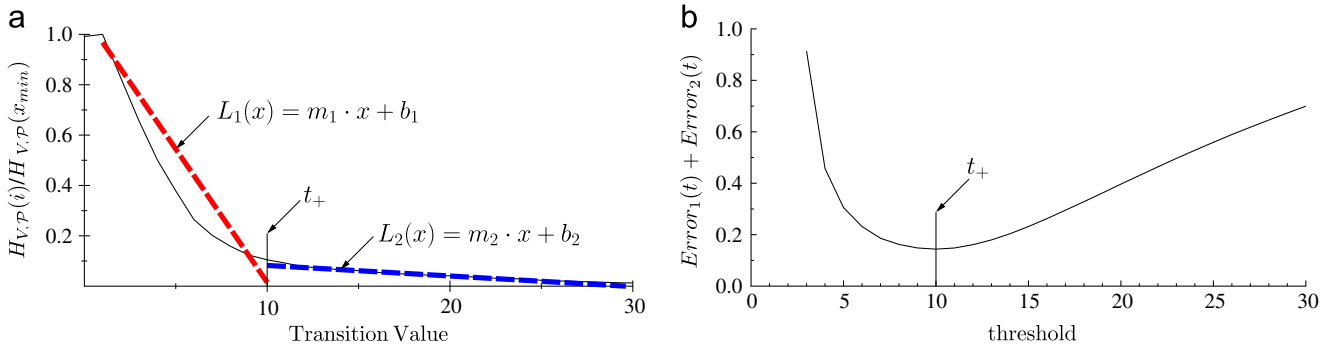

Fig. 5. (a) The scaled density function is approximated by the joining of two lines L_1 and L_2. (b) Plot of $Error_1(t)$ + $Error_3(t)$ between x_{min}=3 and x_{max}=30.

We use the *differences rate estimator* [22] to compute $\hat{m}_1$. This estimator is defined as

$$\hat{m}_1 = \frac{6}{t(t+1)(t+2)} \sum_{i=0}^{t} (2i-t) y_i. \tag{15}$$

However, the slope can be computed by *regression methods* as in [23,24].

Unfortunately, there is no differences-rate estimator for the intercept term b, therefore we use the *least-square estimator*

$$\hat{b}_1 = \frac{1}{t+1} \sum_{i=0}^{t} (y_i - \hat{m}_1 \cdot i). \tag{16}$$

A natural error function for (13) can be defined as

$$Error_1(t) = \sum_{i=0}^{t} (y_i - \hat{m}_1 \cdot i - \hat{b}_1)^2. \tag{17}$$

In the same way, an error function for (14) is defined as

$$Error_2(t) = \sum_{i=t}^{n} (y_i - \hat{m}_2 - x_i \cdot i - \hat{b}_2)^2, \tag{18}$$

where

$$\hat{m}_2 = \frac{6}{(n-t)(n-t+1)(n-t+2)} \sum_{i=0}^{n-t} (2i-n+t) y_i \tag{19}$$

and

$$\hat{b}_2 = \frac{1}{n-t+1} \sum_{i=t}^{n} (y_i - m_2 \cdot i). \tag{20}$$

Finally, t_+ is computed as

$$t_+ = \arg\min_{t \in [1,n]} Error_1(t) + Error_2(t) + x_{min} + 2. \tag{21}$$

Fig. 5(b) plot of (21).

3.3. Restoration of transition set

The restoration of the transition set $\hat{\mathcal{P}}_t$ is the process of adding and removing pixels from $\hat{\mathcal{P}}_t$ with the aim of increasing the cardinality while reducing the noise.

Morphological operators, like *erosion* and *dilation* [25], could be adapted to enhance $\hat{\mathcal{P}}_t$. However, those operators will add or remove pixels without considering neither gray intensities nor transition values. Those operators in their original form will alter the trusty foreground sample $\hat{\mathcal{F}}_t$, loosing the confidence in the transition set approximation. We propose morphological operators that preserve the confidence of the transition set approximation.

3.3.1. Isolation transition operator

The *cross* and *diagonal isolate operators* were successfully used in [16] in order to remove false positives of the approximated transition sets. The *cross isolate operator* is defined as

$$\text{Given } \boldsymbol{p} \in \hat{\mathcal{F}}_t, \text{ set } \hat{\mathcal{F}}_t \leftarrow \hat{\mathcal{F}}_t \setminus \{\boldsymbol{p}\} \text{ if } |\hat{\mathcal{F}}_t \cap \mathcal{N}_{\oplus}(\boldsymbol{p})| = 0. \tag{22}$$

where $\mathcal{N}_{\oplus}(\boldsymbol{p}_{i,j}) = \{\boldsymbol{p}_{i-1,j}, \boldsymbol{p}_{i+1,j}, \boldsymbol{p}_{i,j-1}, \boldsymbol{p}_{i,j+1}\}$. Instead the cross neighborhood, the *diagonal transition operator* uses the diagonal neighborhood $\mathcal{N}_{\otimes}(\boldsymbol{p}_{i,j}) = \{\boldsymbol{p}_{i-1,j-1}, \boldsymbol{p}_{i-1,j+1}, \boldsymbol{p}_{i+1,j-1}, \boldsymbol{p}_{i+1,j+1}\}$.

Some applications, like text recognition or fingerprint classifier, assume that the foreground consists of large connected components. These components are commonly larger than a particular rectangular area. Then, the connected components that are completely contained in small rectangular neighborhoods may be removed. We define the *rectangular transition operator* as

$$\hat{\mathcal{F}}_t \leftarrow \hat{\mathcal{F}}_t \setminus \{\boldsymbol{p}\} \text{ if } \boldsymbol{p} \in \hat{\mathcal{F}}_t \text{ and } |\hat{\mathcal{F}}_t \cap \mathcal{N}_{x,y}(\boldsymbol{p})| = |\hat{\mathcal{F}}_t \cap \mathcal{N}_{x+1,y+1}(\boldsymbol{p})|, \tag{23}$$

where $\mathcal{N}_{y,x}(\boldsymbol{p})$ is defined as the rectangular neighborhood centered at the pixel $\boldsymbol{p}$ of sides with length $2y+1$, $2x+1$.

3.3.2. Incidence transition operator

The blue pixels in Fig. 6(a) depict pixels with high positive transition value. In red, those pixels with high negative transition value. In the same figure, whereas the isolated blue pixel $\boldsymbol{q}$ (right-bottom corner) is an outlier and easily removed by cross, diagonal or rectangular transition operators, the red pixels around $\boldsymbol{q}$ form a large "isolated" connected component (24 pixels) that cannot be removed by those operators.

