A robust thresholding algorithm for unimodal image histograms

Nicolas Coudray, Jean-Luc Buessler, Jean-Philippe Urban

A robust thresholding algorithm for unimodal image histograms

Nicolas Coudray, Jean-Luc Buessler, Jean-Philippe Urban

Université de Haute-Alsace, Laboratoire MIPS, 4 rue des Frères Lumière, 68093 Mulhouse, France

{nicolas.coudray, jean-luc.buessler, jean-philippe.urban}@uha.fr

corresponding author: Jean-Philippe Urban, jean-philippe.urban@uha.fr

Tel : (+33)-3-89-33-64-32

Fax : (+33)-3-89-33-60-84

Abstract

This article introduces a method to determine the threshold of unimodal image histograms in a robust manner. It is based on a piecewise linear regression that finds the two segments that fit the descending slope of the histogram. The algorithm gives a good estimation of the threshold, and is practically insensitive to the noise distribution, to the quantity of objects to segment, and to random histogram fluctuations.

Keywords

automatic thresholding; image histogram; unimodal distribution; edge detection

1 Introduction

The study of electron microscopy images of biological specimens, which are especially noisy and low contrasted, has lead us to develop a new method to automatically threshold the edges in a robust manner, compatible with a multiresolution approach. The objective is that the threshold remains reliable whatever the amount of edges, the noise distribution or the histogram statistical fluctuations. These parameters vary considerably between different acquisitions and different resolutions.

We propose a general thresholding method called the *T-point algorithm*. The original minimization criterion to set the threshold is studied in this article. The algorithm is simple, has low computational complexity, and requires little *a priori* knowledge concerning the distribution and proportion of pixels to segment.

To achieve these objectives, a novel minimization criterion is proposed. It consists of the minimization of the error between the descending slope of the histogram and its piecewise linear regression. It belongs to the histogram-based algorithms, but, unlike the main available methods, its computation relies on a wide portion of the histogram rather than a single bin or a few local bins.

The main automatic thresholding algorithms are reviewed in section 2. The proposed method is described in section 3. Experiments and results are discussed in section 4. Examples of applications are given in section 5.

2 Thresholding of unimodal histograms

Setting thresholds is a non-trivial problem. Ideally, each coherent set of pixels of the image is characterized by its gray-level and the histogram presents several corresponding maxima (modes): the threshold can simply be placed at the local minima between the peaks of the histograms, detected either directly or indirectly, or it can be placed by analyzing the statistics of the classes to minimize (Otsu 1979; Hou *et al.* 2006; Nakib *et al.* 2008). However, in many cases, when the sets are not clearly distinguished, the histogram becomes unimodal.

Figure 1 illustrates a unimodal histogram of an image composed of two sets of pixels to be separated: *SET 1* and *SET 2*. Two non-exclusive reasons explain unimodality: *SET 2* is very small compared to *SET 1*, and/or the overall Signal to Noise Ratio (SNR) is low (both sets share many gray-levels). These types of histograms can be viewed in low contrasted images (e.g., in microscopy images (Baradez *et al.* 2004)), but they are more frequently observed in high-pass filtered images where *SET 2* represents the segmentation of the edges, and *SET 1* the noise to be removed. The gray-level of the edges being higher than the mean gray-level of the noise, the edges contributes mainly to the upper elongated tail of the histogram, simply referred to here as the tail. The threshold needs to be placed to select as many edges as possible, but with a low amount of false positives (pixels from *SET 1* classified in *SET 2*).

Figure 1 Unimodal histogram composed of two sets of pixels: SET 1 (noise) and SET 2 (edges)

Generally, unimodal histograms result from gradient magnitude images. Edges contribute mainly to the tail of the histogram: they have generally a higher gradient magnitude than homogeneous regions, they are bound to be in minority in the images, and their amplitude can vary over a wide range. To separate the peak of the histogram from the tail, a simple thresholding method called *p-tile* is frequently implemented in software packages: it considers that edges represent a percentage of the pixels in the image (typically 10-20 % of the more contrasted pixels). This method requires however *a priori* knowledge often unavailable and this simple approximation can lead to highly erroneous results.

Other automatic methods make assumptions on the noise, the shape of the histogram, and other parameters. Voorhees and Poggio (1987) use a technique based on statistical properties: given an initial image corrupted with Gaussian noise, the probability law of its gradient image can be described as a Rayleigh distribution (detailed in section 3.1). Consequently, the threshold T is expressed by:

$$T = g_M \times \sqrt{-2\ln(P_F)} \; . \qquad (1)$$

with g_M the position of the histogram peak, and P_F the allowed proportion of false edges. The peak position of the distribution is proportional to the amount of noise and, as expressed in (1), the precision of the threshold detection is directly related to the precision of the peak detection.

Among the parametric methods, histogram-fitting approaches have been studied; the technique developed by Henstock and Chelberg (Henstock and Chelberg 1996) relies on a statistical model based on the weighted sum of two gamma density distributions. However, parameters need to be set and the edges may not always lead to a gamma distribution when the image contains many edges of various contrasts. The method used by Wiltschi *et al.* (Wiltschi *et al.* 2000) also requires a decision from the user: the first derivative of the histogram is computed to search for a specific slope value determined empirically.

Histogram-shape-based methods represent another set of approaches. Tsai (1995) developed a method based on the maximum local curvature to detect discontinuities, and like other methods (e.g. Antoine *et al.* 2001), requires a preliminary smoothing of the histogram. Smoothing should however be avoided: it sometimes alters the information too much and the choice of the appropriate filter and kernel size may be difficult to determine (Baradez *et al.* 2004). Furthermore, the searched discontinuity corresponds to the intersection of the two overlapping distributions. It can lead to the selection of an important amount of false positives, for instance when the SNR is low and the overlapping of the two distributions important. Our goal is not to minimize the classification error of both classes, but to limit the amount of false positives in *SET 2*.

Another shape-based method has been proposed more recently (Rosin 2001). Rosin developed an interesting geometrical method inspired from the triangle algorithm of Zack *et al.* (1977). A straight line is drawn from the maximum to the end of the histogram, to form a triangular-like shape (Figure 2). The threshold is selected at the point of the histogram that maximizes the perpendicular distance from the histogram to the straight line. This method gives suitable results in many cases, but tends to be sensitive to parameters such as the statistical fluctuations of the histogram, and the position of the endpoint (highest gray level).

Figure 2 Triangular method proposed by Rosin.

3 The *T*-point algorithm

We introduce a robust and fully automatic technique to determine the threshold of unimodal distributions. After a presentation of the notations, the *T*-point algorithm is described along with a computationally convenient implementation of the algorithm.

3.1 *Notations and Fundamentals*

A discrete histogram is constructed with the N pixels of a considered image, discretized on bins $[1, L]$. Bin i refers to gray level g_i with a precision of $\pm w/2$, where w is the bin width defined by:

$$w = g_{i+1} - g_i . \qquad (2)$$

Let h_i be the number of pixels in bin i, with gray levels in the interval $[g_i - w/2, g_i + w/2[$. Therefore:

$$\sum_{i=1}^{L} h_i = N . \qquad (3)$$

This histogram is unimodal if its slope is increasing on bins $[1, M]$ and decreasing on bins $[M, L]$, where M is the bin position of mode

$$M = \arg\max_i (h_i), \qquad (4)$$

and g_M the corresponding gray-level for bin M.

Theoretical distribution

The threshold must select the pixels whose value is significantly above the noise. It is therefore determined by the statistical noise distribution in the image. A Rayleigh distribution can be established, assuming additive Gaussian noise. A theoretical distribution will be used in this article to test the results of the algorithm on known distributions.

