



A new model of image segmentation with multi-threshold

Cai Bo^{1,2*}, Zhigui Liu^{1,2} and Junbo Wang^{1,2}

¹Southwest University of Science & Technology, Mianyang Sichuan, China

²China Academy of Engineering Physics, Mianyang, Sichuan, China

ABSTRACT

Image segmentation is a fundamental and challenging problem in image processing and often a vital step for high level analysis. The aim of image segmentation is to divide an image into different categories based on features, such as intensity, color, histogram or context, where each pixel in the image should belong to one class and only one class. For the image segmentation by the histogram thresholds, several methods have been proposed. However, image segmentation can be two-phase (two categories) or multiphase (more than two categories), the number of categories becomes an important problem in this kind of segmentation. And the segmented results according to the thresholds whether or not consistent to the image is also a problem should be considered. In this paper, we use the bi-level threshold way to divide the image's histogram step by step until the total variance between the histogram and the fitting curve match the stop conditions. After these thresholds have been calculated, we use the structure and contour characterization of the image to deal with the rough result of the segmentation. Experimental results and a comparative study with the other efficient and known multilevel threshold methods over synthetic and real images have shown that the proposed method is consistent to the image contents especially in nature images.

Key words: Image segmentation; multilevel thresholds; Image structure; Image contour.

INTRODUCTION

Image segmentation has been the subject of intensive research and a wide variety of segmentation techniques has been reported in recent decades. In general, image segmentation divides an image into related sections or regions, consisting of image pixels having related data feature or structure characterization values. It is an essential issue since it is the first step for image understanding and any other, such as feature extraction, recognition and matching[1-5], heavily depends on its results. Segmentation algorithms are based on two significant criteria: the homogeneity of a region (threshold) and the discontinuity between adjacent disjoint regions (contour of image). Since the segmented image obtained from the homogeneity criterion has the advantage of smaller storage space, 1-dimension problems, fast processing speed and easy in manipulation, threshold techniques are considered the most popular.

Among the segmentation approaches, the techniques for separating objects from background image or discriminating objects from image that have distinct gray-levels has led to the development of new efficient methods for segmenting different types of images. The pixels belonging to the same object have gray levels within a specific range defined by two or several thresholds. To find the thresholds which are generally located at the valleys of the gray-level histogram, one may use parametric or non-parametric approaches. In parametric approaches, the gray-level distribution of each class has a probability density function that is assumed to obey to a given distribution obtained by estimating its parameters, so that it better matches up the given histogram data. In most cases, this distribution is taken as Gaussian[6-8]. In the case of non-parametric approaches, the thresholds are obtained by optimizing an objective function, such as Otsu's[9] function or the Kapur's[10] function, according to some criteria, without estimating the parameters of the two distributions. Some authors utilized the two approaches in their

thresholding method. In this method, the histogram is approximated by a bi-level function for dividing an image into two regions, and then, the thresholds are determined by minimizing the sum of square errors or the variance of the two regions. Progressively, the thresholding operation has evolved from the bi-level thresholding to the multilevel stage. In bi-level thresholding, the histogram of the image is usually assumed to have one valley between two peaks, which correspond to the background and the objects of this image. If the advantage of bi-level thresholding methods is to give satisfactory results in cases where it is known a priori that the image contains only two principal gray levels, the main drawback is that they are very computationally time consuming when extended to multilevel thresholding, since they search thresholds for optimizing objective functions. However, in the extension of these methods to the multilevel case, some problems have appeared. The first one is the computation time which is still relatively high depending on the complexity of the image to be processed. The second problem is how to determine the threshold number, corresponding to the number of the regions constituting the image. Finally, the transitional boundaries of the image regions can't be simply determined by one or more thresholds.

To overcome the first problem, several fast techniques have been proposed. Some of them are designed especially for computation acceleration of a specific objective function [11-13], while others are designed to be used with a general purpose. Sequential dichotomization [14-16] use an iterative scheme and meta-heuristic optimization methods [17, 18] to realize the quickly compute of threshold. Furthermore, in another approach, in order to accelerate the calculation of thresholds research, the resolution of the histogram is first reduced using the wavelets transform. After that, the optimal thresholds are determined faster by optimizing the objective function based on an exhaustive search [19] or through meta-heuristics. For the same purpose, Chang and Wang [20] use a low-pass and high-pass filter repeatedly for adjusting the peaks or valleys to a desired number of classes. The valleys in the filtered histogram are then considered as threshold values. In [21], the authors propose the histogram smoothing by convoluting it with a Gaussian filter for extracting the peaks and the valleys. In [22], Arifin and Asano use hierarchical clustering method to solve the multilevel thresholding problem. Each non-empty gray level of the histogram is, initially, considered as a separate mode representing a cluster. Then, the similarities between adjacent clusters are computed and the most similar pair is merged. The estimated thresholds which are defined as the highest gray levels of the clusters are obtained by iterating this operation until the desired number of clusters is found. In [23], Chen and Wang use the support vector regression as a fitting tool rather than other histogram smoothing techniques and the support vectors on boundary (BSV) are determined to provide candidate thresholds. The optimal threshold values are then obtained by seeking among the BSV set where negative to positive transition of the first-order derivatives of the fitted histogram is occurring. It can be pointed out that the Otsu's function is modified by incorporating a weight in order to ensure that the threshold values will always stand in the valleys of the histogram [24, 25].

The Gaussian functions curve fitting and other clustering solutions, as mentioned above, want to find the correct clusters which accordance to the histogram. Although the Gaussian mixture model has the inconveniences of having no systematic or analytic solution, slow convergence and high computational cost [26], it represents a flexible method of statistical modelling with a wide variety of scientific applications [27]. In general, GM involves the model selection, i.e., to determine the number of components in the mixture (also called model order), and the estimation of the parameters of each component in the mixture that better adjust the statistical model. Gaussian mixture model is considered a difficult optimization task as above mentioned. As an optimization problem, the presented here requires an objective function, which makes use of Hellinger distance to compare the GM candidate and the original histogram. This distance measure works with probability density functions, making it appropriate to the problem presented in this work, and was shown that this distance is the most suitable to construct a minimum distance estimator [28-30].

For the purpose of calculating the thresholds, previous studies performed to assess the performance of DE, PSO, and ABC algorithms included the work in [31] showing that ABC performs better than PSO, and DE on a suite of classical benchmark functions. It was shown that the performance of ABC is better or at least similar than DE and PSO while having a smaller number of parameters to tune. In [32, 33] several DE variants were empirically compared over a benchmark of 13 functions, finding that the version *best / 1 / bin* has the best behavior regardless quality and robustness. A study comparing variations of PSO over power systems is made, finding that the enhanced general passive congregation PSO shown the best performance, but also has a high computational cost. The aforementioned studies suffer from one limitation: the comparisons are based on a set of synthetic functions with exact and well-known solutions and none of them were applied to image processing. The proposed study overcomes such drawbacks by assessing the performance of the set of evolutionary algorithms when they are applied to the image processing problem of segmentation, particularly multi-threshold segmentation, where an exact solution does not exist.

Few papers have been published in the literature for solving the problems of the threshold number and the image structure. In most cases, they focus on the optimal thresholds number, which optimizes a cost function, and then determined. Y. Zou *et al.* [8] use the bidirectional restriction methods automatically calculate the lower bound f_l and the upper bound f_h of the gray level distribution of the transition region. The pixels with gray level in the interval $[f_l, f_h]$ are treated as the transition region. The effective average gradient (EAG) approach [34] is a representative among the bidirectional restriction methods. The EAG method uses the clip transformation followed by computing the effective average gradient to obtain the lower bound f_l and the upper bound f_h . The EAG method is fully automatic, and no limitations in shape and size of objects are imposed. This method, however, faces two primary limitations: (1) for the relatively complex images, the gray level interval $[f_l, f_h]$ is often too loose, which will misclassify many object or background pixels into the transition region; (2) Groenewald *et al.* [35] proved the existence of $f_l > f_h$. In this case, the transition region cannot be extracted, and this has been experimented in [8], where $f_l > f_h$ occurs 3 times among all 50 real images.

Generally, the threshold segmentation of image is mainly focus on the calculation of the thresholds according to the histogram of the processing image and then using the thresholds to divide the image gray into some categories. The authors of [36, 37] have shown the concept of topological derivative of image. Inspired by this concept, [25] we propose a fast and efficient method in finding the threshold number and values. And introduce a new multilevel thresholding method using a multiphase level set technique for segmenting complex images with reduced computation time whatever the number of regions constituting these images. This proposed method consists in applying the multiphase level set technique on the histogram and not on the image. Considering of the different phase of the histogram and the image regions, we assume that the different region of the image has different phase in the histogram, and segment the image according to the consistence between these two characters.

The remainder of this work is organized as follows: In section 2, we present the method following Gaussian approximation of the histogram. For the purpose of calculating the exact threshold number, we've been given the analysis and comparison of the performance of different algorithms in section 3. In section 4, based on the analysis of the section 2 and 3, we modeling the process of image segmentation by reducing the complexity of the histogram and the image characterization. In section 5, as an example of the proposed framework, we show the features of this model comparing to the other thresholding segmentation algorithms. After the threshold number and threshold values have been detected according to the proposed model, we give the analyzing of the regions and edges in section 6. The experimental results and discussion of this model are presented in section 7. Finally, the main conclusions are drawn in section 8.

