International Journal of Innovative Computing, Information and Control
Volume **7**, Number **10**, October **2011**

ICIC International ©2011 ISSN 1349-4198
pp. 5631–5644

AUTOMATIC MULTILEVEL THRESHOLDING BASED ON TWO-STAGE OTSU'S METHOD WITH CLUSTER DETERMINATION BY VALLEY ESTIMATION

Deng-Yuan Huang[1], Ta-Wei Lin[1] and Wu-Chih Hu[2]

[1]Department of Electrical Engineering
Dayeh University
No. 168, University Rd., Dacun, Changhua 515, Taiwan
kevin@mail.dyu.edu.tw; daweimailbox@gmail.com

[2]Department of Computer Science and Information Engineering
National Penghu University of Science and Technology
No. 300, Liu-Ho Rd., Makung, Penghu 880, Taiwan
wchu@npu.edu.tw

Received May 2010; revised November 2010

Abstract. *The modified two-stage multithreshold Otsu (TSMO) method based on a two-stage Otsu optimization approach is proposed for multilevel thresholding. The proposed method yields the same set of thresholds as those obtained by using the conventional Otsu method, but it greatly decreases the required computation time, especially for a large number of clusters. In addition, an effective method of histogram-based valley estimations is presented for determining an appropriate number of clusters for an image. Various real-world images were used to evaluate the performance of the proposed method. Experimental results show that the speed of computation for the proposed method is about 19000 times faster than that for the conventional Otsu method when the number of clusters is 7.*

Keywords: Multilevel thresholding, Otsus method, Image segmentation

1. **Introduction.** Image multilevel thresholding is a very straightforward and effective approach that is widely used in the fields of image processing, pattern recognition and computer vision. The main goal of image segmentation is to isolate regions that represent objects or meaningful parts of objects from the rest of the image. The image is generally separated into two or more regions that are homogeneous in features such as gray level, color and texture. The practical applications of image segmentation include video security [1], human-computer interfaces (HCI) [2], optical character recognition (OCR) [3], content-based image retrieval (CBIR) [4], moving object tracking [5], image enhancement [6,7] and medical image diagnoses [8,9].

Image thresholding methods can be roughly divided into two groups: parametric and nonparametric approaches. In parametric approaches, a statistical model is first assumed to fit the gray level distribution of an image, and a set of parameters that control the fitness of the model are found using a histogram. Bazi et al. [10] proposed a parametric global thresholding method, which searches for the threshold by estimating parameters based on the expectation-maximization (EM) method under the assumption that the object and background classes follow a generalized Gaussian distribution. In nonparametric methods, thresholds are chosen by optimizing an objective function, such as maximizing between-class variance [11] or minimizing entropy [12,13]. Nonparametric approaches have proven to be more accurate and robust than parametric ones.

For bilevel thresholding, Otsu's method [11] is a very popular nonparametric thresholding technique, which searches for an optimum threshold by maximizing the between-class variance in a gray level image. However, the objects and backgrounds in real-world images are usually complicated, so bilevel thresholding cannot always achieve satisfactory results. Therefore, multilevel thresholding [14-16], which determines many thresholds to segment images into several clusters, was developed. Although the Otsu method can be easily extended to a multi-threshold version, it suffers from a serious drawback of computational inefficiency when the number of segmented clusters is large. This is due to the fact that Otsu's thresholding involves iterations of the zero- and first-order cumulative moments of a gray level histogram, which requires a great number of multiplication and division manipulations.

Several improved versions of Otsu's method have been developed to reduce the computational cost for multilevel thresholding [14-16]. Liao et al. [14] proposed a modified version of Otsu's method, called recursive Otsu's method, to find the optimal thresholds by accessing pre-computed modified between-class variance values in a look-up table (i.e., H-table). Their method greatly improves the computational efficiency when the number of clusters is relatively small, but it still suffers from poor performance when the number of clusters is high (number of clusters ≥ 5). Dong et al. [15] presented an iterative algorithm for finding optimal thresholds based on the minimization of a weighted sum-of-squared-error objective function, which was proven to be mathematically equivalent to the Otsu method, but it requires a much lower number of computations.

Huang and Wang [16] proposed a two-stage approximation of Otsu's method, called the two-stage multithreshold Otsu (TSMO) method, to improve performance. In the first stage, bin grouping and then multilevel Otsu's thresholding are used for finding the bin groups (or sets) with the maximum between-class variance. In the second stage, bilevel Otsu's thresholding is used to refine each threshold within the set that contains the optimal threshold. Since the gray levels in the histogram are decreased from 256 to 32 sets in the TSMO method, the time required is greatly reduced. However, the sets of thresholds determined by the TSMO method and Otsu's method are slightly different because the estimation of the threshold in stage two of the TSMO method is confined to each chosen set.

Most multilevel thresholding algorithms [10,12-16] can segment objects of interest from their background well, but the number of clusters must be manually determined in advance. In recent years, some methods, such as morphological watersheds [17] and isoperimetric ratios [18,19], have been developed to automatically determine the number of clusters. In [17], Liu et al. employed the watershed method to determine the number of clusters using a gray-level histogram and then used a fuzzy C-mean (FCM) method to separate the object of interest (i.e., a cucumber plant) from complicated backgrounds. Recently, image segmentation based on isoperimetric graph theory has become a popular method in research for determining the number of clusters for multilevel thresholding problems [18,19]. However, due to their high computation complexity and poor real-time performance, these techniques are rarely applied to actual image segmentation problems. Sahoo et al. [20] presented a comprehensive survey of thresholding approaches and concluded that Otsu's method was one of the best thresholding methods for image segmentation with respect to uniformity and shape measures.