Generally, histograms become unimodal after a transformation of the image, e.g. gradient analysis, with $\|\nabla I\|$ the norm of the gradient of image I. Let I be corrupted with an additive Gaussian noise of variance σ_n^2. Since a linear combination of m Gaussian random variables leads to a Gaussian distribution of variance $\sum_1^m \sigma_i^2$, a 3×3 Prewitt filter on the corresponding gradient images ∇I_x and ∇I_y in the x- and y-direction leads to a variance of:

$$\sigma_R^2 = 6 \times \sigma_n^2 . \qquad (5)$$

$\|\nabla I\|$ is obtained through:

$$\|\nabla I\| = \sqrt{\nabla I_x^2 + \nabla I_y^2} . \qquad (6)$$

Based on probability theory, the $\|\nabla I\|$ gray-level distribution can be considered as a Rayleigh distribution (Pitman 1993), represented by:

$$f(g) = g \times \exp\left(\frac{-g^2}{2 \times \sigma_R{}^2} \right) \Big/ \sigma_R{}^2 . \qquad (7)$$

The corresponding cumulative density function, defined by

$$P_F(g) = \int_T^\infty f(g) dg = \exp\left(\frac{-T^2}{2 \times \sigma_R{}^2} \right) \qquad (8)$$

gives the probability P_F of false edges to be above threshold T. Parameter σ_R corresponds to the position g_M of the peak of the distribution. Useful information can be extracted by simple histogram analysis. As a result of (8), T can be computed from the values of g_M and P_F, a property used by Voorhees and Poggio (1987) in (1).

Normalized threshold

To facilitate the comparison of the thresholds computed on different histograms, the normalized threshold T_N is defined by:

$$T_N = T / \sigma_R . \qquad (9)$$

It is specifically defined for testing purposes, where the amount of noise in the image is known. To segment edges with a probability of $P_F \approx 2$ % of false positive, then $T_N \approx 2.8$ whatever the amount of noise in the image.

Histograms as empirical distributions

A histogram corresponds to the estimation of the underlying distribution f^* of the image. Each bin is a random variable whose variance can be related to N and w.

The bin probability $p^*(i)$ is given by:

$$p^*(i) = \int_{g_i - w/2}^{g_i + w/2} f^*(y)dy \simeq w \times f^*(g_i) . \qquad (10)$$

The expected number of pixels h_i^* falling in bin *i* is:

$$h_i^* = p^*(i) \times N \simeq w \times N \times f^*(g_i) . \qquad (11)$$

h_i is an estimation of h_i^* measured from the image and expressed by:

$$h_i = p\ (i) \times N , \qquad (12)$$

$p(i)$ being the estimated probability of a pixel to fall in bin *i*. Its expectation is:

$$E\left[h_i \right] = E\left[p(i) \right] \times N . \qquad (13)$$

Since $p(i)$ has a binomial distribution that tends to a Poisson distribution,

$$E\left[p(i) \right] = p^*(i) , \qquad (14)$$

and

$$\sigma^2\left[p(i) \right] = p^*(i) . \qquad (15)$$

Therefore,

$$E\left[h_i \right] \approx w \times N \times f^*(g_i) \qquad (16)$$

and the standard deviation is

$$\sigma\left[h_i \right] = \sqrt{w \times N \times f^*(g_i)} . \qquad (17)$$

The relative standard error is finally defined by

$$\sigma\left[h_i \right] \Big/ E\left[h_i \right] = 1 \Big/ \sqrt{w \times N \times f^*(g_i)} . \qquad (18)$$

From this equation, it can be seen that the relative value in bin *i* varies more when *N* and *w* are small. If choosing a large bin width reduces the local fluctuations, it also reduces the resolution at which the threshold can be set. On the other hand, a small width intensifies the statistical fluctuations and may generate a statistical error in the estimation of the threshold if the algorithm is not robust. We will see in section 4 that the proposed *T*-point algorithm allows a robust setting of the threshold.

3.2 *Description of the algorithm*

Our method assumes that: 1) a dominant pixel population produces the main peak of the histogram, i.e. the noise on the homogeneous regions in a gradient magnitude image; 2) a secondary population contributes mainly to the tail of the histogram, i.e. the edges. The threshold *T*, desired to select a maximum of edges while ensuring a minimum of false detection, is presumed to be located somewhere after the position of the maximum of the histogram and before the elongated part of the tail. The descending slope can be decomposed into two parts: a steep descending part immediately after the peak and a slightly descending part in the flat tail (Figure 3A). Then, we suggest looking for the two lines that best describe these two descending slopes. The point of abscissa that minimizes the error between the descending slope and the two lines, called the breakpoint, is chosen as the threshold. This method is referred to as the *T-point* algorithm, corresponding to the point where the line representation switches from the steep to the slight slope line.

Figure 3 Principle of the *T*-point algorithm: A/ Histogram; B/ Square errors $\varepsilon_1(k)$ and $\varepsilon_2(k)$; C/ Total error $\varepsilon(k)$; D/ Histogram with the threshold *T*, and with the corresponding best two fitted lines.

The study considers to the descending slope defined on bins $[M, L]$ (Figure 3A). Each bin $k \in]M, L[$ is successively considered as the potential breakpoint. For each k , the two lines are computed using linear regressions: using the least squares method, line $\mathcal{D}_1$ is identified for the left part of the descending slope computed with bins $[M, k]$ and line $\mathcal{D}_2$ is computed on bins $[k+1, L]$. The segmented linear estimation $\hat{h}_i(k)$ of bin *i* for a given *k* can be written as:

$$\hat{h}_i(k) = \begin{cases} a_1(k) \times g_i + b_1(k) & if\ i \in [M, k] \\ a_2(k) \times g_i + b_2(k) & if\ i \in]k, L] \end{cases} \qquad (19)$$

where $a_1(k)$, $b_1(k)$ and $a_2(k)$, $b_2(k)$ are the respective parameters of the lines $\mathcal{D}_1$ and $\mathcal{D}_2$.

For each *k*, errors between the fittings and the histogram are computed (Figure 3B). The sum of the square errors for $\mathcal{D}_1$ can be expressed as:

$$\varepsilon_1(k) = \sum_{i=M}^{k} (h_i - \hat{h}_i(k))^2 , \qquad (20)$$

and for $\mathcal{D}_2$:

$$\varepsilon_2(k) = \sum_{i=k+1}^{L} (h_i - \hat{h}_i(k))^2 \,. \qquad (21)$$

The total error $\varepsilon(k)$ of the fittings, represented on Figure 3C, is the sum of the errors:

$$\varepsilon(k) = \varepsilon_1(k) + \varepsilon_2(k) = \sum_{i=M}^{k} \left(h_i - a_1(k) \times g_i - b_1(k)\right)^2 + \sum_{i=k+1}^{L} \left(h_i - a_2(k) \times g_i - b_2(k)\right)^2 \,. \qquad (22)$$

The threshold T is set at the breakpoint that minimizes the above cost function and therefore best satisfies the piecewise linear fitting (Figure 3D):

$$T = \arg\min_k(\varepsilon(k)) \,. \qquad (23)$$

Fast algorithm

To ease the implementation of this cost function, a more efficient computational equation can be derived for $\varepsilon_1(k)$ and $\varepsilon_2(k)$ (Allen 1997). First, it can be shown, using general equations, that:

$$\begin{aligned} \varepsilon(n) &= \sum_{i=1}^{n} \left(y(i) - \hat{y}(i)\right)^2 \\ &= \sum_{i=1}^{n} y(i)^2 - \sum_{i=M}^{k} \hat{y}(i)^2 \\ &= n \times \mathrm{var}\left(y(i)\right) - \alpha \times n \times \mathrm{var}\left(x(i)\right) \\ &= n \times \mathrm{var}\left(y(i)\right) - n \times \frac{\mathrm{cov}\left(x(i), y(i)\right)^2}{\mathrm{var}\left(x(i)\right)} \end{aligned} \,. \qquad (24)$$

Second, let's consider the intermediate variables defined by the sum of the first *k* values:

$$S_x(k) = \sum_{i=M}^{k} x(i), \quad S_{xx}(k) = \sum_{i=M}^{k} x(i)^2, \quad S_y(k) = \sum_{i=M}^{k} y(i), \quad S_{xx}(k) = \sum_{i=M}^{k} y(i)^2, \quad S_{xy}(k) = \sum_{i=M}^{k} x(i) \times y(i) \,. \qquad (25)$$

It can be noticed that these variables can be evaluated recursively, e.g. $S_x(k+1) = S_x(k) + x(k+1)$. From (24) and (25), $\varepsilon_1(k)$ can be expressed as a function of these variables:

$$\varepsilon_1(\mathrm{k}) = \mathrm{S}_{\mathrm{yy}}(\mathrm{k}) - \frac{S_y(\mathrm{k})}{n} - \frac{\left(n \times S_{xy}(\mathrm{k}) - S_x(\mathrm{k}) \times S_y(\mathrm{k})\right)^2}{n \times \left(n \times S_{xx}(\mathrm{k}) - S_x(\mathrm{k})^2\right)}, \qquad (26)$$

with *n* the number of elements used for the summations in (25). Error $\varepsilon_2(k)$ is determined in a similar manner, but for (25), since *i* will go from *k+1* to *L*, variables are evaluated iteratively by counting down from *L* to *M+1*. The algorithm principle is given in Table 1. The cost function can be defined on interval $[M, L]$ with a complexity *O*(*n*), *n=L-M+1*. The complexity is thus proportional to the number of bins.