By definition, a background t-transition pixel $\boldsymbol{p}$ contains at least one foreground t-transition pixels in $\mathcal{N}_t(\boldsymbol{p})$. That is $|\mathcal{F}_{t,t}(\boldsymbol{p})| \geq 1$, if $\boldsymbol{p} \in \mathcal{P}_t$. Moreover, $|\mathcal{F}_{t,2t}(\boldsymbol{p})| > |\mathcal{F}_{t,t}(\boldsymbol{p})|$ in most of the transition pixels. Thus, the neighborhood $\mathcal{N}_{2t}(\boldsymbol{p})$ of a pixel with high positive transition value may contain several pixels with high positive transition value. An example is Fig. 6(a), where $|\hat{\mathcal{F}}_{2,2}(\boldsymbol{p})| = 1$ and $|\hat{\mathcal{F}}_{2,4}(\boldsymbol{p})| = 8$. In opposite to $\boldsymbol{p}$, the pixel $\boldsymbol{q}$ and all the red pixels around it only contains one blue pixel in $\mathcal{N}_4(\boldsymbol{q})$.

To deal with pixels like $\boldsymbol{q}$, we define a generalization of isolated transition pixel as: Given two positive integers a and b, a pixel $\boldsymbol{p} \in \mathcal{P}_t$ is a *k-isolated transition pixel* if

$$|\hat{\mathcal{F}}_{t,k}(\boldsymbol{p})| < a \quad \text{or} \quad |\hat{\mathcal{B}}_{t,k}(\boldsymbol{p})| < b. \tag{24}$$

The *incidence transition operator* removes all the k-isolated transition pixels. We recommend setting $k = 2t$, $a = b = 1 + t$.

3.3.3. Dilation transition operator

Suppose that $\boldsymbol{p}$ and $\boldsymbol{q} \in \mathcal{N}_t(\boldsymbol{p})$ are two foreground pixels such that $\boldsymbol{q} \in \mathcal{F}_t$ and $\boldsymbol{p} \notin \mathcal{F}_t$, thus $V(\boldsymbol{q}) \geq t_+$ and $V(\boldsymbol{p}) < t_+$. This implies that $\boldsymbol{p}$ is excluded from $\hat{\mathcal{F}}_t$. However, we can assume

$$P(I(\boldsymbol{q}) \geq I(\boldsymbol{p})) \approx P(I(\boldsymbol{q}) \leq I(\boldsymbol{p})) \quad \text{if } \boldsymbol{p},\boldsymbol{q} \in \mathcal{F} \cap \mathcal{N}_t(\boldsymbol{p}) \tag{25}$$

because $I(\boldsymbol{q}) \approx I(\boldsymbol{p})$ for all $\boldsymbol{q} \in \mathcal{F} \cap \mathcal{N}_t(\boldsymbol{p})$. So,

$$P(I(\boldsymbol{q}) \geq I(\boldsymbol{p})) \approx P(I(\boldsymbol{q}) \leq I(\boldsymbol{p})) \quad \text{if } \boldsymbol{p},\boldsymbol{q} \in \mathcal{F}_{t,t}(\boldsymbol{p}). \tag{26}$$

In other words, about half of the pixels in $\hat{\mathcal{F}}_{t,t}(\boldsymbol{p})$ have a gray intensity equal or lower than $I(\boldsymbol{p})$ (Fig. 6(b), top). In addition, the gray intensities of the background are strictly higher than $I(\boldsymbol{p})$ in the ideal case. Therefore, the number of pixels that are equal or lower in gray intensity than $I(\boldsymbol{p})$ may be zero or close to zero. Table 2 is constructed following the same reasoning, although a formal proof of the probabilities is beyond the scope of this paper.

Using the conditional probabilities of Table 2, a high number of pixels in $\mathcal{F}_{t,t}(\boldsymbol{p})$ that are equal or lower in gray intensity than $I(\boldsymbol{p})$ is a strong evidence that $\boldsymbol{p}$ belongs to the foreground. We derived a similar argument for background pixels.

To measure this conditional probabilities, we define the *t-transition balance*:

$$TB_t(\boldsymbol{p}) = |\{\boldsymbol{q} \in \mathcal{F}_{t,t}(\boldsymbol{p}) | I(\boldsymbol{q}) \geq I(\boldsymbol{p})\}| - |\{\boldsymbol{q} \in \mathcal{B}_{t,t}(\boldsymbol{p}) | I(\boldsymbol{q}) \geq I(\boldsymbol{p})\}|. \tag{27}$$

So, $TB_t(\boldsymbol{p}) \approx \frac{1}{2}|\mathcal{F}_{t,t}(\boldsymbol{p})|$ if $\boldsymbol{p}$ is foreground, and $TB_t(\boldsymbol{p}) \approx -\frac{1}{2}|\mathcal{B}_{t,t}(\boldsymbol{p})|$ if $\boldsymbol{p}$ is background. Hence, $TB_t(\boldsymbol{p})$ is approximated with $\widehat{TB}_t(\boldsymbol{p})$ which uses $\hat{\mathcal{P}}_t(\boldsymbol{p})$ instead of $\mathcal{P}_t(\boldsymbol{p})$.

Given $\boldsymbol{p} \notin \mathcal{P}_t$, the *dilation transition operator* set

$$\hat{\mathcal{F}}_t \leftarrow \hat{\mathcal{F}}_t \cup \{\boldsymbol{p}\} \quad \text{if } \widehat{TB}(\boldsymbol{p}) \geq a \tag{28}$$

and

$$\hat{\mathcal{B}}_t \leftarrow \hat{\mathcal{B}}_t \cup \{\boldsymbol{p}\} \quad \text{if } \widehat{TB}(\boldsymbol{p}) \leq -b, \tag{29}$$

where a and b are two positive integers. We recommend setting $a = b = 1 + t$.

3.4. Positive and negative sets

Only pixels in $\hat{\mathcal{P}}_{t,r}(\boldsymbol{p}) = \hat{\mathcal{F}}_{t,r}(\boldsymbol{p}) \cup \hat{\mathcal{B}}_{t,r}(\boldsymbol{p})$ are considered to compute $T(\boldsymbol{p})$. At the same time, outliers are discarded by labeling as background those pixels $\boldsymbol{p}$ that satisfy either $|\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})| < n_+$ or $|\hat{\mathcal{B}}_{t,r}(\boldsymbol{p})| < n_-$, where n_+ and n_- depend on r and objects of interest; the higher the n_+, the larger the objects that can be removed from the foreground. For instance, consider a simple horizontal line with height 1 as the foreground. The line's extremes are evaluated if $n_+ \leq r+1$. Otherwise, the line's extremes are labeled as background without even computing $T(\boldsymbol{p})$. We suggest $n_+ = n_- = 5$ for detecting small foreground objects.

A second criterion to discard outliers uses the difference between the mean of gray intensities of both transition set approximations. The pixel $\boldsymbol{p}$ is labeled as background if

$$\mu_{I,\hat{\mathcal{B}}_{t,r}(\boldsymbol{p})} - \mu_{I,\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})} < c, \tag{30}$$

where c is an integer, which depicts the minimum contrast expected between the foreground and background. We suggest $c = 15$.

Table 2
The probability of the inequality is approximated given $\boldsymbol{p}$ and $\boldsymbol{q} \in \mathcal{N}_r(\boldsymbol{p})$.