2 The Gaussian approximation of the image histogram

In what follows, histogram $h(g)$ represents a gray level distribution of an image with L gray levels $[0, 1, \dots, L-1]$; it is also summed that $h(g)$ is normalized, considered as a probability distribution function:

$$h(g) = \frac{n_g}{N}, \quad h(g) \geq 0, \quad (1)$$

$$N = \sum_{g=0}^{L-1} n_g, \quad \text{and} \quad \sum_{g=0}^{L-1} h(g) = 1,$$

Where n_g denotes the number of pixels with gray level g , whereas N represents the total number of pixels contained in the image. The mix of Gaussian probability functions:

$$p(x) = \sum_{i=1}^K P_i \cdot p_i(x) = \sum_{i=1}^K \frac{P_i}{\sqrt{2\pi\sigma_i}} \exp\left[-\frac{(x-u_i)^2}{2\sigma_i^2}\right] \quad (2)$$

We can approximate the original image histogram, dealing with P_i as the a priori probability of class i , $p_i(x)$ as the probability distribution function of gray-level random variable x in class i , u_i and σ_i as the mean and standard

deviation of the i -th probability distribution function and K as the number of classes contained in the image. In addition, the constraint $\sum_{i=1}^K P_i = 1$ must be made certain.

The Hellinger distance is used to estimate the $3K(P_i, u_i$ and $\sigma_i, i = 1, \dots, K)$ parameters, comparing in such way the mixture of Gaussian functions (or candidate histogram) and the original histogram:

$$E = \sqrt{\sum_{j=1}^n \left[\sqrt{p(x_j)} - \sqrt{h(x_j)} \right]^2} \quad (3)$$

Where $p(x_j)$ is the histogram formed with the candidate Gaussian mixture, and $h(x_j)$ is the experimental histogram that corresponds to the gray level image. Such a formula represents the fitness function used by the three nature inspired algorithms reported in this work and it does not need extra parameters. Fig.1 shows the two Gaussian functions fitting result of the image histogram with Hellinger distance (the entire following curve fitting results are using red channel of the image Lena.jpg.)

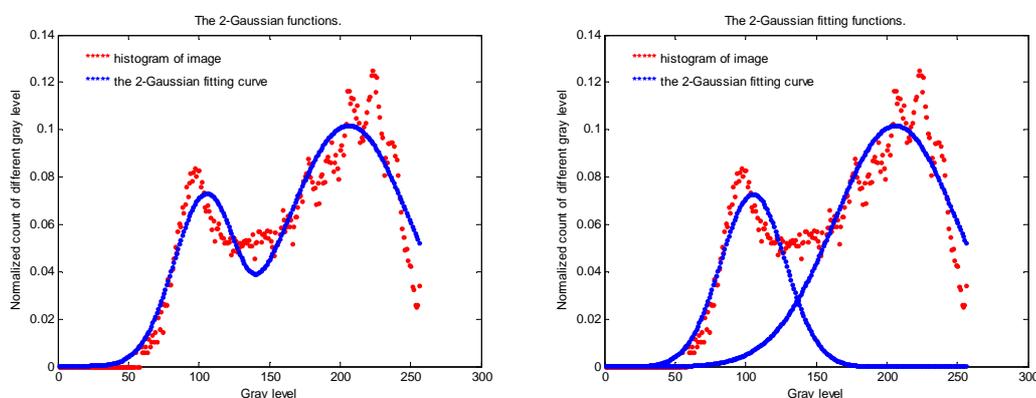


Fig.1 The global searching result of the histogram fitting with 2 Gaussian functions

title('Curve fitting of histogram with 2 Gaussian functions') The global searching results

u(i)=104.074656003080

204.986681817711

P(i)=0.216056706315944

0.783943293684056

Sigma(i)=16.2408028635014

30.5831753655454

ylabel('Normalized count of different gray level sqrt(h(g))')

For the purpose of determining the optimal threshold number and values, several methods have been reported as in

[38]. According to Otsu[9], in order to find two thresholds, the number of possible combinations is $\binom{L}{2}$. The

combination of nature inspired algorithms [39-41] and classic techniques [9], as a good option to deal with computational cost has been proved in the case of image processing problems. The first and the second row of Fig.2 shows the different results of 4-Gaussian functions by using the particle swarm optimization (PSO) and the direct curve fitting algorithms. Seeing from the fitting results, we may find where the more number of Gaussian functions been used in the histogram fitting, the results of different algorithms may causing much inconsistency. In the following section, we'll give the detail analysis of the threshold number choosing algorithm.

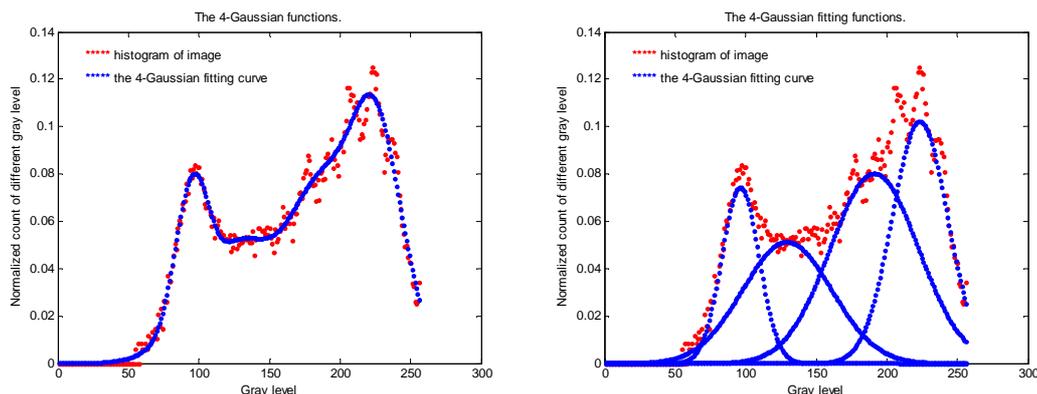


FIG.2 Curve fitting of histogram with 4 Gaussian functions') The global searching results

title('Curve fitting of histogram with 4 Gaussian functions') The global searching results

u(i)=97.4553809762226

174.447200800555

218.164477533569

255.999998854861

P(i)=0.161483103881132

0.407056635859043

0.429992413623962

0.00146784663586275

Sigma(i)=11.1116602966955

34.9198371081020

15.9270299132381

2559.9999977099

xlabel('of gray level 0-255')

ylabel('Normalized count of different gray level sqrt(h(g))')

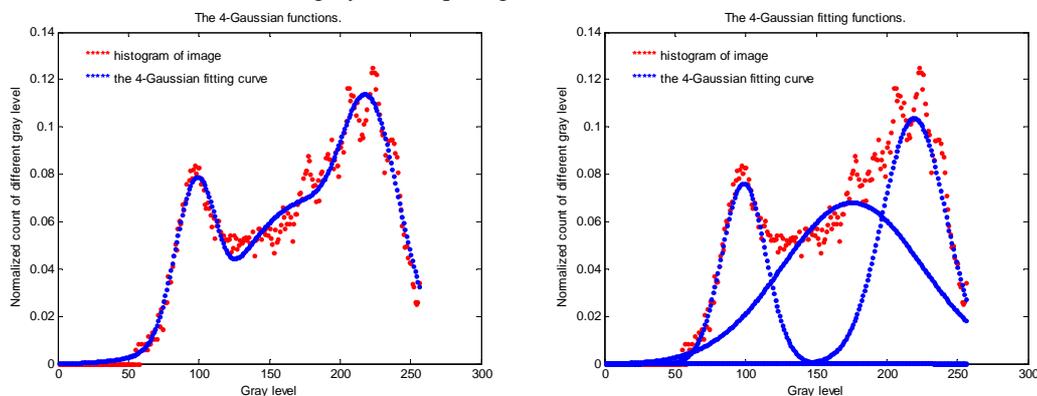


FIG.2 The particle swarm optimization (PSO) and the direct curve fitting results of the image histogram

title('Curve fitting of histogram with 4 Gaussian functions (another type of result)') The global searching results

u(i)=95.0905071156364

128.240776274058

189.830787688705

222.210231291886

P(i)=0.147565147224805

0.128530943014258

0.354890777576301

0.369013132184636

Sigma(i)=9.29506807161366

22.3433258885996

21.9703186747988

13.8904939163273

xlabel('of gray level 0-255')

ylabel('Normalized count of different gray level sqrt(h(g))').

3 The problem of Gaussian function number

3.1 The fitting processing

How many Gaussian functions or parameters should be chosen for the optimize algorithm is always the main problem need to be solved. Unfortunately, many thresholding algorithms are not able to automatically determine the required number of thresholds, as has been noted by Whatmough[42]. [25] using a two-step multiphase level set method to solve this problem: The first step, the complex histogram is approximated by a weighted sum of Heaviside functions using the Chan-Vese segmentation model in order to highlight the valleys of the histogram; In the second step, the number and the values of the thresholds are extracted by a simple search of the minima of these valleys.

Furthermore, in another approach, in order to accelerate the calculation of thresholds research, the resolution of the histogram is first reduced using the wavelets transform. After that, the optimal thresholds are determined faster by optimizing the objective function based on an exhaustive search [19] or through meta-heuristics. Chang [20] uses a lowpass/highpass filter repeatedly to adjust (decrease/increase) the number of peaks or valleys to a desired number of classes and then the valleys in the filtered histogram are used as thresholds. Inspired by the low-pass and high-pass filter algorithm, we use the two Gaussian functions and Hellinger distance to fit the histogram of the image step by step like a binary searching processing.

According to the above analysis, the proposed multilevel thresholding method proceeds in two steps. The first step concerns the histogram approximation and the second, the thresholds determination that defines the limits of the image gray level classes. In our model, the only parameter to introduce is the stop condition \mathcal{E} , the total fitting variance. The lower the parameter, the better the approximated histogram matches the original one. This means that the number of detected valleys is large, since the non-significant valleys can also be detected. If the value of \mathcal{E} is too large, some valleys of the histogram, therefore some thresholds, may not be detected. Therefore, the adjustment of threshold number with in a large range where the number of detected thresholds remains constant and different from one can considered as a good procedure for optimizing the number of thresholds, when nothing is known a priori about the distribution of the gray levels. In the proposed algorithm, we let the iteration processing the large range and the large independent variance thresholds firstly.