Table 1 Computationally convenient algorithm for the search of the optimal breakpoint.

4 Experiments and results – Robustness analysis

We have verified experimentally that the proposed method possesses the required characteristics to segment unimodal histograms:

- the minimization criterion $\varepsilon(k)$ has an absolute minimum, which determines a suitable threshold T for the images;
- the method segments edges, keeping the amount of false edge at a low percentage P_{F} of false edges; the threshold is robust to segment unimodal histograms for different quantities of edges and different contrasts.
- Applied on histograms of random Rayleigh distributed variables, the threshold set with the T-point algorithm is found at $T_n \approx 2.8$ ($P_F \approx 2\%$).

Threshold T_n needs to be kept constant to keep the amount of false edges stable and allow the identification of possible low contrasted edges. Through the experiments presented in this section, we will see that the proposed method is almost insensitive to statistical variations of the histogram.

A comparison with two of the main techniques is undertaken: Voorhees and Poggio's (1987) statistical method, and Rosin's (2001) geometrical method, called here the triangular method. The method proposed by Rosin has been chosen because it is one of the most recent algorithms and its performances has been well rated in Medina-Carnicer *et al.* (2005) where it is compared to other thresholding methods in the context of edge-detection. For the statistical method, the threshold is set such as $T = 2.8 \times g_M$ to have a probability P_{F} of segmenting false edges of 2 % (see (1)).

In the next two subsections, the robustness of the method is validated on Rayleigh distributions of gradient images obtained from images corrupted with a Gaussian noise. The method is extended to other distributions through an example given in the section 5.

4.1 *Influence of image size and bin width*

This subsection shows that the proposed method is robust to the statistical fluctuations of the histograms induced by small size images and small bin widths (see (18)).

The tests have been made on histograms of gradient magnitude images $\left\|\nabla I\right\|$, I being corrupted with Gaussian noise. Histograms have been normalized so that the theoretical position of the peak is $\sigma_R = 1$. The bin widths w of the histograms have been selected from 0.01 to 0.13 (corresponding to bin numbers from 500 to 60). The image size has been varied from $N = 64 \times 64$ pixels to $N = 1024 \times 1024$ pixels. For each set of conditions (a given size and a given bin width), 100 noisy images were processed to extract statistical data about the mean threshold μ_T and the standard deviation s_T of the threshold (Figure 4).

Figure 4 shows that the mean thresholds μ_T measured with the three algorithms are quite constant. The standard deviation s_T shows that the thresholds are less stable for small values of N and w. The figure illustrates that, for a given N and w, the proposed method surpasses the two others in robustness.

The resolution w of the histogram is a parameter defined by the user. It should be small to select the threshold with precision, but a small w increases s_T, and a compromise has to be found. As the proposed algorithm is weakly influenced by the fluctuations of the histogram, the threshold is more constant, the algorithm is robust (s_T remains small for all w), and small bin widths can be used to find the threshold with a good resolution.

Figure 4 Repeatability of the estimated threshold. A/ mean threshold μ_T as a function of the image size; B/ mean threshold μ_T as a function of the bin width. Errorbars (s_T) are shown slightly offset around a given value in the x-axis to prevent overlapping.

4.2 *Influence of the quantity and contrast of edges*

In this section, we check if the threshold is constant in terms of percentage of false edge pixels segmented, whatever the quantity and contrast of the edges.

Synthetic images $I_{\sigma,p}$ have been created such as

$$I_{\sigma,p} = \sigma_n + I_{p\%}, \qquad (27)$$

with σ_n the standard deviation of the Gaussian noise added and $I_{p\%}$ the image of the objects leading to p% of edge pixels in the gradient magnitude image $\left\|\nabla I_{\sigma,p}\right\|$. The standard deviation of the noise has been varied from σ_n=5 to 50. To vary the quantity of edge pixels, synthetic images of squares have been created (example in

Figure 5). The number and size of squares in the images have been changed to vary p from 0 to 8 %. The values of σ_n have been taken on a large interval to test the limits of the algorithms: a low SNR involves that the two sets for the distributions are very close and as we want to avoid false edge pixels, only a few true edge pixels will be segmented (Figure 6A). On the other hand, a high SNR lead to a bi-modal histogram if the amount of edges is important (Figure 6D).

Figure 5 Example of a synthetic image (gray-level of the background being 0, and gray-level of the objects from 40 (top left) to 80 (lower right)).

Figure 6 Histogram of synthetic gradient images corrupted with additive noise of standard deviation σ_n.Histogram of the whole image appear in plain black line and histogram of the edges appear in dotted blue lines.

For each condition (σ_n, p), 100 draws have been computed with random Gaussian noise. From these 100 images, the average normalized threshold value and its standard deviation have been extracted (Figure 7).

In Figure 7A, we see that, with the statistical method, the standard deviation of the experimental thresholds is very high (around 0.15). It can be easily proved that this method responds linearly to the imprecision of the detection of the peak position, which confirms that the peak position is poorly estimated. We notice that both geometrical methods (Triangular and T-point methods) are quite insensitive to errors on M.

For the triangular method, the standard deviation value is better (around 0.05). However, the SNR influences the average threshold, since the average T_N tends to be higher when σ_n decreases (Figure 7B), that is when the endpoint L is far from the peak. As the tail of the histogram corresponds to the edges, an endpoint far from the peak of the histogram means that there are well contrasted edges. As Rosin (2001) observed, the further L is from the peak, the higher the threshold. Nonetheless, such a histogram does not mean that there are no low contrasted edges. It is therefore important to keep the threshold in an acceptable range to avoid under-segmentation.

The T-point method resists better to noise variation than the triangular method, and the algorithm is much less sensitive to random fluctuations (standard deviation around 0.02). The function minimized in the triangular method depends only on a point-to-point distance from a line to the

histogram, whereas the *T*-point algorithm relies on the whole descending slope, which leads to a smaller sensitivity to the fluctuations of the bin values.

Figure 7 Influence of the quantity and contrast of edges on the automatic thresholds. Experimental mean value and standard deviation (errorbars) are displayed.

5 Applications

The *T*-point algorithm is first applied to a multiresolution edge detection on a synthetic image, then to threshold other unimodal distributions, and finally, to the multiresolution edge analysis of electron microscope images.

5.1 *Application to multiresolution edge-detection*

When the overlapping between the two sets is important, there is no efficient threshold to distinguish them without segmenting too many false pixels. In practice, separation between modes can be improved by smoothing methods, like multiresolution approaches where the image is analyzed at different resolutions (Coudray *et al.* 2007). In dyadic pyramidal cases, smoothing is followed by decimation (Lindeberg and Bretzner 2003). As the scale gets coarser, the size of the image is reduced, the proportion of edges in the image is increased and the histogram tends to be less smooth. As a result, a robust threshold has to be used to detect the edges through the scales.

To illustrate the evolution of the histograms and thresholds through the scales, the synthetic image presented in Figure 5 was used. Figure 8 shows the good performances of the *T*-point method proposed in this context, despite the fluctuations of the bin values. The amount of false edge pixels remains sufficiently low to segment as many edge pixels as possible while keeping a reasonable amount of false edge pixels.

Figure 8 Thresholding of gradient images (obtained from Figure 5) through the scales of a pyramidal transform, using the *T*-point. A/ scale 2 (image size: 512x512); B/ scale 3 (image size: 256x256); B/ scale 4 (image size: 128x128).