	$P(I(\boldsymbol{p}) \geq I(\boldsymbol{q}))$		$P(I(\boldsymbol{p}) \leq I(\boldsymbol{q}))$	
	$\boldsymbol{q} \in \mathcal{B}$	$\boldsymbol{q} \in \mathcal{F}$	$\boldsymbol{q} \in \mathcal{B}$	$\boldsymbol{q} \in \mathcal{F}$
$\boldsymbol{p} \in \mathcal{B}$	≈ 0.5	≈ 1	≈ 0.5	≈ 0
$\boldsymbol{p} \in \mathcal{F}$	≈ 0	≈ 0.5	≈ 1	≈ 0.5

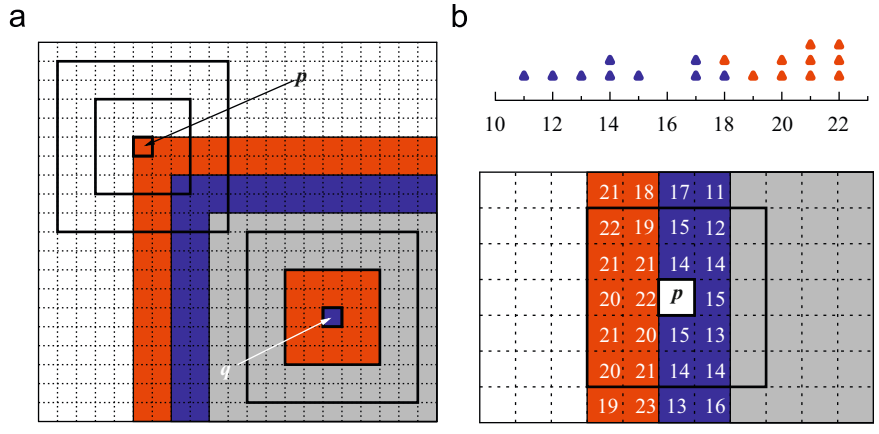

Fig. 6. (a) The transition values were computed using $\mathcal{N}_2$. The pixels with high positive value are shown in blue, in red the pixels with high negative value. (b) Bottom: the gray-intensity of the blue pixel in $\mathcal{N}_2(\boldsymbol{p})$ is lower or equal to $I(\boldsymbol{p})$. (b) Top: the approximation of the transition balance of $\boldsymbol{p}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

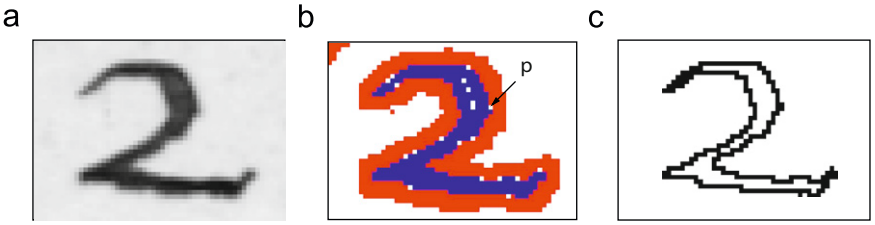

Fig. 7. (a) Original image. (b) Transition set approximation. (c) Edge image by transition operator.

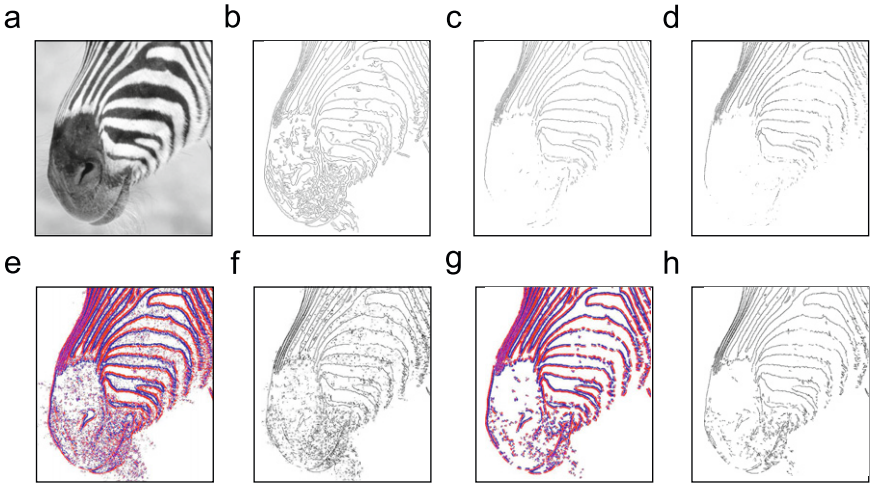

Fig. 8. (a) Original image. (b) Edge image by Canny method. (c) Edge image by Prewitt method. (d) Edge image by Roberts method. (e) Raw transition image. (f) Edge image of (e) computed by the simple edge transition operator. (g) Restored transition image of (a). (h) Edge image of (g) computed by the simple edge transition operator.

3.4.1. Statistical thresholds

Normal threshold: Assume that $H_{I,\mathcal{F}_{t,r}(\boldsymbol{p})}(i) \approx c_+ \phi(i;\mu_+,\sigma_+^2)$, where $\phi(x;\mu,\sigma)$ is the normal probability density function with mean μ and variance σ^2. Then, $H_{I,\mathcal{F}_{t,r}(\boldsymbol{p})}$ is approximated when

$$c_+ = |\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})| \text{ (complete form)} \quad \text{or} \quad c_+ = 1 \text{ (simple form)},$$
$$\mu_+ = \mu_{I,\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})},$$
$$\sigma_+^2 = \max(\sigma^2_{I,\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})}, 1). \tag{31}$$

The intersection of those curves is the root $\mu_+ < x_0 < \mu_-$ of the quadratic equation with coefficients a, b and c given by

$$a = \frac{1}{\sigma_+^2} - \frac{1}{\sigma_-^2},$$

$$b = \frac{2\mu_-}{\sigma_-^2} - \frac{2\mu_+}{\sigma_+^2},$$

$$c = \frac{\mu_+^2}{\sigma_+^2} - \frac{\mu_-^2}{\sigma_-^2} - 2\ln\left(\frac{\sigma_- \cdot c_-}{\sigma_+ \cdot c_+}\right). \tag{32}$$

Lognormal threshold: Assume that $H_{I,\mathcal{F}_{t,r}(\boldsymbol{p})}(i) \approx c_+ \lambda(i;\mu_+,\sigma_+^2)$, where $\lambda(\mu,\sigma^2)$ denotes the lognormal probability density function with parameters μ and σ^2.

The intersection is computed using the root x_0 of the quadratic equation with coefficients given by (32). This corresponds to $\exp(x_0)$.