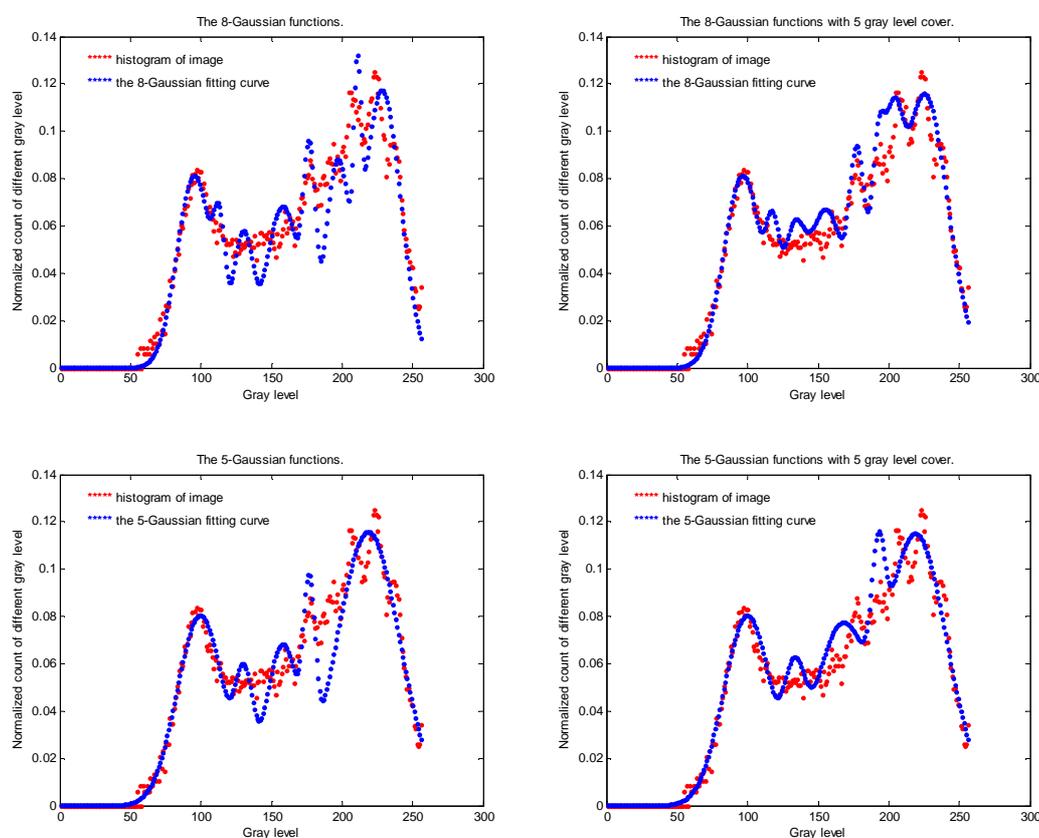


FIG.3 The fitting results of regions without and partially covering, the first row is the 8-Gaussian functions fitting result; the second row is the 5-Gaussian functions fitting result (Lena.jpg).

Simply dividing the gray levels of the image histogram into some independent regions, the curve of the fitting results may change dramatically on the up and down boundary of the fitted region. For the purpose of let the fitting curve more smoothly, we let the regions of gray scales partially covered with each other as:

$$\text{Region}_i = [T_i - a, T_{i+1} + a] \quad (4)$$

Where T_i and T_{i+1} is the i -th and $(i+1)$ -th threshold. In our experiments, we choose the parameter $a = 5$. Fig.3 shows the results of 5 and 8-Gaussian functions fitting results of regions without and partially covering. Except the value of the total variance of fitting result, we consider the changing of fitting variance σ_i , $i = 1, 2, \dots, p$ at the same time. Fig.4 shows the fitting variance of some testing images with different threshold number. According to the changing of the fitting variance, we give another stop condition as $\sigma_{i+1} > \sigma_i$ to avoid over-segmentation.

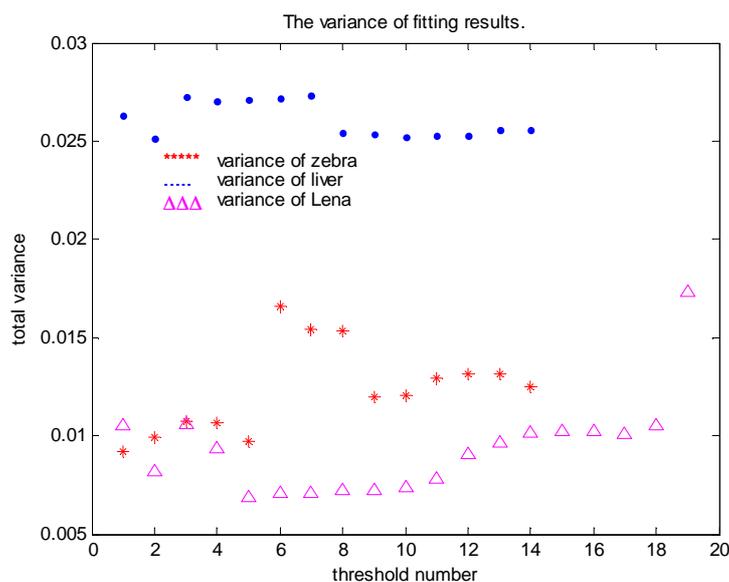


FIG.4 The variance different Gaussian functions' fitting results (Lena, zebra and liver).

3.2 The relation between threshold number and the image gray levels

Considering of the discrimination capacity of human eyes, the dividing of the image gray scales should not be too small. In fact, this ability of human is variable according to the image contents. When the contrast of the image is higher, the vision discrimination ability may become weak, on contrary, when the contrast is lower, the ability may become strong in discriminating the weak edges. The authors of [43-46] had given the explanation of this kinds of visual characters. In this process, we divide the gray scales into 32 stages, when the grayscale of the input image is 0-255, then we choose the smallest difference between thresholds as $Tr = 8$ gray scales, so the biggest number of thresholds is 31. On this condition, much of the papers aimed at the adjustment of the histogram according to the local and global characters of the input image. The adaptive histogram equalization (AHE) [46] equalize an input image I with quantized gray levels (GL) scaled between $-1/2$ and $1/2$. By using the Kronecker delta function $\delta(i, j)$, which equals 1 if $i = j$ and 0 otherwise, they sift the pixels in the image with GL g . But, according to our experiments, all of the equalization algorithms may cause the histogram change to some extent, and let the distribution of the histogram more difficult to fitting. Fig.5 shows the comparison of rough image and the equalized results.

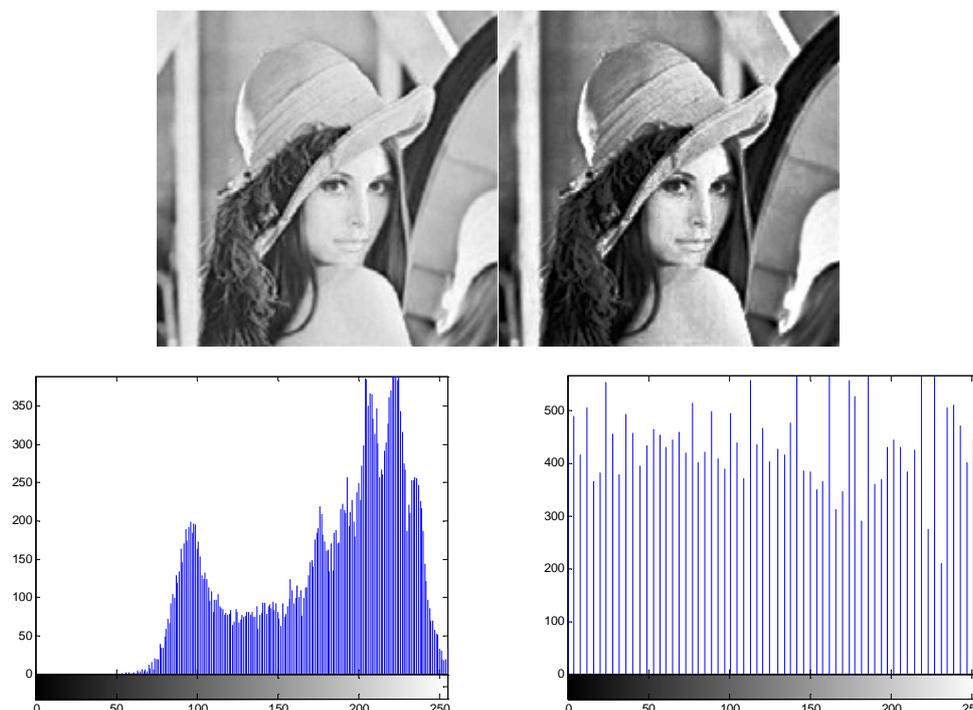


FIG.5 The comparison of rough image and the equalized result

For the purpose of letting our thresholds keep consistency with the visual discrimination capacity, and avoiding the influence of noise distribution, we use 95 percent of the total pixel number's distribution as the contrast of the image. Assume $M \times N$ is the input image size, $h(i)$, $i = 0 \dots 255$ is the histogram of each gray level, the maximum t_h is the chosen threshold of the input image histogram's main distribution. The calculation of this distribution is as follows.

$$\max \{t_h = h(j)\}, \quad \left(\sum_{h(i) \geq t_h} h(i) \right) \geq 0.95 \times M \times N, \quad i, j = 0 \dots 255 \quad (5)$$

After getting the maximum t_h , we set the minimum range Tr of gray level as

$$Tr = \left\lfloor \sum_{i=0}^{255} \frac{h_r(i)}{32} \right\rfloor, \quad i = 0 \dots 255 \quad (6)$$

$$h_r(i) = \begin{cases} 0, & \text{if } h(i) < t_h \\ 1, & \text{otherwise} \end{cases}$$

According to the above analysis, we use the $\sigma_{i+1} > \sigma_i$, $\sigma_i < \varepsilon$ and $threshold_{i+1} - threshold_i < Tr$ as the stop condition to get the threshold number.