5.2 *Application to other unimodal distributions*

The robustness of the proposed method has been validated on known distributions: Rayleigh distributions obtained through the gradient transform of an image corrupted with a Gaussian noise. However, it must be noted that the *T*-point remains valid for other types of unimodal distributions, from images corrupted with other noise distributions, and after other filter transformations. Below, we illustrate this point with three examples.

a/

Figure 9 illustrates an image corrupted with both additive and multiplicative noise, and Figure 10 shows the quantity of missing and false edges segmented as the multiplicative noise is increased. With the method developed by Voorhees and Poggio (Figure 10A) only a few true positive edges are missing, but the amount of false edge pixels is too important (Figure 10B). Both geometrical methods are appropriate to segment the edges while keeping the amount of false positive constant. As the threshold set by the *T*-point is lower, the amount of false edge pixels is slightly higher than with the triangular method (2 % instead of 1 %), and more low contrasted edges are segmented (Figure 10A).

Figure 9 Edge thresholding of images corrupted with additive and multiplicative noise. A/ synthetic image (coded between 0 and 1) with an additive Gaussian noise (σ_n =0.02) and a normally distributed multiplicative noise (σ_m =0.06); B/ gradient image; edges segmented with C/ the method of Voorhees and Poggio, D/ the triangular method, and E/ the *T*-point.

Figure 10 A/ Percentage of true edge pixels not detected and B/ of false edge pixels detected on the histograms of the gradient image of

Figure **9** (with Gaussian noise of σ_n =0.02 and different multiplicative noise with zero-mean).

b/

Figure 11 shows an image of a biological membrane where heterogeneities have been identified using a local standard deviation filter (a non linear and non directional edge detector). Assuming the initial image corrupted with a Gaussian noise, the resulting image histogram can be considered as a Chi distribution. Results of the segmented images confirm that the method leads to good results. Tested on synthetic images, we evaluated the amount of false edges to be below 2 %.

Figure 11 Segmentation of heterogeneities. A/ Initial microscope image; B/ Image filtered with a 5x5 standard deviation filter; C/ Histogram of image B, and threshold set with the *T*-point; D/ Segmented image.

c/

Figure 12 displays results of segmentations applied directly to synthetic images corrupted with Gaussian noise. In the histograms used, *L* is the highest gray level in the image. The image of the first line contains one low contrasted object which has been segmented correctly. The amount of false pixels is low. On the second line, a high contrasted object has been added: since the triangular method is sensitive to *L*, the low contrasted object is poorly segmented, whereas the segmentation obtained with the *T*-point is stable.

Figure 12 Segmentation of objects in an image corrupted with Gaussian noise. First column: initial synthetic images; second column: results of the segmentation using the *T*-point; last column: results using the triangular method.

5.3 *Electron microscope images*

We have tested the *T*-point algorithm on the multiresolution approach proposed by Coudray *et al.* (2007) to analyze electron microscope images that are very noisy and where the contrast and the quantity of edges fluctuate a lot. Furthermore, the noise tends to be both additive and multiplicative. The multiresolution techniques developed require the edges to be detected at several resolutions of a pyramidal transform, but common thresholding methods perform poorly considering the characteristics of these images.

Pyramidal transformations generate images of smaller size where the noise is filtered and the proportion of contours is higher (Figure 13B through 13D). Figures 13E through 13G display the corresponding histograms with the *T*-point thresholds, and Figures 13H through 13J display the segmented images. We see that the threshold is adapted to detect more edges at each scale, and that the amount of noise remains reasonably low.

We have run tests on many different images acquired with different microscopes to verify that the *T*-point algorithm achieves the objectives, i.e. set a reliable threshold whatever the amount of edges, the noise distribution or the histogram statistical fluctuations. Furthermore, its efficient implementation (computation time

of about 0.2 ms for a 1000 bin histogram and a 1024 x 1024 pixels image) makes it well adapted for real-time applications.

Figure 13 Image segmentation in a multiresolution process.

6 Conclusion

An automatic method is proposed to select a threshold in unimodal histograms. The method is based on a piecewise linear regression of the descending slope of the histogram. The *T*-point is defined as the position of the breakpoint that minimizes the residual square error between the two fitted segments and the slope. By defining the cost function over the whole descending curve instead of some particular points only, the method proves to be more robust than other tested methods.

Results show that the method can be applied to different unimodal distributions which are induced by image pre-treatments, noise statistics, and the quantity of edges to be identified. The example related to electron microscope images further underlines this need for robustness. These images must be processed at different scales to threshold all the edges keeping the amount of false edges low at each scale. Constraints on this kind of images are important since the noise in not purely additive, the multiresolution transformations modify considerably the distribution, and the histograms have important statistical fluctuations at coarser scales, getting closer to bimodal distributions. The required efficiency of thresholding in this context underlines the interest of the *T*-point algorithm.

Acknowledgment

This work was supported by the EU 6th framework (HT-3DEM, LSHG-CT-2005-018811).

References

Allen, M. P., 1997. Understanding regression analysis. Kluwer Academic Publishers, New York.

Antoine, C., M. D. Lloyd J. Antoine, 2001. A robust thresholding algorithm for halftone dots. Journal of Pulp and Paper Science, 1-18.

Baradez, M.-O., C. P. McGuckin, N. Forraz, R. Pettengell A. Hoppe, 2004. Robust and automated unimodal histogram thresholding and potential applications. Pattern Recognition. 37(6), 1131-1148.

Coudray, N., J.-L. Buessler, H. Kihl J.-P. Urban, 2007. Multi-scale and first derivative analysis for edge detection in TEM images. 4th International Conference on Image Analysis and Recognition (ICIAR 2007), Montréal, Canada, Springer Lecture Notes in Computer Science LNCS 4633.

Henstock, P. V. D. M. Chelberg, 1996. Automatic gradient threshold determination for edge detection using a statistical model. A description of the model and comparison of algorithms. Purdue Libraries. Purdue, Indiana, USA, School of Electrical And Computing Engineering, Purdue University**:** 33.

Hou, Z., Q. Hu W. L. Nowinski, 2006. On minimum variance thresholding. Pattern Recognition Letters. 27(14), 1732-1743.

Lindeberg, T. L. Bretzner, 2003. Real-time scale selection in hybrid multi-scale representations. Scale-Space'03, Isle of Skye, Scotland, Springer Lecture Notes in Computer Science.

Medina-Carnicer, R., F. J. Madrid-Cuevas, N. L. Fernández-García A. Carmona-Poyato, 2005. Evaluation of global thresholding techniques in non-contextual edge detection. Pattern Recognition Letters. 26(10), 1423-1434.

Nakib, A., H. Oulhadj P. Siarry, 2008. Non-supervised image segmentation based on multiobjective optimization. Pattern Recognition Letters. 29(2), 161-172.

Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics. 9(1), 62-66.

Pitman, J., 1993. Probability. Springer-Verlag, New York.

Rosin, P. L., 2001. Unimodal thresholding. Pattern Recognition. 34(11), 2083-2096.

Tsai, D.-M., 1995. A fast thresholding selection procedure for multimodal and unimodal histograms. Pattern Recognition Letters. 16(6), 653-666.

Voorhees, H. T. Poggio, 1987. Detecting blobs as textons in natural images. Image Understanding Workshop, Los Angeles, Science Applications International Corporation, McLean, VA.

Wiltschi, K., A. Pinz T. Lindeberg, 2000. An automatic assessment scheme for steel quality inspection. Machine Vision and Applications. 12(3), 113-128.

Zack, G. W., W. E. Rogers S. A. Latt, 1977. Automatic measurement of sister chromatid exchange frequency. Journal of Histochemistry and Cytochemistry. 25(7), 741-753.