In practice, the parameters μ_+ and σ_+^2 are estimated based on the estimated mean and variance of the lognormal distribution using the relations

$$\mu_+ = \ln(\mu_{I,\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})}) - \frac{1}{2}\sigma_+^2 \quad \text{and} \quad \sigma_+^2 = \ln\left(1 + \frac{\sigma^2_{I,\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})}}{[\mu_{I,\hat{\mathcal{F}}_{t,r}(\boldsymbol{p})}]^2}\right). \tag{33}$$

3.5. Edge detection

In binarization context, an *edge pixel* $\boldsymbol{p}$ can be defined as a foreground pixel that contains background pixels within $\mathcal{N}_1(\boldsymbol{p})$. Therefore, $\mathcal{F}_1$ is the set of edge pixels. We could define an edge pixel $\boldsymbol{p}$ as the pixel that contains both foreground and background pixels within $\mathcal{N}_1(\boldsymbol{p})$, or as those background pixels that contain foreground pixels in the neighborhood of radius 1. Nevertheless, we prefer the first edge pixel definition.

We can approximate $\mathcal{F}_1$ by

$$\hat{\mathcal{F}}_1 = \{\boldsymbol{p} | \boldsymbol{p} \in \hat{\mathcal{F}}_t \text{ and } |\hat{\mathcal{B}}_{t,1}(\boldsymbol{p})| > 0\}. \tag{34}$$

The pixel $\boldsymbol{p}$ in Fig. 7(b), which belongs to $\hat{\mathcal{P}}_t^c$, can be considered as edge pixel since it is exactly between pixels in $\hat{\mathcal{F}}_t$ and $\hat{\mathcal{B}}_t$. Hence, we define the *simple edge transition operator* as

$$\hat{\mathcal{F}}_1 = \{\boldsymbol{p} | 0 < |\hat{\mathcal{F}}_{t,1}(\boldsymbol{p})| \text{ and } |\hat{\mathcal{B}}_{t,1}(\boldsymbol{p})| > 0\}. \tag{35}$$

Fig. 8(b)–(d) were computed on MatLab [26] using Canny [27], Prewitt [28] and Roberts Cross methods,[3] respectively. Fig. 8(f) was computed following steps 1 and 2 of the transition method (Fig. 8(e)) and applying the simple edge transition operator. The raw transition set approximation (without restoration process) generates many false positives. In contrast, Fig. 8(h), which follows the transition method with a restored transition set, reports a lower number of false negatives than Fig. 8(f). Unfortunately, the combination of transition operators used in Fig. 8(e) includes more than one cross, diagonal and incidence transition operators in a non-trivial order.

[3] The default parameters of MatLab are chosen heuristically in a way that depend on the input data.

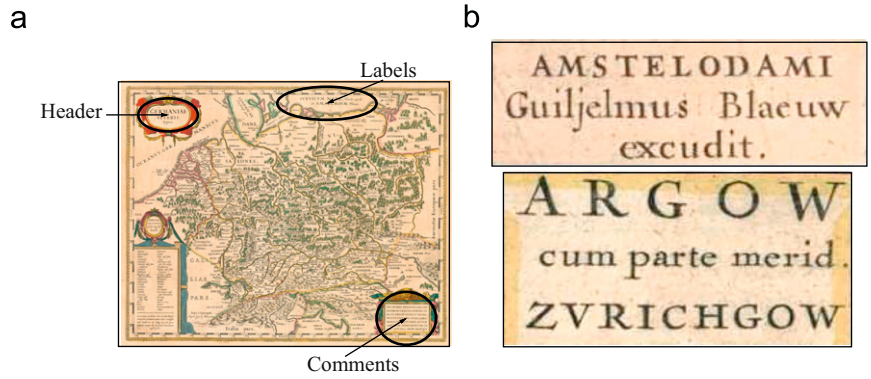

Fig. 9. (a) Example of map that contains header, labels a comment. (b) Example of label and header.

4. Design of experiments

To perform an objective evaluation of the binarization results, we have used an unsupervised measure, based on gray-intensity variances, and an optical character recognition (OCR) software.

4.1. Test images

Historical documents usually present several kinds of degradations, such as ink stains, burned areas, weak ink strokes and wide variations in the background. Thus, the binarization algorithms were tested with digitalized images of the historical atlas "*Theatrum orbis terrarum*, *sive*, *Atlas novus*" (Blaeu Atlas) [29] at 150 dpi resolution.

We reported the results of $n=86$ images I_i, for $i=1,\ldots,n$, extracted from 61 maps. These images are mainly composed of map headers, map comments and region labels without stylized handwriting characters, see Fig. 9.

4.2. Description of binarization algorithms

We compare Kittler's [9], Otsu's [11], Portes's [12], Sauvola's [10], Wolf's [15] methods (top ranked by [7,8,30]) with four variants of the transition method. We implemented all nine algorithms in their local versions to increase their accuracy, despite local implementations raise the running-time dramatically.

Because Kitller's, Otsu's and Porte's algorithms assume that the set of pixels can be divided into two clusters, computing $T(\boldsymbol{p})$ with these algorithms will systematically produce false positives due to neighborhoods that are completely contained in the background. To avoid these false positives, $T(\boldsymbol{p})=t^*$ if the difference between the gray-intensity means of the clusters is higher than c, where t^* is the optimal local threshold according to the binarization algorithm and c depicts the minimum contrast expected between the foreground and the background. Otherwise the pixel is classified as background. The difference of gray-intensity means between the clusters is given by

$$\mu_0-\mu_1<c \quad \text{with } \mu_i=\frac{1}{|\mathcal{C}_i|}\sum_{\boldsymbol{q}\in\mathcal{C}_i} I(\boldsymbol{q}) \text{ for } i=0,1, \tag{36}$$

where $\mathcal{C}_0=\{\boldsymbol{q}\in\mathcal{N}_r(\boldsymbol{p})|I(\boldsymbol{q})\le t^*\}$ and $\mathcal{C}_1=\{\boldsymbol{q}\in\mathcal{N}_r(\boldsymbol{p})|I(\boldsymbol{q})>t^*\}$.

Real applications rarely use more than one parameter set. This is the main reason we fixed Sauvola's $\alpha=0.5$ and $\beta=128$, Portes's $\alpha=2$, and Wolf's $\alpha=0.5$, which are the recommended parameters.[4]

[4] The higher the α, the higher the threshold in both Sauvola's and Wolf's algorithms. Portes et al. did not determine the relation between α and their proposed threshold.

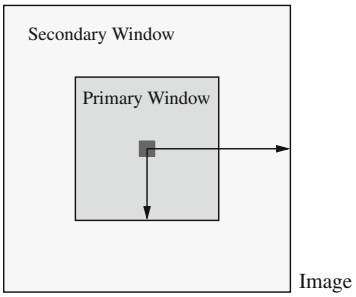

Fig. 10. All algorithms gather the threshold information from a primary neighborhood, although Wolf's algorithm uses a secondary neighborhood to compute any "global information".

We set the primary neighborhood radius to $r=50$, local windows of 101 $\times$ 101 pixels, and set the secondary neighborhood radius to 100 for Wolf's method, see Fig. 10.