4The proposed model

Based on the analysis of part 2 and 3, we modeling the multi-threshold searching algorithm as:

(1) Initializing the convergence condition \mathcal{E} , calculating the image's histogram and normalizing it as equation (1), calculating the stop condition Tr as equation (5-6).

(2) Normalizing the chosen region of the histogram as the following equation;

$$h_i(k) = \frac{n_k}{N_i}, \quad N_i = \sum_{k=l_i}^{u_i} n_k, \quad \sum_{k=l_i}^{u_i} h_i(k) = 1 \quad (7)$$

Where i the i -th gray level range is need to be fitted, h_i is the normalized histogram of this range, $[l_i - a, u_i + a]$ is the lower and upper bounds of this range in gray scale. In our experimentations, we choose the parameter a as 5.

(3) Using the Hellinger distance and two Gaussian functions to fit the i -th normalized region of the histogram;

(4) Choosing the threshold of this fitting results and calculating the variance σ_i , departing the rough histogram into up and low parts;

(5) Choosing the biggest part repeating (2)-(4) until the following condition matched:

$$\begin{aligned} & \sigma_{i+1} > \sigma_i \\ & \text{or } \sigma_i > \varepsilon \\ & \text{or } \text{threshold}_{i+1} - \text{threshold}_i < Tr \end{aligned} \quad (8)$$

Where $\text{threshold}_{i+1} - \text{threshold}_i < Tr$ is the minimum region between two thresholds?

(6) When all the region matching the stop conditions, we can get the number and value of the thresholds. Fig.6 shows the basic processing of our proposed algorithm.

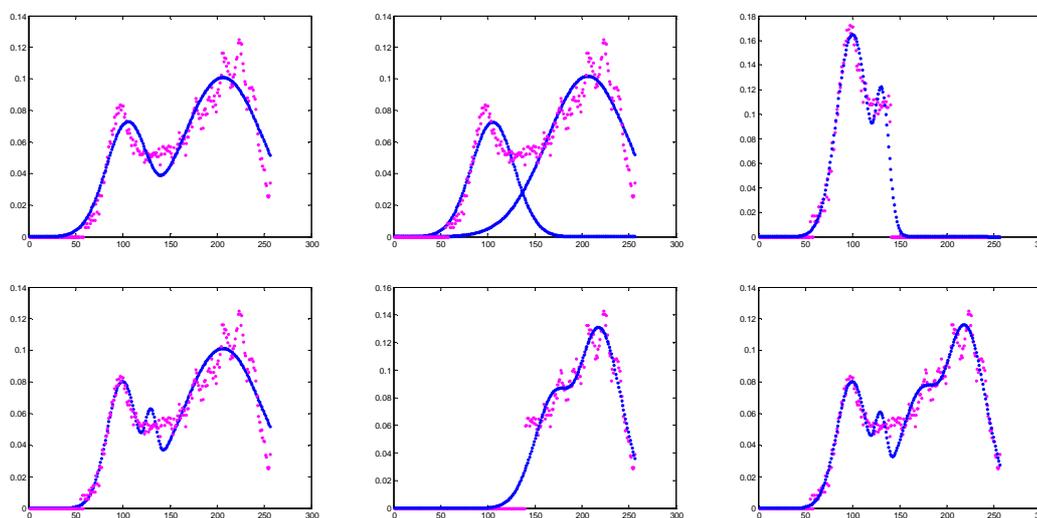


FIG.6 The processing of the curve fitting.

5 Thresholds searching

Multilevel thresholding is a process that segments a gray-level image into several distinct regions. This technique determines more than one threshold for the given image and segments the image into certain brightness regions, which correspond to one background and several objects. The method works well for objects with colored or complex backgrounds, on which bi-level thresholding fails to produce satisfactory results. Redid et al. [47] proposed an iterative form of Otsu's method, so as to generalize it to multilevel thresholding. Ridler and Calvard algorithm [48] uses an iterative clustering approach. An initial estimate of the threshold is made (e.g., mean image intensity), pixels above and below are assigned to the white and black classes respectively.

Compare to the abovementioned algorithms, V. n. Osuna-Enciso, et al. [30] consider that the data classes are organized such that $\mu_1 < \mu_2 < \dots < \mu_K$, the threshold values can thus be calculated by estimating the overall probability error for two adjacent Gaussian functions, as follows:

$$\begin{aligned} E(T_i) &= P_{i+1} \cdot E_1(T_i) + P_i \cdot E_2(T_i), \\ i &= 1, 2, \dots, K-1 \end{aligned} \quad (9)$$

$$E_1(T_i) = \int_{-\infty}^{T_i} p_{i+1}(x) dx, \quad (10)$$

And

$$E_2(T_i) = \int_{T_i}^{\infty} p_i(x) dx \quad (11)$$

$E_1(T_i)$ is the probability of mistakenly classifying the pixels in the $(i+1)$ -th class to the i -th class, while $E_2(T_i)$ is the probability of erroneously classifying the pixels in the i -th class to the $(i+1)$ -th class; as was abovementioned, formulae (9), (10) and (11) states for two consecutive Gaussian functions in order to get one threshold point and the process is repeated for each pair until get all the threshold points. P_j is the priori probabilities within the combined probability density function, and T_i is the threshold value between the i -th and the $(i+1)$ -th classes. One T_i value is chosen such as the error $E(T_i)$ is minimized. By differentiating $E(T_i)$ with respect to T_i and equating the result to zero, it is possible to use the following equation to define the optimum threshold value T_i :

$$AT_i^2 + BT_i + C = 0 \quad (12)$$

Considering

$$A = \sigma_i^2 - \sigma_{i+1}^2$$

$$B = 2 \cdot (\mu_i \sigma_{i+1}^2 - \mu_{i+1} \sigma_i^2) \quad (13)$$

$$C = (\sigma_i \mu_{i+1})^2 - (\sigma_{i+1} \mu_i)^2 + 2 \cdot (\sigma_i \sigma_{i+1})^2 \cdot \ln \left(\frac{\sigma_{i+1} P_i}{\sigma_i P_{i+1}} \right)$$

When the above quadratic equation has two possible solutions, they choose the positive one which falls within the interval. Fig.7 shows the calculation of each threshold, when the Gaussian function number is 4. Seeing from Fig.7, intuitively, we may found using the algorithm of [30], whatever the threshold T_i between u_i , and u_{i+1} is, it seems always inconformity. Because some of the thresholds are located on the valleys while others are located on the steep slopes of the histogram.

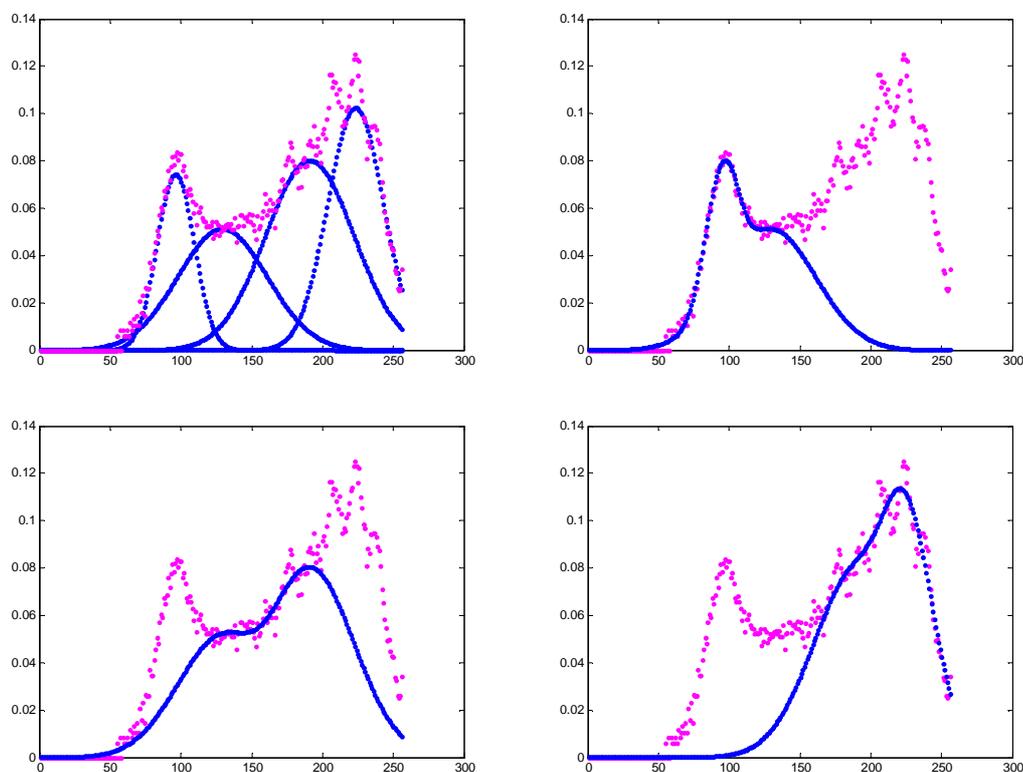


Fig.7 The 4-Gaussian functions fitting result and the adjacent 2-Gaussian fitting results.