Figure1-Color

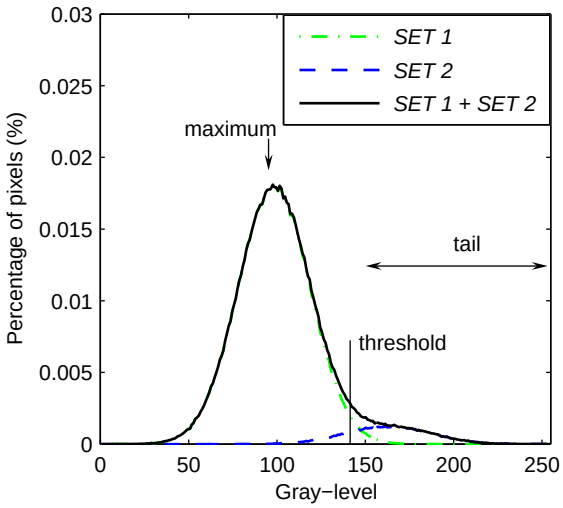


Figure1-GrayLevel

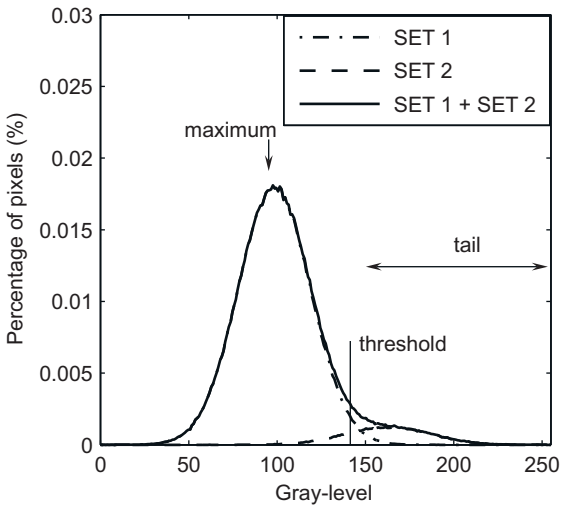


Figure2

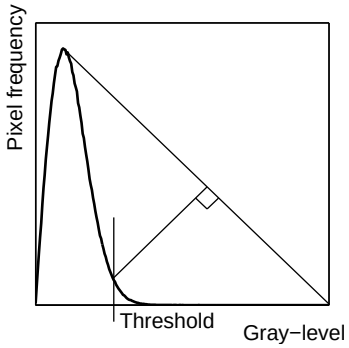
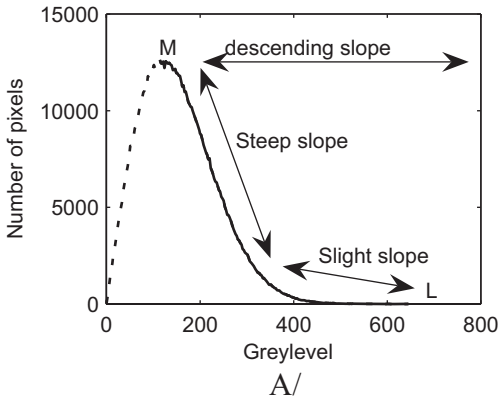
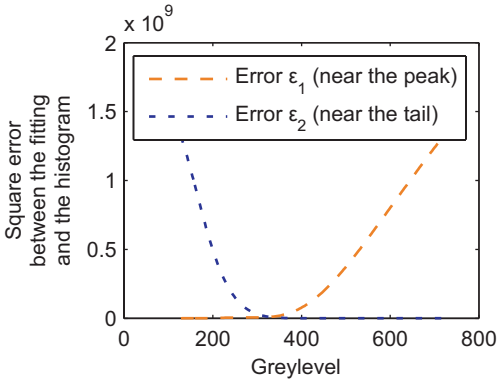