The transition algorithms, denoted by the prefix T, are composite methods with the following combination of operators:

- Max–min function using $\mathcal{N}_2$.
- Double-linear threshold for transition values using either the empirical scaled density function denoted by DF or the empirical complementary cumulative distribution function denoted by CCD.
- Isolate transition operators (in order): cross transition operator, diagonal transition operator and rectangular transition operator ($x=y=2$).
- Incidence transition operator ($k=4$, $a=b=3$).
- Dilation transition operator ($a=b=3$).
- Gray-intensity threshold. Setting $n_+=n_-=25$, $c=15$, and using either the normal threshold (simple form) denoted by N or the lognormal threshold (simple form) denoted by L.

We named our four variants as T-DF-N, T-DF-L, T-CCD-N and T-CCD-L depending on how the algorithm computes the transition and gray-intensity thresholds.

We also tested three variants of T-CCD-L in order to analyze the influence of transition operators on the transition method: T-CCD-L-A does not include any transition operator, T-CCD-L-B includes only the isolate transition operators, and T-CCD-L-C includes both isolate transition operators and incidence transition operators.

4.3. Post-processing step

All binarized images were post-processed removing from the foreground small stains (connected components containing four or less pixels) before computing any evaluation measure. The following operators were applied in this order: cross operator,

diagonal operator and rectangular operator ($x=y=2$) which correspond to cross, diagonal and rectangular transition operators but filtering the binary image.

4.4. Evaluation measures

Since there is not a public standard benchmark available for binarization algorithms, we use the *uniform variance* (UV) measure [16]

$$UV = \sum_{p \in \mathcal{P}} \frac{\sigma_{\mathcal{B}}(\boldsymbol{p}) \cdot |\mathcal{B} \cap \mathcal{N}_r(\boldsymbol{p})| + \sigma_{\mathcal{F}}(\boldsymbol{p}) \cdot |\mathcal{F} \cap \mathcal{N}_r(\boldsymbol{p})|}{|\mathcal{N}_r(\boldsymbol{p})|}$$

to evaluate the image segmentation quality.

Besides the UV measure, we used *TopOCR* [31] to recognize the text from binarized images. Our evaluation measures are *accuracy* (*AC*) and *precision* (*PR*) computed as

$$AC = \frac{\#(\text{characters of } T_{\text{match}})}{\#(\text{characters of } T_{\text{in}})} \quad \text{and} \quad PR = \frac{\#(\text{characters of } T_{\text{match}})}{\#(\text{characters of } T_{\text{out}})},$$

where T_{in} is the original text in the image, T_{out} denotes the recognized text from the OCR, and T_{match} denotes the maximum matching text between T_{in} and T_{out}. T_{match} is computed using Needleman–Wuntsh algorithm [32].

TopOCR was tested with four parameter sets. The program tester reports the AC and PR measures from the parameter set that scores higher in terms of AC measure. If there is any draw, PR measure is used.

AC measure is an important measure for OCR applications, because a high AC measurement increases the possibility to extract, by further algorithms, relevant information. However, the mean and variance of *AC performance* $x_{i,j}$, for each binarization method j and image I_i, highly depend on the maximum AC performance x_i^* that the OCR can compute for I_i. Consequently, we cannot infer from $x_{i,j}$ how effective is the algorithm j to maximize the OCR performance. For instance, suppose that $x_i^* < 0.3$, for $i=1,\ldots,n$, this implies that the average of $x_{i,j}$ (fixing j) is lower than 0.3. However, we cannot affirm that the binarization algorithm j is ineffective to maximize the OCR performance. Hence, our observations are mainly based on pairwise tables and the ratio $y_{i,j}$ of $x_{i,j}$ to x_i^*. This ratio represents the *AC efficacy* of algorithm j to maximize the AC performance in I_i. We have computed x_i^* by sweeping parameters (including the neighborhood radius) of Kittler's, Niblack's, Otsu's, Porte's, Sauvola's and Wolf's algorithms.

Each cell (y-row, x-column) of the pairwise tables contains two values n_{yx} and p_{yx}. The number n_{yx} represents the times that the algorithm y has a higher score than the algorithm x, while $p_{yx} = n_{yx}/(n_{yx}+n_{xy})$ represents the conditional probability of y's score being higher than x's score. We based our observations on the values p_{yx}. We ascertain that algorithm x is better than algorithm y if $n_{yx} \geq 1.33 n_{xy}$, which is equivalent to $p_{yx} \geq 0.57$.

5. Results

We arranged the test images such that the graph of AC performance is decreasing. That is $x_1^* \geq \cdots \geq x_n^*$. However, Fig. 11(a) shows that AC performance of Otsu's and T-CCD-L methods fluctuates irregularly although x_i^*'s graph is decreasing. We cannot visually assess the binarization performance because of the fluctuations.

To deal with these fluctuations, we present graphs and statistics of points $(i/n, y_{z_{i,j},j})$ where $z_{i,j}$ are indexes such that $y_{z_{1,j},j} \geq \cdots \geq y_{z_{n,j},j}$ for each binarization method j.

UV measure penalizes eroded and overestimated foreground boundaries, but it also penalizes stains (ink stains and dark background spots) that are classified as background so that algorithms that compute foreground boundaries correctly and classify stains as foreground are highly scored, like Wolf's thresholding that is the best in terms of UV measure, see Table 3.

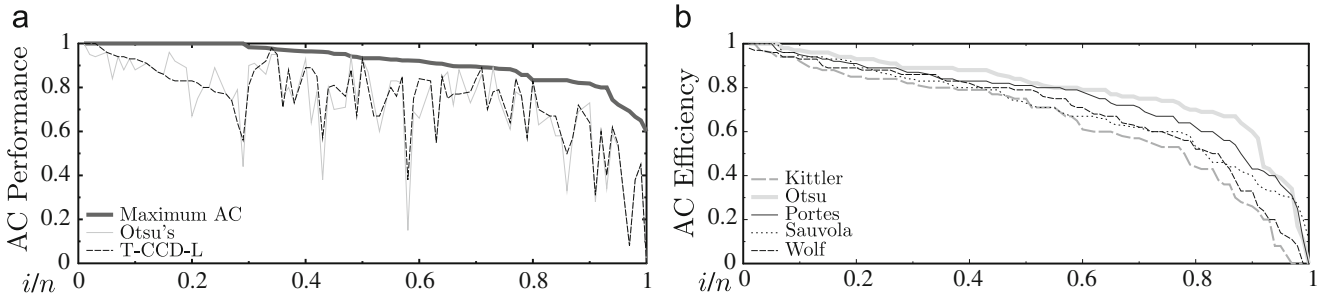

Fig. 11. (a) The graph of the maximum AC performance is decreasingly arranged. (b) Ordered AC efficiency of no-transition algorithms.

Table 3
UV pairwise comparison.