Our main propose is to segment the image, and hopefully the thresholds calculated from the Gaussian fitting functions may be used to segment the image accurately. Unfortunately, the image regions often are transitional and these edges can't be segmented simply by one or several thresholds. Seeing from Fig.8(a), the fitting result of the input image histogram, we may easily found there are three kinds of thresholds, the mean of peak-to-peak, the valley of the fitted curve, and the cross point of the two Gaussian functions (the result has been shown in Fig.8 (b), (c), and (d)).About this problem, many researchers have been given discussions in detail [19,20,22,39,41].

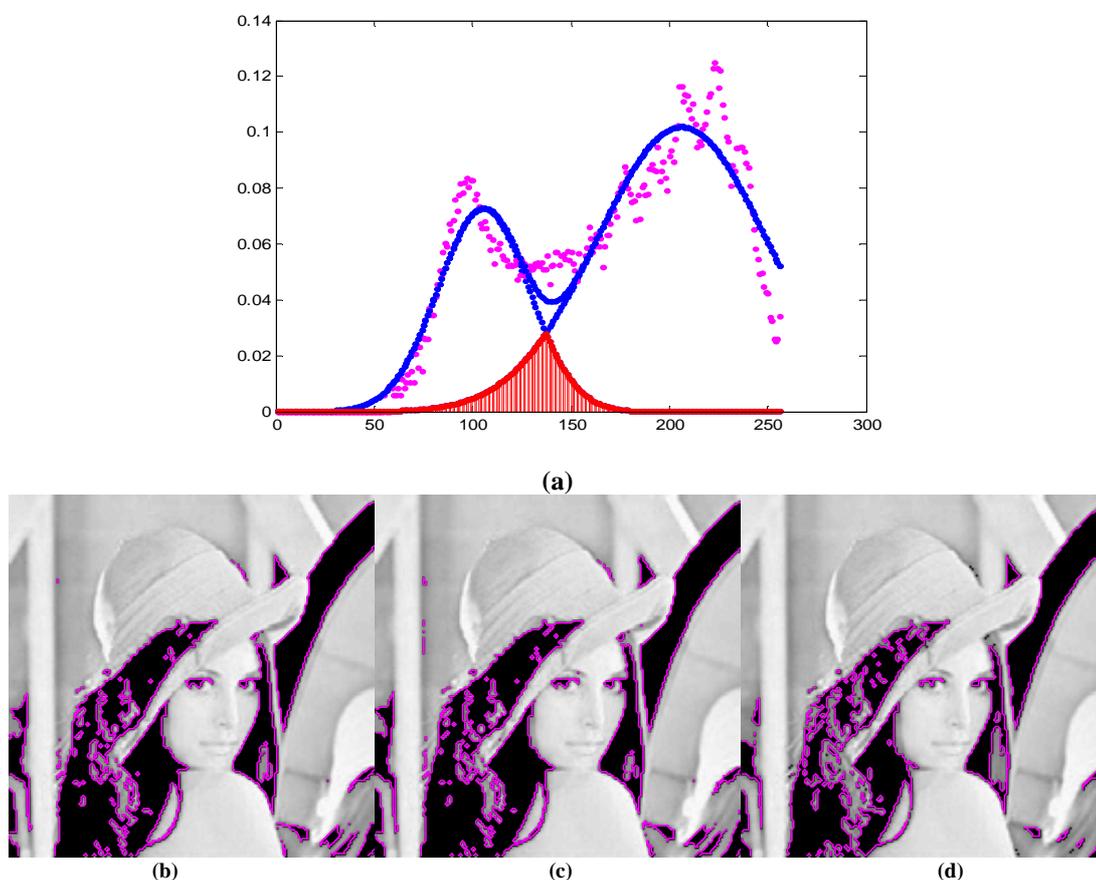


Fig.8 The 2-Gaussian curve fitting and segment result of different thresholds; (a) is the fitting result and the two Gaussian distribution; (b) is the segment result using the cross point; (c) is the segment result using the valley as threshold; (d) is the segment result by taking the threshold as the mean of peak-to-peak.

The thresholds search constitutes the second step of our algorithm. The proposed method consists in finding the number and the values of thresholds by considering the approximated histogram under its form. When our search processing stop with the condition of equation (8), all the last fitted Gaussian functions are our searching results. For the purpose of searching the thresholds, we use the algorithm of [19]. In this paper, we use the derivative of the approximated histogram function $\tilde{V}(i)$ with respect to the gray-level i . This derivative is as follows:

$$\tilde{V}'(i) = \tilde{V}(i+1) - \tilde{V}(i) \quad (14)$$

The graph of $\tilde{V}'(i)$ is a sequence of Dirac peaks of various positive and negative amplitudes as it is depicted in Fig.9, showing the derivative of the approximated histogram Fig.3c. Every negative peak followed by a positive peak delimits a valley. Each detected valley contains a threshold. The value of this threshold is taken as equal to the gray level value of the minimum of the original histogram included in this valley.

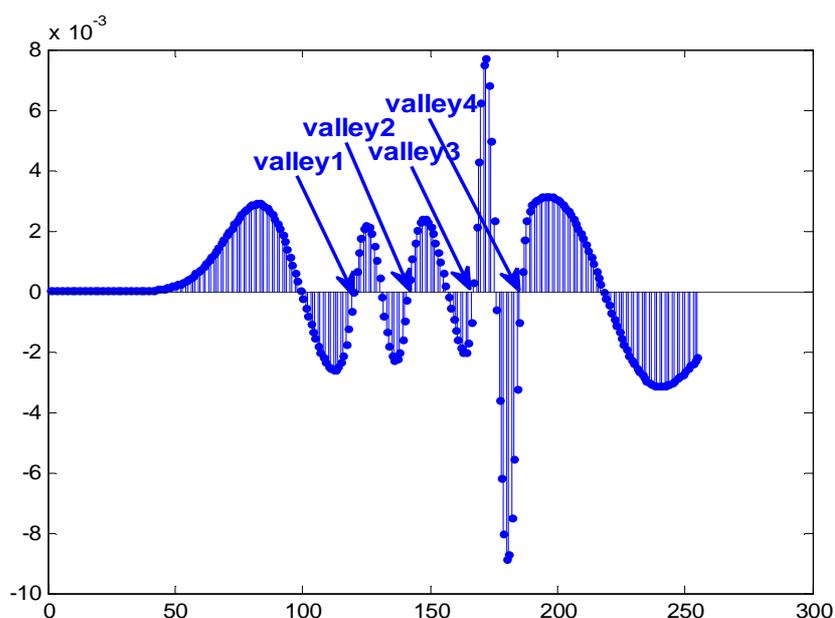


FIG.9 The derivative of the 5-Gaussian functions fitting result

6 The analyzing of the regions and edges

Comparing to the fitting result of the image histogram, the effective segment results of the image is more important. Many methods have been proposed to determine threshold automatically over the past decades. However, applying new ideas and concepts to image thresholding remains an interesting and challenging research area. Unlike all thresholding methods mentioned above, this paper presents an image thresholding approach inspired by the active contour approach [49-51]. For a gray level image, both the thresholding result and the contour detection result are actually a finite point set in the 2-D plane.

Seeing from the LCV/LGIF iteration result (Fig.10), we may easily found that the ACM model is a two-phase segmentation algorithm. Based on Chan-Vese (CV) model [51], the piecewise constant (PC) model for the image with several regions by using a multiphase level set formulation. However, the CV model and the PC model still have some intrinsic limitations. Firstly, if the intensities in internal and external of evolution curves are not homogeneous, it often leads to poor segmentation results due to the wrong movement of the evolving curves. Secondly, the placement of initial contour is still an important issue especially for complicated images. Thirdly, the periodical re-initialization step to get successful segmentation may become time-consuming in the numerical iteration.

Comparing to ACM model, the multi-threshold algorithm is a multi-phase segment algorithm. In the iteration processing, the ACM model use the gradient information to estimate the curves. Inspired by the difference of the two algorithms, in this paper, we'll use the multi-threshold segmentation algorithm to segmentation the image into some phases, and then use the image gradient information to deal with the regions and edges of the natural image.

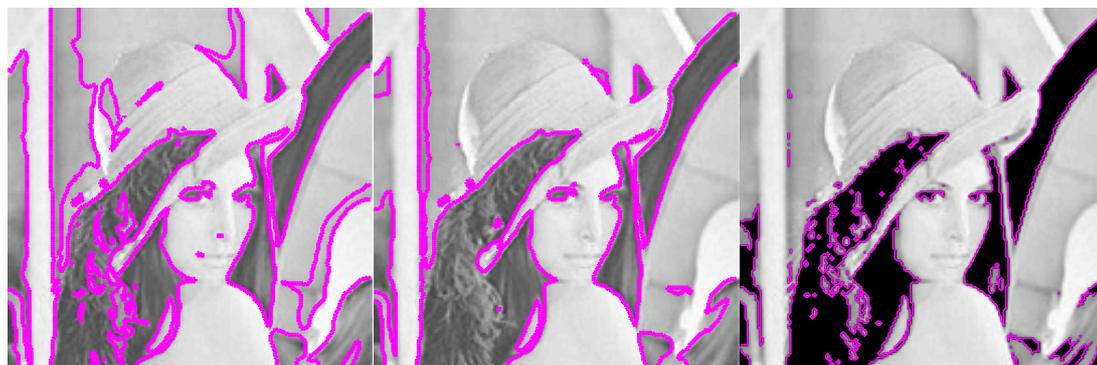


FIG.10 The comparison of the LGIF, LCV and the first threshold segment results (Fig.5.c)