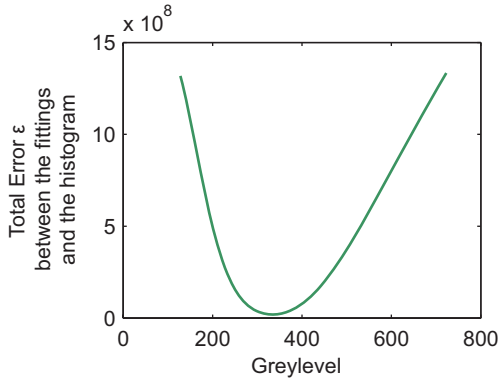
Figure3-Color



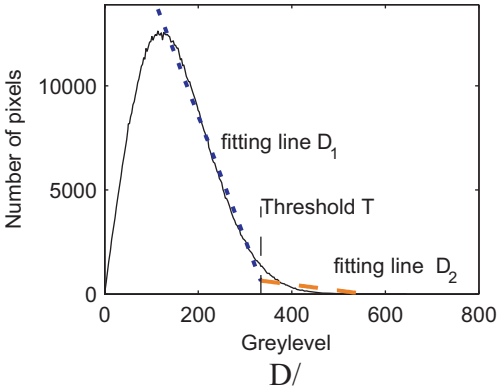
A/



B/

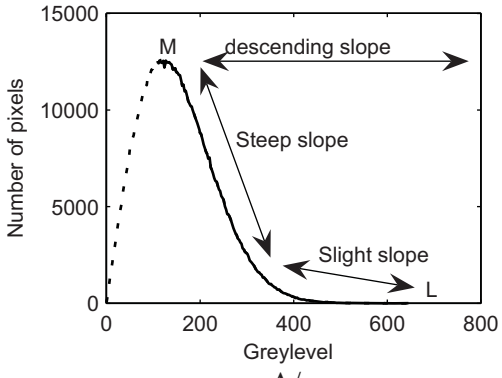


C/

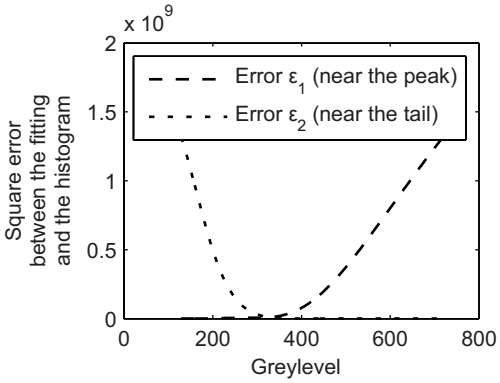


D/

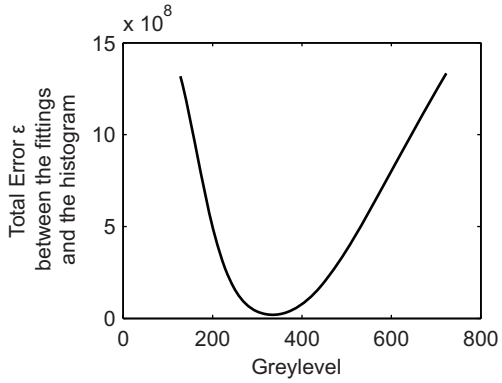
Figure3-GrayLevel



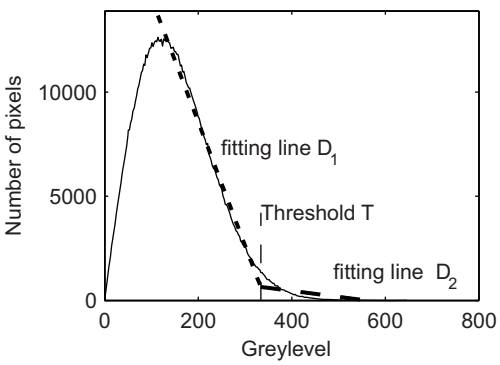
A/



B/



C/



D/

Figure4-Color

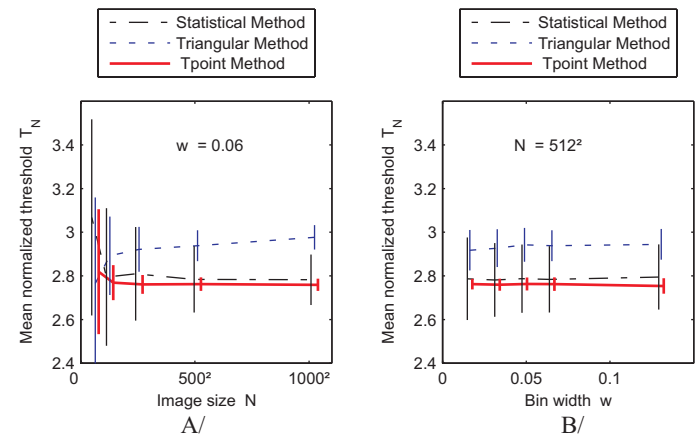


Figure4-GrayLevel

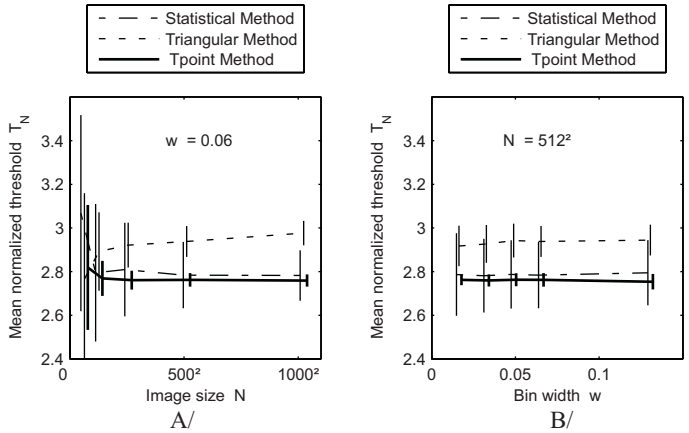


Figure5

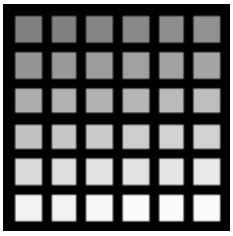
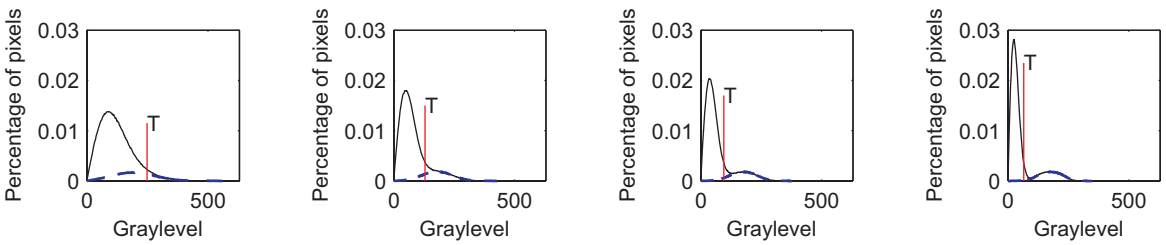
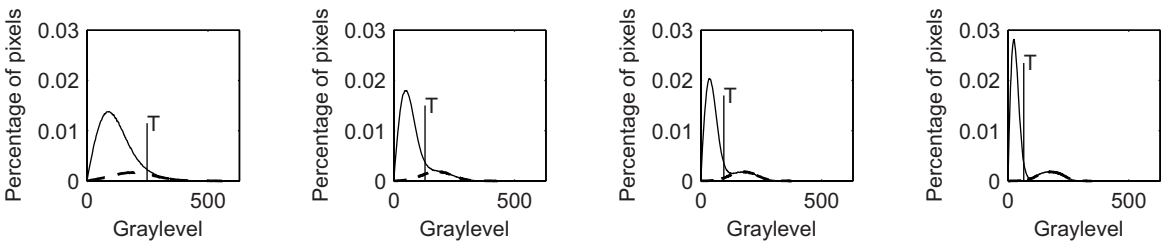


Figure6-Color



A/ $\sigma_n = 35$ B/ $\sigma_n = 20$ C/ $\sigma_n = 15$ D/ $\sigma_n = 10$

Figure6-GrayLevel



A/ $\sigma_n = 35$ B/ $\sigma_n = 20$ C/ $\sigma_n = 15$ D/ $\sigma_n = 10$

Figure7-Color

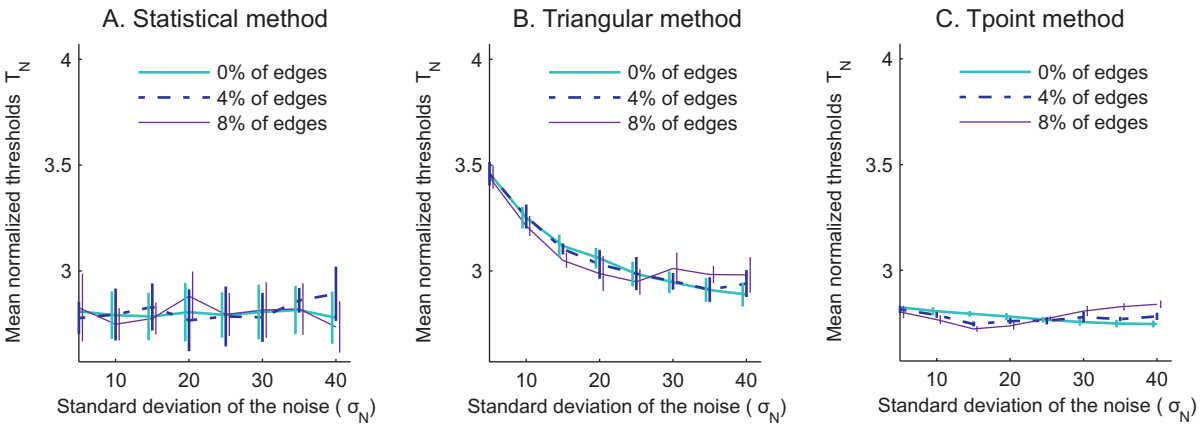


Figure7-GrayLevel

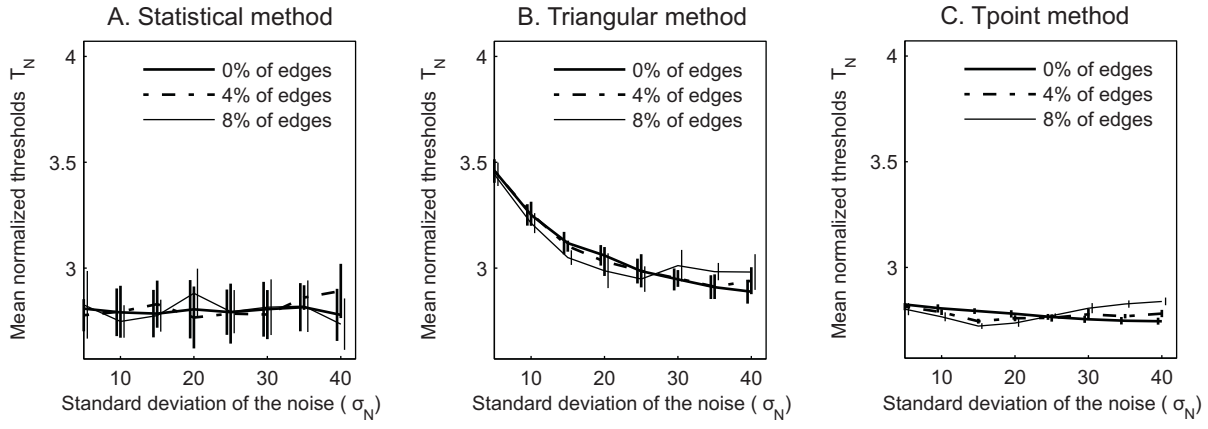
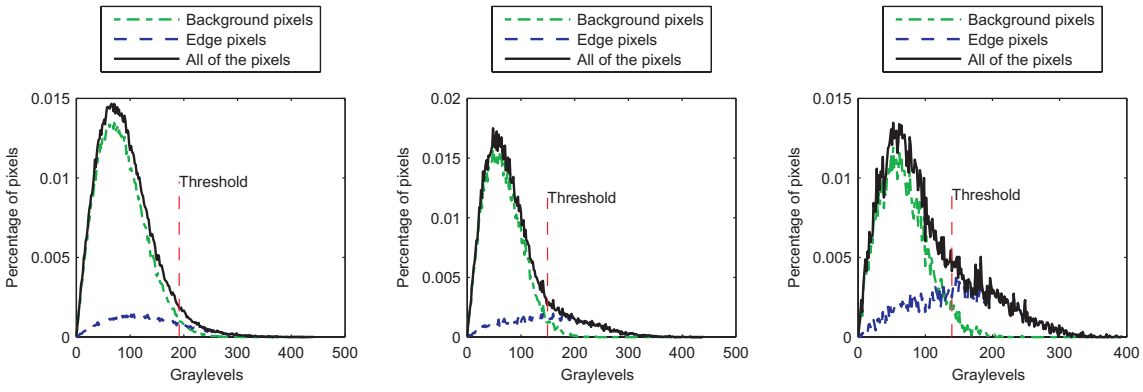
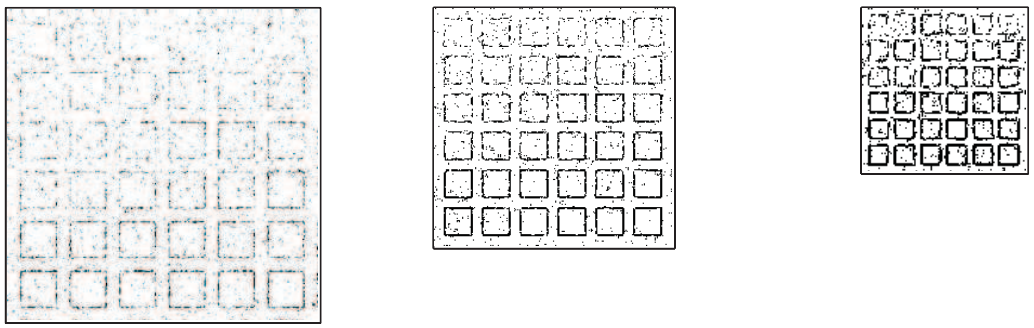