	Kittler	Otsu	Portes	Sauvola	Wolf	T-CCD-L	T-CCD-N	T-DF-L	T-DF-N
Kittler	–	72 0.84	35 0.41	**86 1.00**	29 0.34	30 0.35	56 0.65	32 0.37	58 0.67
Otsu	14 0.16	–	12 0.14	85 0.99	7 0.08	4 0.05	8 0.09	4 0.05	6 0.07
Portes	51 0.59	74 0.86	–	**86 1.00**	**36 0.42**	46 0.53	61 0.71	44 0.51	60 0.70
Sauvola	0 0.00	1 0.01	0 0.00	–	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
Wolf	57 0.66	79 0.92	**50 0.58**	**86 1.00**	–	**50 0.58**	66 0.77	**52 0.60**	68 0.79
T-CCD-L	**56 0.65**	**82 0.95**	40 0.47	**86 1.00**	**36 0.42**	–	**83 0.97**	44 0.52	79 0.92
T-CCD-N	30 0.35	78 0.91	25 0.29	**86 1.00**	20 0.23	3 0.03	–	4 0.05	47 0.56
T-DF-L	54 0.63	**82 0.95**	42 0.49	**86 1.00**	34 0.40	40 0.48	82 0.95	–	**81 0.94**
T-DF-N	28 0.33	80 0.93	26 0.30	**86 1.00**	18 0.21	7 0.08	37 0.44	5 0.06	–

The cell y-row, x-column shows how many times the algorithm y surpass the algorithm x in terms of UV measure. T-CCD-L and T-DF-N have even scores in a couple of images. The highest values are shown in bold.

However scattered stains and a slight overestimate of the foreground contour lead Wolf's method to low OCR performance, see Tables 4–6.

Kitller's thresholding also classifies stains as foreground but, contrary to Wolf's method, it overestimates the foreground boundaries greatly. In consequence, Kittler's method is a medium rank in terms of UV measure and reports the lowest AC efficacy because of the overestimated foreground boundary, see Fig. 11.

Sauvola's algorithms compute low thresholds, which discard stains from the foreground, but low thresholds also produce eroded foreground boundaries that are strongly penalized by UV measure (Table 3). As a result, Sauvola's method was the worst in terms of UV. What is more, this also affects the OCR performance badly (Table 3).

Portes's classifies stains as foreground frequently and overestimates the foreground contour slightly. In combination, this reduces the OCR performance but increases the UV measurements.

Otsu's and transition algorithms determine sharp foreground contours. However, Otsu's generated a great deal of stains in neighborhoods that are completely contained in the background despite the restriction of (36).

Transition algorithms differ as a product of two factors: the function that computes the transition thresholds and the function that computes the gray-intensity thresholds. Transition algorithm based on the complementary cumulative distribution resists more noise than those based on the density distribution so that both T-CCD-L and T-CCD-N generate less stains (penalized

Table 4
AC pairwise comparison.

	Kittler	Otsu	Portes	Sauvola	Wolf	T-CCD-L	T-CCD-N	T-DF-L	T-DF-N
Kittler	–	15 0.21	18 0.25	26 0.35	20 0.33	14 0.18	13 0.18	17 0.22	11 0.14
Otsu	58 0.79	–	44 0.62	**55 0.76**	52 0.78	20 0.34	23 0.41	29 0.45	22 0.39
Portes	53 0.75	27 0.38	–	37 0.51	40 0.61	21 0.30	27 0.38	25 0.36	24 0.35
Sauvola	48 0.65	17 0.24	35 0.49	–	38 0.54	19 0.26	20 0.27	25 0.32	17 0.25
Wolf	40 0.67	15 0.22	26 0.39	33 0.46	–	14 0.20	17 0.23	21 0.28	13 0.20
T-CCD-L	64 0.82	**38 0.66**	**49 0.70**	54 0.74	**56 0.80**	–	**29 0.62**	24 **0.60**	**27 0.54**
T-CCD-N	61 0.82	33 0.59	44 0.62	53 0.73	**56** 0.77	18 0.38	–	28 0.52	23 0.47
T-DF-L	61 0.78	35 0.55	44 0.64	52 0.68	53 0.72	16 0.40	26 0.48	–	25 0.42
T-DF-N	**66 0.86**	34 0.61	45 0.65	52 0.75	53 **0.80**	**23 0.46**	26 0.53	**34** 0.58	–

The sum $n_{xy}+n_{yx}\leq 86$ because some algorithms have even scores in several images. The highest values are shown in bold.

Table 5
PR pairwise comparison for text.

	Kittler	Otsu	Portes	Sauvola	Wolf	T-CCD-L	T-CCD-N	T-DF-L	T-DF-N
Kittler	–	20 0.24	32 0.39	44 0.52	32 0.41	20 0.24	21 0.25	21 0.25	21 0.25
Otsu	**65 0.76**	–	52 0.63	59 0.73	51 0.64	**32 0.42**	31 0.42	**37** 0.49	30 0.43
Portes	51 0.61	31 0.37	–	53 0.64	45 0.56	28 0.34	31 0.38	26 0.31	33 0.40
Sauvola	41 0.48	22 0.27	30 0.36	–	34 0.41	19 0.23	20 0.25	24 0.29	20 0.24
Wolf	46 0.59	29 0.36	35 0.44	49 0.59	–	25 0.31	27 0.33	27 0.33	26 0.33
T-CCD-L	62 **0.76**	**44 0.58**	55 0.66	**64 0.77**	**55 0.69**	–	**40 0.59**	31 **0.58**	**38 0.58**
T-CCD-N	62 0.75	42 **0.58**	51 0.62	61 0.75	**55** 0.67	28 0.41	–	33 0.49	32 0.52
T-DF-L	62 0.75	39 0.51	**57 0.69**	59 0.71	54 0.67	22 **0.42**	35 0.51	–	34 0.50
T-DF-N	63 0.75	39 0.57	50 0.60	62 0.76	53 0.67	28 **0.42**	30 0.48	34 0.50	–

The sum $n_{xy}+n_{yx}\leq 86$ because some algorithms have even scores in several images. The highest values are shown in bold.

Table 6
Mean, variance and quantiles of AC efficacy for each binarization method.

	Mean	Var.	Values i/n such that $y_{z_{i,j}}$ equal to or greater than								
			1.00	0.95	0.90	0.85	0.80	0.75	0.70	0.60	0.50
Kittler	0.646	0.261	0.02	0.05	0.10	0.21	0.37	0.50	0.55	0.65	0.78
Otsu	0.787	0.196	0.06	**0.16**	0.27	0.47	0.59	0.73	0.80	0.90	0.91
Portes	0.748	0.203	0.05	0.10	0.21	0.34	0.56	0.64	0.70	0.83	0.88
Sauvola	0.702	0.212	0.06	0.09	0.19	0.28	0.42	0.47	0.55	0.77	0.81
Wolf	0.691	0.246	0.00	0.06	0.14	0.34	0.47	0.57	0.60	0.74	0.84
T-CCD-L	**0.805**	**0.175**	0.08	0.13	0.30	0.48	**0.67**	**0.76**	**0.84**	0.91	**0.95**
T-CCD-N	0.798	0.182	0.08	0.13	**0.31**	0.45	**0.67**	**0.76**	0.79	0.90	**0.95**
T-DF-L	0.795	0.196	**0.09**	0.12	**0.31**	**0.51**	0.65	**0.76**	0.79	0.87	**0.95**
T-DF-N	0.796	0.189	0.08	0.15	0.28	**0.51**	0.63	0.72	0.81	**0.92**	0.94

The best values are shown in bold.