There exists a wide variety of edge detection methods in the literature, using very different strategies for the measurement of intensity changes. Most of the methods are based on the representation of the intensity change by means of a vector, usually called gradient. This observation applies to the methods of Canny, Sobel[52], Feldman and Prewitt. The magnitude of the gradient plays a major role in further processing steps. Given a gray scale image, a gradient is represented by its horizontal G_x and vertical G_y components. Given a gray level image $I_{M \times N}$, we take the magnitude of the gradient as:

$$G(m,n) = \sqrt{G_x(m,n)^2 + G_y(m,n)^2}, \quad n = 1, 2, \dots, N; m = 1, 2, \dots, M \quad (15)$$

[Gx,Gy] = gradient (double (img));
NormGrad = sort (Gx.^2 + Gy.^2);

For the purpose of estimating the stability of the regions after each threshold been achieved, we use the coincidence degree between the edge of threshold segmented region and the gradient as the stability of the segment result. If the stability is higher, the region would be extracted from the image and the histogram, before the next fitting. The calculation of the normalized stability may be simply defined as:

$$Stab_i = \frac{1}{Count(Edge_{\Omega_i})} \sum_{m,n \in \Omega_i} G(m,n) \times Edge_{\Omega_i}(m,n) \quad (16)$$

Where Ω_i is the i -th region of present threshold segmented results; $G(m,n)$ is the magnitude of the gradient; $Count(Edge_{\Omega_i})$ is the pixel number of Ω_i 's edge; the $Edge_{\Omega_i}(m,n)$ is defined as a binary function, when the point (m,n) is on the edge of Ω_i , the value is 1, otherwise 0.

$$Edge_{\Omega_i}(m,n) = \begin{cases} 0, & \text{if } (m,n) \in \text{inner}(\Omega_i) \\ 1, & \text{otherwise} \end{cases} \quad (17)$$

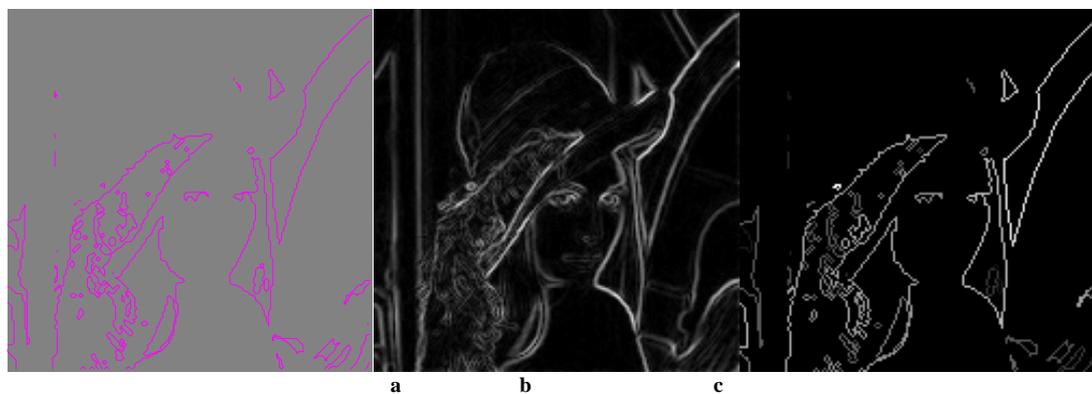


FIG.11 The gradient of the input image;(a) is the first threshold segmented result; (b) is the magnitude of the image; (c) is the stability of each region after the first segmentation; threshold=140.

Seeing from the figure of Fig.11.c, we may easily found the important and unimportant regions of the present segmented result. When all the thresholds have been calculated, the segment results and the regions' stability may also be calculated at last. Fig.12 (b-h) shows the results of different thresholds.

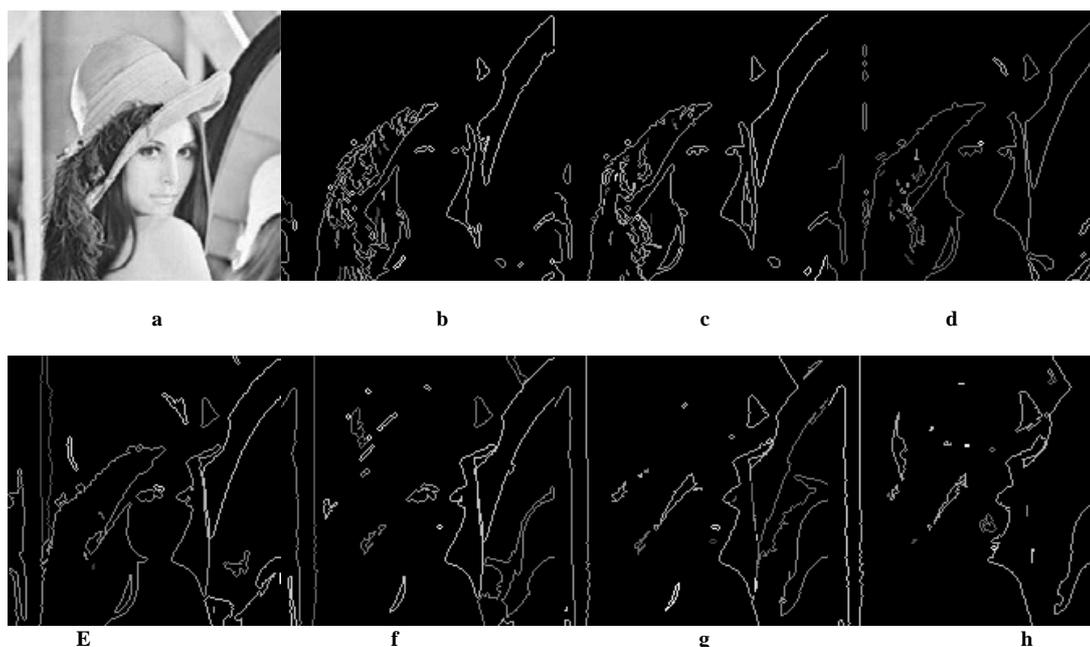


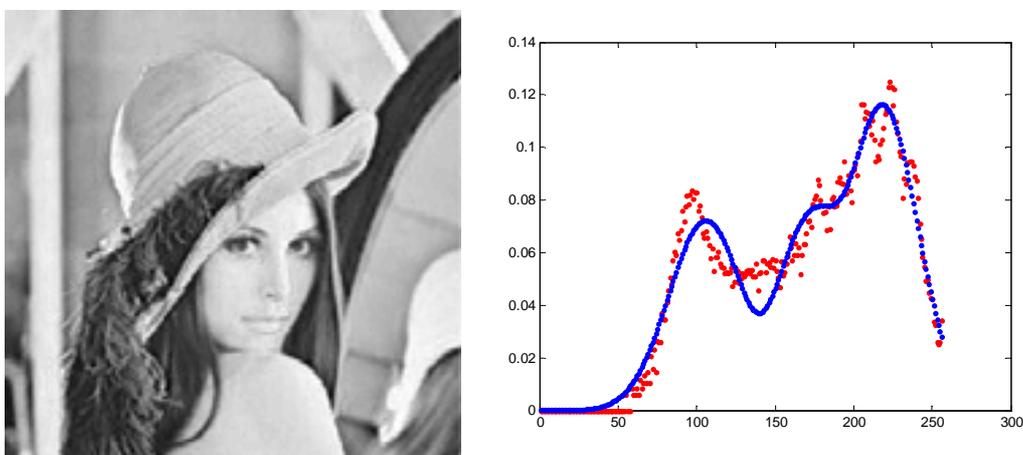
FIG.12 The rough and the segment result of the 7 thresholds, threshold=[108,123,142,167,184,200,214].

7Experimental results

In all experiments, each candidate solution of the segment gray level region holds the elements $[P_1^i, P_2^i, u_1^i, u_2^i, \sigma_1^i, \sigma_2^i]$. For the purpose of avoiding the local minimization solution, we use the mean of the range gray value to roughly dividing the gray level region, and then using the mean and variance of upside and downside as the initial value. Except, we use the directly curve fitting algorithm (DCF), multi-start (MS) algorithm and particle swarm optimal (PSO) algorithm to calculate the results.

To verify the effectiveness in the practical applications, the proposed method is further compared with the direct multi-thresholds segmentation algorithm. Because of the number of the thresholds is different when the input image differ from each other, we set all of the input image's threshold number as 3. It is important remark that all the experiments were performed using a desktop computer with Xeon E5-2643 3.3GHz microprocessor, with 4GB in RAM and programmed in Matlab R2012a.

In the first part of the experiments the main idea was to find the correct separation between two objects and the background in real images, as well as the time spent by each algorithm taking into account the number of function evaluations. In all the multi-thresholds fitting algorithms, the number of functions is very difficult to decide. By using the stop condition as equation (8), we firstly testing the number of Gaussian functions. The result of our testing images has been shown that the number of threshold is consistence to the input images' histogram. Fig.13 shows the fitting result of different images.