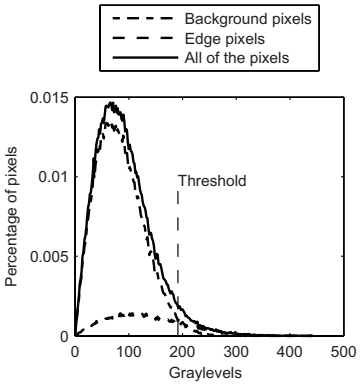
Figure8-Color

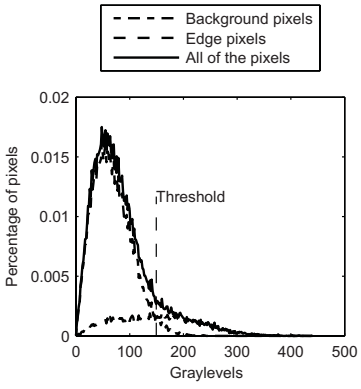


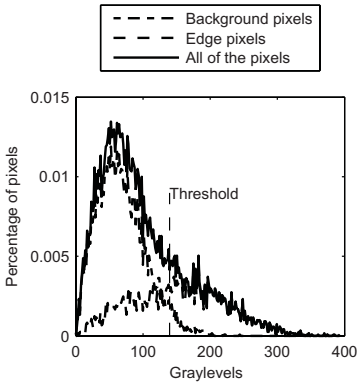


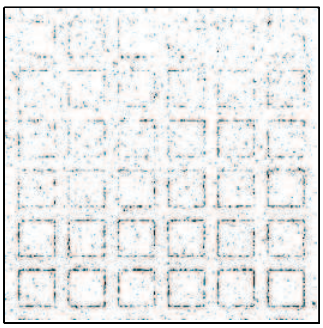
A/ B/ C/

Figure8-GrayLevel

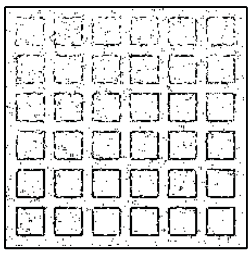




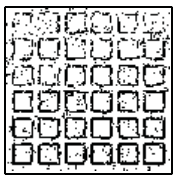




A/

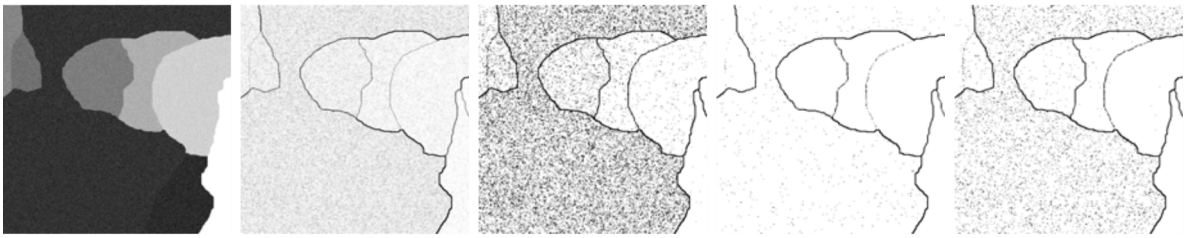


B/



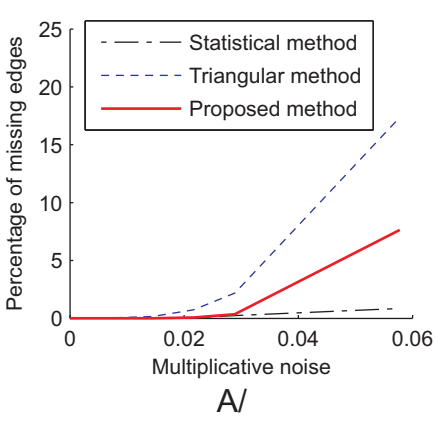
C/

Figure9



A/ B/ C/ D/ E/

Figure10-Color



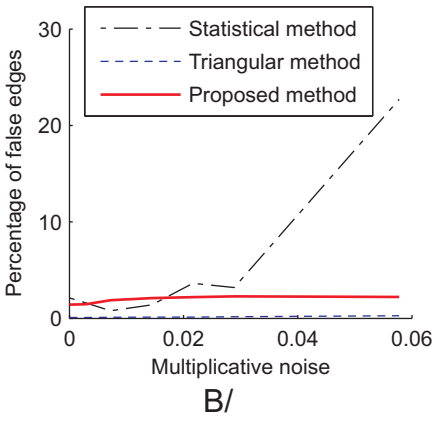
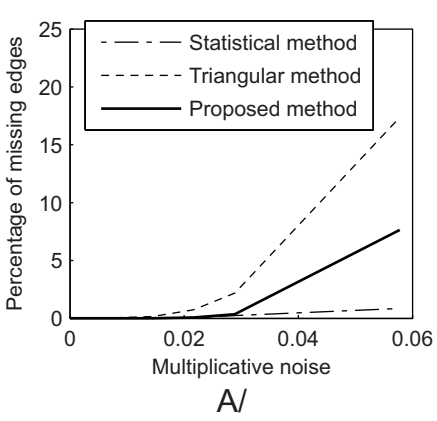
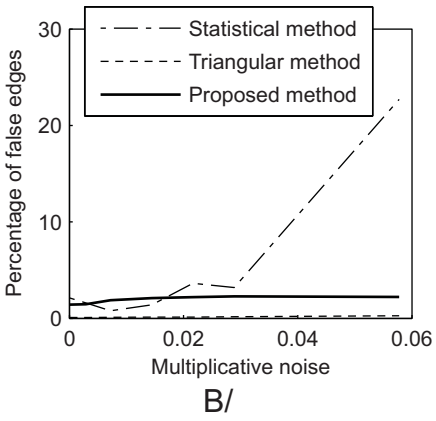