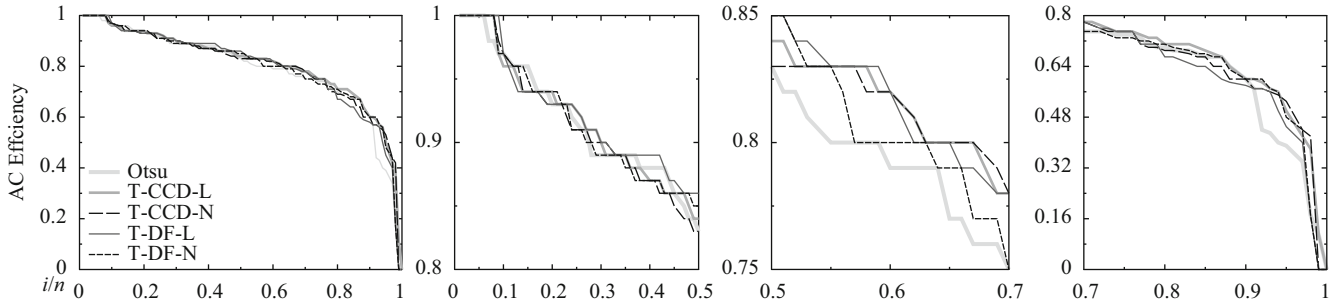

Fig. 12. Ordered AC efficacy graphs of transition algorithms. Visually, all four are almost indistinguishable from each other. However, T-CCD-L and T-CCD-N have the highest mean and lowest variance. Details of the left graph are shown in central and right graphs. Otsu's graph is plotted as a graph of reference.

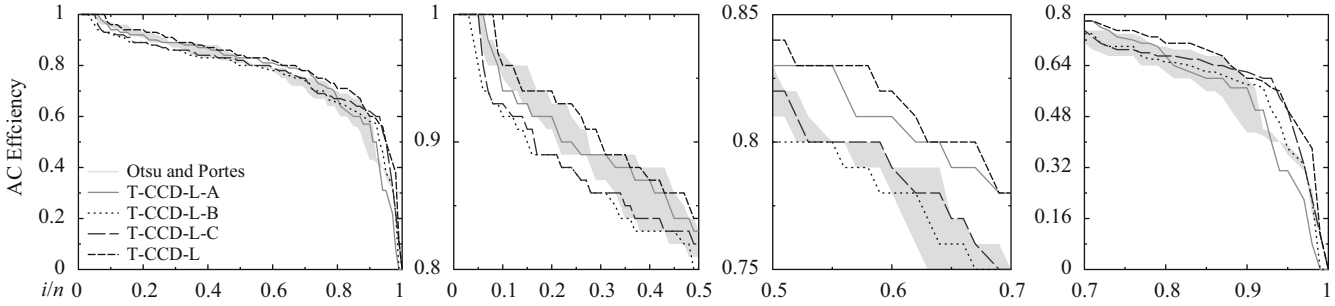

Fig. 13. Ordered AC efficacy graphs of T-CCD-L and variants. The area between Otsu's and Portes's graphs is plotted in light gray as reference.

Table 7
Mean, variance and quantiles of AC efficacy for T-CCD-L and variants.

	Mean	Var.	Values i/n such that $y_{z_{i,j}}$ equal to or greater than								
			1.00	0.95	0.90	0.85	0.80	0.75	0.70	0.60	0.50
T-CCD-L-A	0.771	0.213	0.06	0.09	0.24	0.44	0.64	0.73	0.79	0.87	0.92
T-CCD-L-B	0.758	0.186	0.3	0.05	0.15	0.33	0.55	0.69	0.76	0.88	0.93
T-CCD-L-C	0.771	0.175	0.5	0.06	0.16	0.36	0.59	0.70	0.73	**0.93**	**0.95**
T-CCD-L	**0.805**	**0.175**	**0.08**	**0.13**	**0.30**	**0.48**	**0.67**	**0.76**	**0.85**	0.91	**0.95**

The best values are shown in bold.

UV measurements) than T-DF-L and T-DF-N. Therefore, they have the highest AC efficacy, see means and variances in Fig. 12. On the other hand, transition algorithms based on lognormal threshold have a sharper foreground contour than those based on the normal threshold. Although these differences are visually minimal, they are reflected on the UV measurements.

We only present the influence of transition operator in T-CCD-L and variants A, B and C because the rest of transition methods are influenced in a similar manner.

The relative high variance of T-CCD-L-A and the graph behavior of the AC efficacy, see central-right graph of Fig. 13 and Table 7, suggest that T-CCD-L-A resists moderate noise.

Incidence and isolate transition operators increase the AC efficacy in images with high noise level at the cost of dropping the cardinality of transition set and, in consequence, the AC efficacy decreases in images whose foreground contains small connected component like punctuation marks and small characters. However, the incidence operator does not remove dense salt and pepper noise. Thus, it has to be applied after isolate transition operators.

Dilation transition operator counterbalances the unwanted effect of the incidence and isolate transition operators by increasing the cardinality of diminished transition sets. We should remark that this operator has to be applied in images with low noise level, or after isolate and incidence transition operators. Otherwise, the noise is magnified.

6. Conclusions

Our double-linear method is able to compute an accurate threshold which selects a representative samples of both foreground and background. We introduced three transition operators: both incidence and dilation transition operators refine transition set approximations, while the simple edge transition operator detects foreground edges.

Our test used a large database of text extracted from historical maps with different levels of degradation. We evaluated our experiments with four measures: an unsupervised evaluation method based on gray-intensity variances (UV measure) and three OCR measures (AC, PR measure and AC efficacy).

Our results point out that the transition method resists highest levels of noise. These tables also indicate that the complementary cumulative distribution function decreases the impact of outliers on the double-linear threshold and the use of lognormal threshold generates sharper foreground contours. Moreover, while the incidence transition operators can remove noise that isolate operators cannot, the dilation transition operator improves the performance of normal and lognormal thresholds.

Acknowledgments

We thank the anonymous referees for helpful and relevant comments. This research was partially supported by The National Council on Science and Technology (CONACYT) of Mexico (Grant number: 218253).