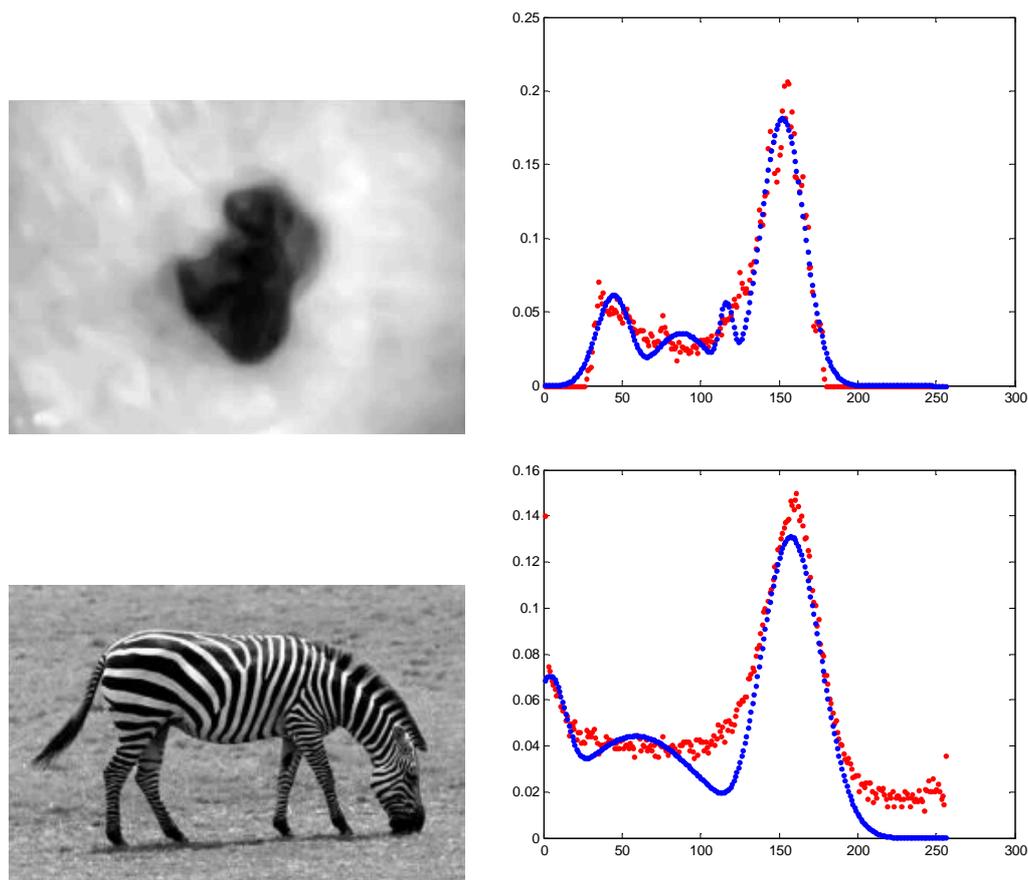
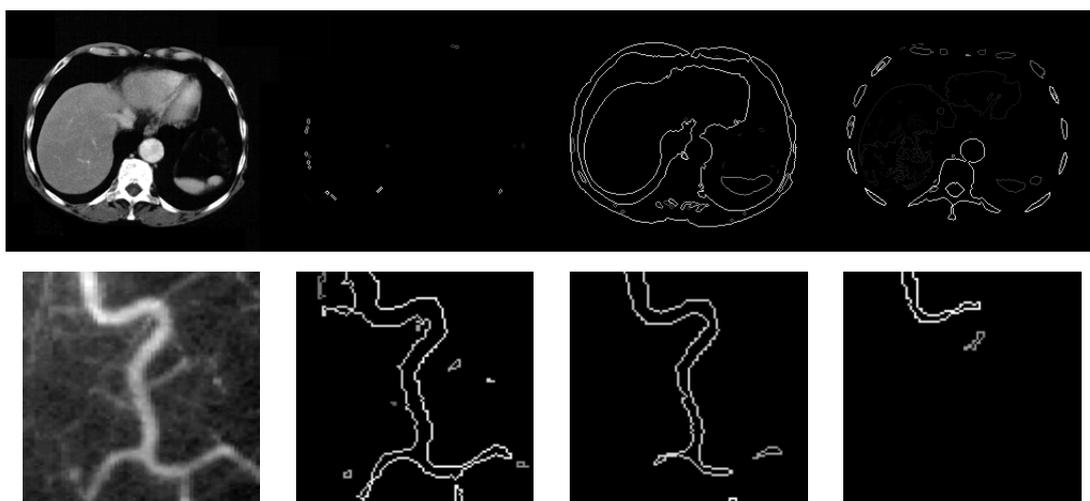


FIG.13 The segment results of Lena, boring and zebra. The thresholds and number of functions: Lena [140,182], 3-Gaussian functions, boring [65,106,124], 4-Gaussian functions, zebra [27,113], 3-Gaussian functions.

For the purpose of comparing the stability, we set the threshold as a constant number 3, and then using the direct multi-threshold curve fitting and the proposed algorithm to get the segment result. The results are shown in Fig.14 and Tab.1. From Tab.1, we may found, the proposed algorithm may easily get the set number of thresholds, but the direct multi-threshold fitting algorithm can't get the thresholds on some of the testing images. Tab.2 shows the detail of the fitting result of these two algorithms with the image Lena.



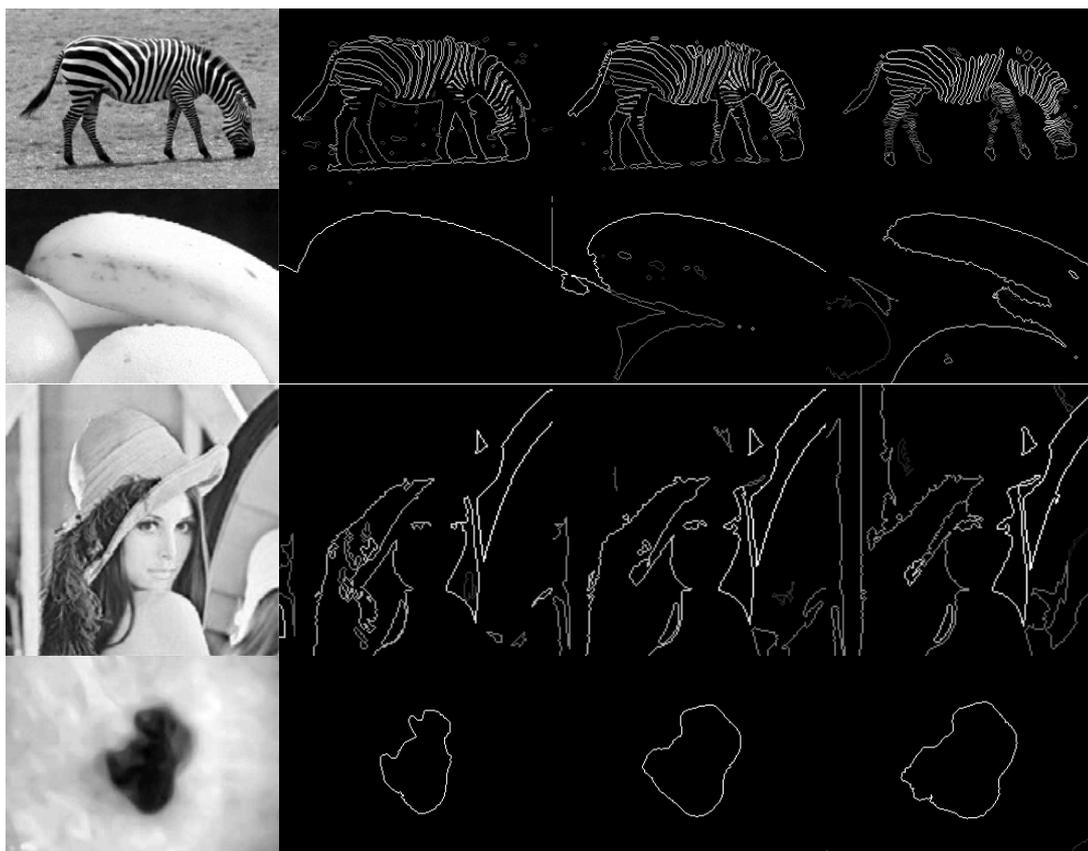


FIG.14the rough and the segment results of the 3 thresholds.The first row is liver image, the second row is vessel image, the third row is zebra image, the fourth row is banana image, the fifth row is Lena image, and the sixth row is boring image.

TAB.1 The comparison of different images' fitting result (multi-start algorithm)

Image	Direct Multi-threshold fitting			The proposed algorithm		
	threshold	u	sigma	threshold	u	sigma
Liver	[26,90]	[2.1, 12.1,73.8,255]	0.0244	[24,101,120]	[2.89,67.4,111.8,122]	0.0272
Boring	[89]	[52.7,129.6,152.1,256]	0.0097	[65,106,124]	[43.5,86.7,115.3,150.9]	0.0111
Vessel	[26,89,248]	[2.2,67.8,122.7,256]	0.0237	[72,81,95]	[59.5,75.2,85.4,96.8]	0.0106
Zebra	[37,108]	[0,0.58,105.3,156.4]	0.0059	[22,77,118]	[2.6,53.5,94.6,156.3]	0.0107
Banana	[64]	[45.9,229.1,250.1,251.8]	0.0116	[66,119.5,214]	[1,66,173,210]	0.0152
Lena	[121,143,206]	[95.1,128.2,189.8,222.2]	0.0051	[137,167,186]	[104.8,156, 175.4,217.5]	0.01059

TAB.2 The comparison of the proposed algorithm and the direct multi-threshold fitting

Image Lena	Direct Multi-threshold fitting			The proposed algorithm		
	DCF	MS	PSO	DCF	MS	PSO
u1	95.098	68.27	3.25	105.01	104.79	104.79
sigma1	9.31	100	256.79	16.91	16.75	16.75
u2	128.29	104.05	8.77	150.26	156.01	163.12
sigma2	22.35	16.21	494.45	6.55	9.34	11.28
u3	189.62	204.98	79.84	171.85	175.389	173.47
sigma3	21.85	30.58	647.44	5.88	3.69	16.97
u4	222.16	215.38	230.95	217.53	217.46	217.46
sigma4	13.93	48.26	663.06	15.71	15.76	15.76
thresholds	[121,143]	140	NA	[136,161,186]	[137,167,186]	[137,167,186]
total variance	0.0051	0.0105	0.0374	0.0108	0.0106	0.0106

8 Conclusion and future work

This paper presents a multi-threshold segmentation algorithm based on two Gaussian functions fitting. It aimed at the minimizing the Hellinger distance between the original and the candidate histogram. Based on the problem of threshold number, we've been given the detail analyzing between the stop condition and the proposed fitting processing. Besides, by using the image contour and the segmentation results, we've been given the calculation of the stability of the image edges. According to the experiments of different images, the results show that the algorithm may easily found the number of multi-threshold comparing to the other multi-threshold algorithms.