Figure10-GrayLevel

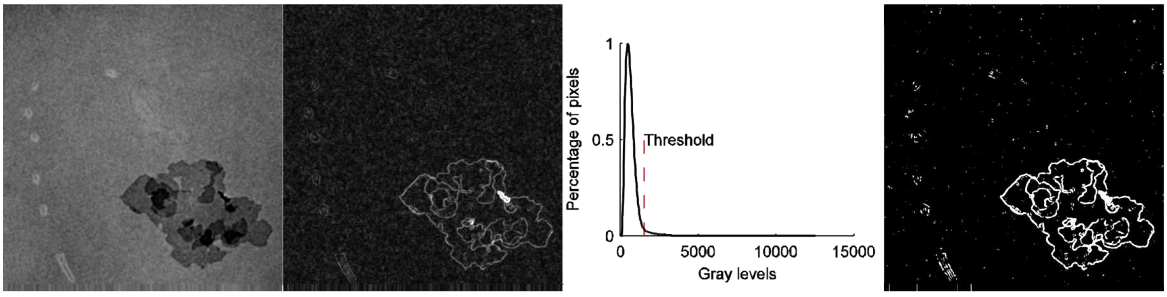


A/



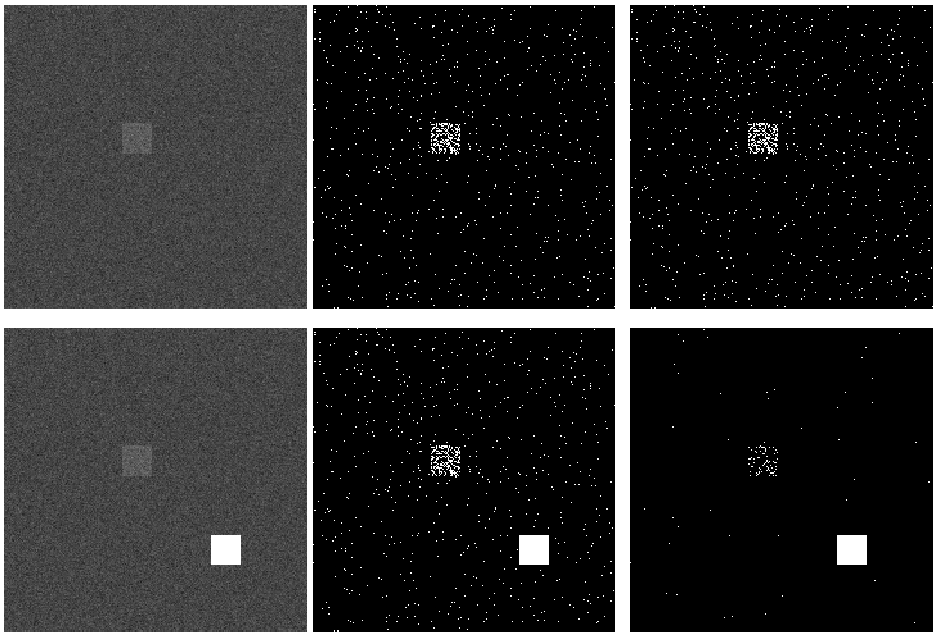
B/

Figure11



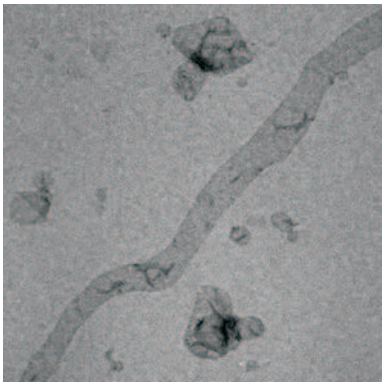
A/ B/ C/ D/

Figure12

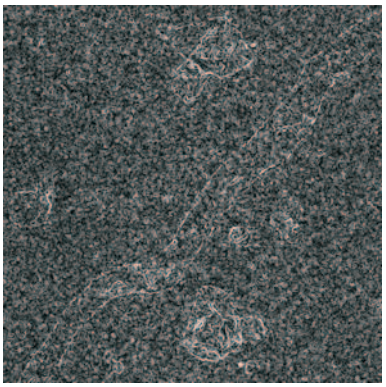


Initial image *T*-point method Triangular method

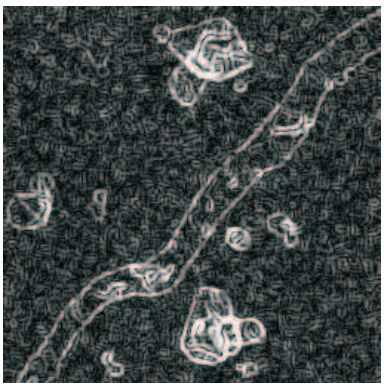
Figure13



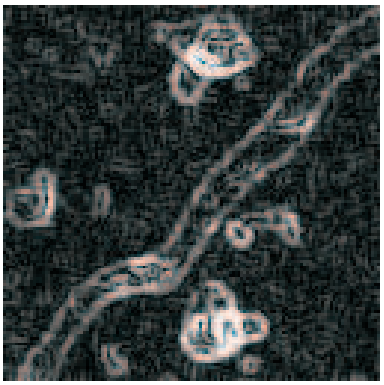
A/ Initial image.



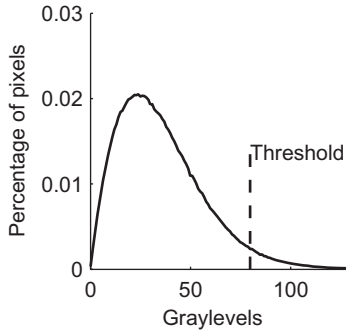
B. Gradient at scale 1 of image A. (1024x1024 pixels).



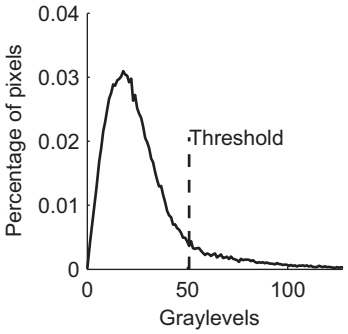
C. Gradient at scale 4 of image A. (256x256 pixels)



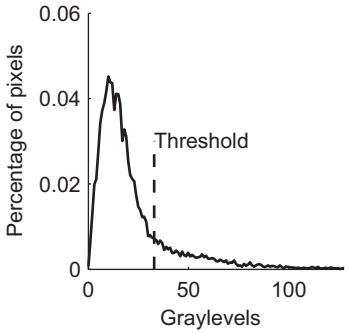
D. Gradient at scale 8 of image A. (128x128 pixels).



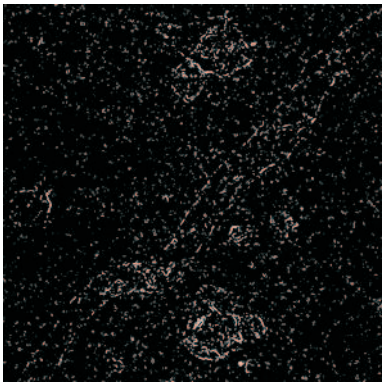
E. Histogram of image B.



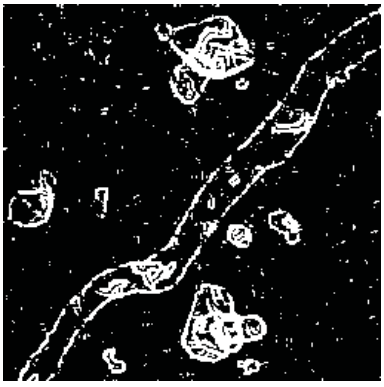
F. Histogram of image C.



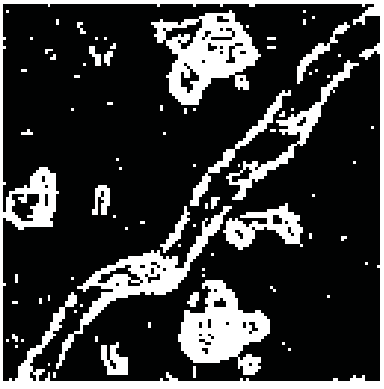
G. Histogram of image D.



H. Image B binarized.



I. Image C binarized.



J. Image D binarized.

Table 1 Computationally convenient algorithm for the search of the optimal breakpoint.

T-point Algorithm: search for the optimal breakpoint
(1) variables n, S_x, S_{xx}, S_y, S_{yy} and S_{xy} initialized at 0 for k = M to L n = n + 1 computation of S_x, S_{xx}, S_y, S_{yy} and S_{xy} with g_k and h_k , using (25) compute $\varepsilon_1(k)$ using (26) end for (2) variables n, S_x, S_{xx}, S_y, S_{yy} and S_{xy} initialized at 0; $\varepsilon_2(L) = 0$ for k = L to M+1 (step -1) n = n + 1 computation S_x, S_{xx}, S_y, S_{yy} and S_{xy} with g_k and h_k , using (25) compute $\varepsilon_2(k-1)$ using (26) end for (3) look for the index k that minimizes the sum $\varepsilon(k) = \varepsilon_1(k) + \varepsilon_2(k)$