References

[1] H. Kohmura, T. Wakahara, Determining optimal filters for binarization of degraded characters in color using genetic algorithms, in: ICPR '06: Proceedings of the 18th International Conference on Pattern RecognitionIEEE Computer Society, Washington, DC, USA, 2006, pp. 661–664.
[2] C.-H. Chou, W.-H. Lin, F. Chang, A binarization method with learning-built rules for document images produced by cameras, Pattern Recognition 43 (4) (2010) 1518–1530.
[3] Y. Caron, P. Makrisa, N. Vincent, Use of power law models in detecting region of interest, Pattern Recognition 40 (2007) 2521–2529.
[4] R. Milewski, V. Govindaraju, Binarization and cleanup of handwritten text from carbon copy medical form images, Pattern Recognition 41 (4) (2008) 1308–1315.
[5] Q. Chen, Q.-s. Sun, P. Ann Heng, D.-s. Xia, A double-threshold image binarization method based on edge detector, Pattern Recognition 41 (4) (2008) 1254–1267.
[6] C. Mello, A. Sanchez, A. Oliveira, A. Lopes, An efficient gray-level thresholding algorithm for historic document images, Journal of Cultural Heritage 9 (2) (2008) 109–116.
[7] Ø.D. Trier, A.K. Jain, Goal-directed evaluation of binarization methods, Transactions on Pattern Analysis and Machine Intelligence 17 (12) (1995) 1191–1201.
[8] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic Imaging 13 (1) (2004) 146–168.
[9] J. Kittler, J. Illingworth, Minimum error thresholding, Pattern Recognition 19 (1) (1985) 41–47.
[10] J. Sauvola, M. Pietikinen, Adaptive document image binarization, Pattern Recognition 33 (2) (2000) 225–236.
[11] N. Otsu, A threshold selection method from grey-level histograms, IEEE Transaction on Systems, Man, and Cybernetics 9 (1) (1979) 62–66.
[12] M. Portes de Albuquerque, I. Esquef, A.G. Mello, M. Portes de Albuquerque, Image thresholding using Tsallis entropy, Pattern Recognition Letters 25 (2004) 1059–1065.
[13] C. Tsallis, Nonextensive Statistical Mechanics and Thermodynamics, Group of Statistical Physics, 2009. URL 〈http://tsallis.cat.cbpf.br/biblio.htm〉.
[14] W. Niblack, An Introduction to Digital Image Processing, Prentice-Hall, Birkeroed, Denmark, 1985.
[15] C. Wolf, J.-M. Jolion, Extraction and recognition of artificial text in multimedia documents, Pattern Analysis and Applications 3 (2003) 309–326.
[16] M.A. Ramírez-Ortegón, E. Tapia, L.L. Ramírez-Ramírez, R. Rojas, E. Cuevas, Transition pixel: a concept for binarization based on edge detection and gray-intensity histograms, Pattern Recognition 43 (2010) 1233–1243.
[17] L. Kish, Survey Sampling, Wiley Classics Library 1995 Edition, John Wiley & Sons, New York, 1965.
[18] A. Chaudhuri, H. Stenger, Survey Sampling Theory and Methods, second ed., M. Dekker, New York, 1992.
[19] C. Cortes, M. Mohri, M. Riley, A. Rostamizadeh, Sample selection bias correction theory, Algorithmic Learning Theory, vol. 5254, Springer, Berlin, Heidelberg, 2010, pp. 38–53.
[20] J. Huang, A.J. Smola, A. Gretton, K.M. Borgwardt, B. Schlkopf, Correcting sample selection bias by unlabeled data, in: B. Schölkopf, J. Platt, T. Hoffman (Eds.), Advances in Neural Information Processing Systems, vol. 19MIT Press, Cambridge, MA, 2007, pp. 601–608.
[21] M.E.J. Newman, Power laws, Pareto distributions and Zipf's law, Contemporary Physics 46 (46) (2005) 323–351.
[22] M.A. Ramírez-Ortegón, E. Tapia, R. Rojas, E. Cuevas, An estimator for the exponent parameter of power law distributions in overestimated measurements, submitted for publication.
[23] E. Roelant, S.V. Aelst, C. Croux, Multivariate generalized s-estimators, Journal of Multivariate Analysis 100 (5) (2009) 876–887.
[24] J. Agulló, C. Croux, S.V. Aelst, The multivariate least-trimmed squares estimator, Journal of Multivariate Analysis 99 (3) (2008) 311–338.
[25] S. Marchand-Maillet, Y.M. Sharaiha, Binary Digital Image Processing. A Discrete Approach, Academic Press, San Diego, 2000.
[26] T. MathWorks, MatLab, 2009. URL 〈http://www.mathworks.com/trademarks〉.
[27] J. Canny, A computational approach to edge detection, Pattern Analysis and Machine Intelligence 8 (1) (1986) 679–698.
[28] R. Maini, J. Sohal, Performance evaluation of Prewitt edge detector for noisy images, International Journal on Graphics, Vision and Image Processing 6 (3) (2006) 39–46.
[29] W. Janszoon, J. Blaeu, Theatrum Orbis Terrarum, Sive, Atlas Novus, Blaeu Atlas, 1645. URL 〈http://www.library.ucla.edu/yrl/reference/maps/blaeu〉.
[30] P. Stathis, E. Kavallieratou, N. Papamarkos, An evaluation technique for binarization algorithms, Journal of Universal Computer Science 14 (18) (2008) 3011–3030.
[31] T. Soft, Top OCR, Top Soft, 2008. URL 〈http://www.topocr.com/〉.
[32] S.B. Needleman, C.D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, Journal of Molecular Biology 48 (3) (1970) 443–453.

Marte Alejandro Ramírez Ortegín is a Ph.D. student at the Department of Mathematics and Computer Science of the Freie Universität Berlin. In 2002 he received a B.S. degree (Computer Science) from the University of Guanajuato (UG)/the Centre for Mathematical Research (CIMAT), Guanajuato, Mexico (1998–2002). His current research fields include pattern recognition, edge detection, binarization, multithresholding and text recognition.

Ernesto Tapia is a researcher of Artificial Intelligence group at the Department of Mathematics and Computer Science of the Freie Universität Berlin. He received a Ph.D. degree from the Freie Universität Berlin (Berlin, Germany) for his work on handwriting recognition. Dr. Tapia holds an M.Sc. degree in Applied Mathematics from the Centre for Mathematical Research (CIMAT), Guanajuato, Mexico, and a B.Sc. degree from the Universidad Veracruzana (UV), Mexico. His current research fields include pattern recognition, image processing neural networks, robotics and e-learning.

Raúl Rojas received the B.S. and M.S. degrees in Mathematics from the National Polytechnic Institute (IPN), Mexico City, Mexico. He obtained the Ph.D. degree at the Freie Universität Berlin, Berlin, Germany. Currently, Dr. Rojas is a full professor in Computer Science at the Freie Universität Berlin and leader of the Work Group on Artificial Intelligence at that University. His research interests include artificial intelligence, computer vision, and robotics.

Erik Cuevas received the B.S. and M.S. degrees in Computer Engineering from the University of Guadalajara, Guadalajara, Mexico, in 1996 and 2000, respectively. He received the Ph.D. degree in Computer Engineering at the Freie Universität Berlin, Berlin, Germany, in 2006. Currently, he is a full professor in the Computer Science Department at the University of Guadalajara. His research interests include computer vision, learning systems, and soft computing.