Future work includes the comparisons of different image segmentation algorithms and finding the more stable stop conditions to get the number of thresholds. Our aim is segment the image and get the reasonable regions of different objects, so the multi-threshold algorithm should be connected to the other algorithms, and at the same time, the algorithm has the absents of segmenting the transitional edges when the input image has too much contents.

Acknowledgements

The authors thank the anonymous reviewers for their manyvaluable comments and suggestions that helped to improve boththe technical content and the presentation quality of this paper.This research is supported by the National NatureScience Foundation of China (60808020).

REFERENCES

- [1]X. Meng, et al., "Building global image features for scene recognition," *Pattern Recognition*, vol. 45, pp. 373-380, **2012**.
- [2]N. Tsapanos, et al., "Shape matching using a binary search tree structure of weak classifiers," *Pattern Recognition*, vol. 45, pp. 2363-2376, **2012**.
- [3]J. Weber and S. Lefèvre, "Spatial and spectral morphological template matching," *Image and Vision Computing*, **2012**.
- [4]B. Wu, et al., "Integrated point and edge matching on poor textural images constrained by self-adaptive triangulations," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 68, pp. 40-55, **2012**.
- [5]G. Aloupis, et al., "Non-crossing matchings of points with geometric objects," *Computational Geometry*, vol. 46, pp. 78-92, **2013**.
- [6]E. Navon, et al., "Color image segmentation based on adaptive local thresholds," *Image and Vision Computing*, vol. 23, pp. 69-85, **2005**.
- [7]S. L. S. Abdullah, et al., "Segmentation of Natural Images Using an Improved Thresholding-Based Technique," *Procedia Engineering*, vol. 41, pp. 938-944, **2012**.
- [8]Y. Zou, et al., "Image bilevelthresholding based on stable transition region set," *Digital Signal Processing*, **2012**.
- [9]N.Otsu, "A threshold selection method from grey-level histograms," *IEEE Transactions on Systems Manand Cybernetics*, vol. 9, pp. 62-66, **1979**.
- [10]T. Pun, "A new method for grey-level picture thresholding using the entropy of the histogram," *Signal Processing*, vol. 2, pp. 223-237, **1980**.
- [11]P.-S. Liao, et al., "A fast algorithm for multilevel thresholding," *J. Inf. Sci. Eng.*, vol. 17, pp. 713-727, **2001**.
- [12]K. C. Lin, "On the statistical and computational performance of image thresholding and determination of class number," in IECON 02 [Industrial Electronics Society, IEEE 2002 28th Annual Conference of the], **2002**, pp. 2179-2184.
- [13]B. J. Devereux, et al., "An efficient image segmentation algorithm for landscape analysis," *International Journal of Applied Earth Observation and Geoinformation*, vol. 6, pp. 47-61, **2004**.
- [14]J.-C. Yen, et al., "A new criterion for automatic multilevel thresholding," *Image Processing, IEEE Transactions on*, vol. 4, pp. 370-378, **1995**.
- [15]M. Sezgin and R. Taşaltın, "A new dichotomization technique to multilevel thresholding devoted to inspection applications," *Pattern Recognition Letters*, vol. 21, pp. 151-161, **2000**.
- [16]B.-F. Wu, et al., "Recursive algorithms for image segmentation based on a discriminant criterion," *Int. J. Signal Process*, vol. 1, pp. 55-60, **2004**.
- [17]P.-Y. Yin and L.-H.Chen, "A fast iterative scheme for multilevel thresholding methods," *Signal Processing*, vol. 60, pp. 305-313, **1997**.
- [18]K. Hammouche, et al., "A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem," *Engineering Applications of Artificial Intelligence*, vol. 23, pp. 676-688, **2010**.
- [19]S. Han, et al., "Fast image segmentation based on multilevel banded closed-form method," *Pattern Recognition Letters*, vol. 31, pp. 216-225, **2010**.
- [20]C.-C. Chang and L.-L. Wang, "A fast multilevel thresholding method based on lowpass and highpass filtering," *Pattern Recognition Letters*, vol. 18, pp. 1469-1478, **1997**.
- [21]J.-H. Chang, et al., "Multi-modal gray-level histogram modeling and decomposition," *Image and Vision Computing*, vol. 20, pp. 203-216, **2002**.
- [22]A. Z. Arifin and A. Asano, "Image segmentation by histogram thresholding using hierarchical cluster analysis," *Pattern Recognition Letters*, vol. 27, pp. 1515-1521, **2006**.
- [23]S. Chen and M. Wang, "Seeking multi-thresholds directly from support vectors for image segmentation," *Neurocomputing*, vol. 67, pp. 335-344, **2005**.
- [24]H.-F. Ng, "Automatic thresholding for defect detection," *Pattern Recognition Letters*, vol. 27, pp. 1644-1649, **2006**.

- [25]A. Dirami, et al., "Fast multilevel thresholding for image segmentation through a multiphase level set method," *Signal Processing*, **2012**.
- [26]F. A. Pujol, et al., "On searching for an optimal threshold for morphological image segmentation," *Pattern Analysis and Applications*, vol. 14, pp. 235-250, **2011**.
- [27]M. Janev, et al., "Eigenvalues Driven Gaussian Selection in continuous speech recognition using HMMs with full covariance matrices," *Applied Intelligence*, vol. 33, pp. 107-116, **2010**.
- [28]R. N. Tamura and D. D. Boos, "Minimum Hellinger distance estimation for multivariate location and covariance," *Journal of the American Statistical Association*, vol. 81, pp. 223-229, **1986**.
- [29]A. Cutler and O. I. Cordero-Braña, "Minimum Hellinger distance estimation for finite mixture models," *Journal of the American Statistical Association*, vol. 91, pp. 1716-1723, **1996**.
- [30]V. n. Osuna-Enciso, et al., "A comparison of nature inspired algorithms for multi-threshold image segmentation," *Expert Systems with Applications*, vol. 40, pp. 1213-1219, **2013**.
- [31]D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, pp. 108-132, **2009**.
- [32]J. Vesterstrom and R. Thomsen, "A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems," in *Evolutionary Computation*, **2004**. CEC2004. Congress on, 2004, pp. 1980-1987.
- [33]E. Mezura-Montes, et al., "A comparative study of differential evolution variants for global optimization," in Proceedings of the 8th annual conference on Genetic and evolutionary computation, **2006**, pp. 485-492.
- [34]Y. Zhang and J. J. Gerbrands, "Transition region determination based thresholding," *Pattern Recognition Letters*, vol. 12, pp. 13-23, **1991**.
- [35]A. Groenewald, et al., "Related approaches to gradient-based thresholding," *Pattern Recognition Letters*, vol. 14, pp. 567-572, **1993**.
- [36]M. Burger, et al., "Incorporating topological derivatives into level set methods," *Journal of Computational Physics*, vol. 194, pp. 344-362, **2004**.
- [37]T. F. Chan and L. A. Vese, "Image segmentation using level sets and the piecewise-constant Mumford-Shah model," in Tech. Rep. 0014, Computational Applied Math Group, **2000**.
- [38]Sezgin, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic imaging*, vol. 13, pp. 146-168, **2004**.
- [39]H. Horng, "Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation," *Expert Systems with Applications*, vol. 38, pp. 13785-13791, **2011**.
- [40] Wei and F. Kangling, "Multilevel thresholding algorithm based on particle swarm optimization for image segmentation," in Control Conference, 2008. CCC 2008. 27th Chinese, **2008**, pp. 348-351.
- [41]R. Guo and S. Pandit, "Automatic threshold selection based on histogram modes and a discriminant criterion," *Machine vision and applications*, vol. 10, pp. 331-338, **1998**.
- [42]R. Whatmough, "Automatic threshold selection from a histogram using the "exponential hull"," *CVGIP: Graphical Models and Image Processing*, vol. 53, pp. 592-600, **1991**.
- [43]T.-L. Ji, et al., "Adaptive image contrast enhancement based on human visual properties," *Medical Imaging, IEEE Transactions on*, vol. 13, pp. 573-586, **1994**.
- [44]F. Catté, et al., "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM Journal on Numerical analysis*, vol. 29, pp. 182-193, **1992**.
- [45]P. G. Barten, Contrast sensitivity of the human eye and its effects on image quality vol. 21: SPIE Optical Engineering Press Washington, **1999**.
- [46]J. A. Stark, "Adaptive image contrast enhancement using generalizations of histogram equalization," *Image Processing, IEEE Transactions on*, vol. 9, pp. 889-896, **2000**.
- [47]S. S. Reddi, Rudin, S.F., Keshavan, H.R., "An Optical Multiple Threshold Scheme for Image Segmentation," *IEEE Trans System Man and Cybernetics*, vol. 14, pp. 661-665, **1984**.
- [48]T. Ridler and S. Calvard, "Picture thresholding using an iterative selection method," *IEEE transactions on Systems, Man and Cybernetics*, vol. 8, pp. 630-632, **1978**.
- [49]C.-y. Yu, et al., "A novel active contour model for image segmentation using distance regularization term," *Computers & Mathematics with Applications*, **2013**.
- [50]X. Xie, et al., "Fast Two-Stage Segmentation via Non-Local Active Contours in Multiscale Texture Feature Space," *Pattern Recognition Letters*, **2013**.
- [51]T. F. Chan and L. A. Vese, "Active contours without edges," *Image Processing, IEEE Transactions on* vol. 10, pp. 266-277, **2001**.
- [52]J. M. Prewitt, Object enhancement and extraction vol. 75: Academic Press, New York, **1970**.