In the last decade, Sankur and Sezgin [21] conducted an exhaustive survey of 40 selected image thresholding methods. The results show that the thresholding evaluation rank of 40 NDT (nondestructive testing) images according to the overall average quality score for Otsu's method is relatively high up to rank 6. It means that Otsu's method can provide reasonable thresholds for image segmentation. This result motivates us to propose a method that not only obtains the same thresholds as those of Otsu's method, but also significantly improves the computational inefficiency of the Otsu method. Recently, the research of automatic determination of the number of clusters is increasingly popular. However, thus far most of the multilevel thresholding methods are either lack of the capability of determining the number of clusters [10,12-16], or computational inefficiency to determine clusters [17-19].

To solve the issues mentioned above, a multilevel thresholding method based on a two-stage Otsu's optimization method with automatic cluster determination by valley estimation for image segmentation is proposed. The main advantages of the results over other studies for the proposed method are clear in that it has the following two unique features: (1) it provides the same set of thresholds as those obtained by using the conventional Otsu method [11], but it completely avoids the heavy computational cost of Otsu's method especially for a larger number of clusters. In addition, the proposed method's efficiency is comparable to that of the TSMO method; (2) it can automatically and efficiently determine the number of clusters using a histogram-based valley estimation method, which eliminates both the high computation complexity and poor real-time performance of watershed [17] and isoperimetric graph partitioning [18,19] methods. To verify the performance of the proposed method, several real-world images are used. Experimental results show that the speed of computation for the proposed method is about 19000 times faster than that for the conventional Otsu method when the number of clusters is 7. This result confirms the effectiveness and superiority of the proposed method.

The rest of this paper is organized as follows: in Section 2, brief overviews of multilevel Otsu's thresholding method and the TSMO method are given; in Section 3, the determination of the number of clusters using valley estimation is introduced and the proposed method for multilevel thresholding is described; in Section 4, the results for the proposed method and those for the TSMO and Otsu methods are compared and discussed; Section 5 concludes the paper.

2. Overview of Otsu's Method and the TSMO Method.

In this section, the extended version of Otsu's method for multilevel thresholding problems is described. The principle of the TSMO method is then outlined in Section 2.2.

2.1. Overview of multilevel Otsu's thresholding method.

Otsu's method [11] is popular for image segmentation. It selects a global threshold value by maximizing the separability of the clusters in gray levels. Assume that an image can be represented in L gray levels (0, 1, ..., $L-1$). The number of pixels at level i is denoted by f_i; then, the total number of pixels equals $N = f_0 + f_1 + \ldots + f_{L-1}$. For a given gray level image, the occurrence probability of gray level i is given by:

$$p_i = \frac{f_i}{N}, \quad p_i \geq 0, \quad \sum_{i=0}^{L-1} p_i = 1 \tag{1}$$

If an image is segmented into K clusters $(C_0, C_1, \ldots, C_{K-1})$, $K-1$ thresholds $(t_0, t_1, \ldots, t_{K-2})$ must be selected. The cumulative probability w_k and mean gray level μ_k for each cluster C_k are respectively given by:

$$w_k = \sum_{i \in C_k} p_i \text{ and } \mu_k = \sum_{i \in C_k} i \cdot p_i / \omega_k, k \in \{0, 1, \ldots, K-1\} \tag{2}$$

Therefore, the mean intensity of the whole image μ_T and the between-class variance σ_B^2 are respectively determined by:

$$\mu_T = \sum_{i=0}^{L-1} i \cdot p_i = \sum_{k=0}^{K-1} \mu_k \omega_k \tag{3}$$

and

$$\sigma_B^2 = \sum_{k=0}^{K-1} \omega_k \left(\mu_k - \mu_T\right)^2 = \sum_{k=0}^{K-1} \omega_k \mu_k^2 - \mu_T^2 \tag{4}$$

Hence, the optimal thresholds $(t_0^*, t_1^*, \ldots, t_{K-2}^*)$ can be determined by maximizing the between-class variance as:

$$\left\{t_0^*, t_1^*, \cdots, t_{K-2}^*\right\} = \underset{0 \le t_0 < \ldots < t_{K-2} < L-1}{\arg} \max \left\{\sigma_B^2(t_0, t_1, \cdots, t_{K-2})\right\} \tag{5}$$

2.2. **Overview of the TSMO method.** TSMO [16] uses the concept of bin grouping in the first stage to reduce the computation complexity of the original Otsu method. The histogram of L $(= 256)$ gray levels is divided into Mz sets that contain Nz $(= 256/Mz)$ gray levels. In this paper, Mz is set to 32, with $Nz = 8$. Let Ω represent the sets of the total image space; then, $\Omega = \{\Omega q | q = 0, 1, \cdots, Mz - 1\}$, where q represents the set number. Therefore, the occurrence probability $p_{\Omega q}$ and the mean intensity $i_{\Omega q}$ in the qth set, i.e., Ωq, are respectively given by:

$$p_{\Omega q} = \sum_{i \in \Omega q} p_i, \quad \text{and} \quad i_{\Omega q} = \sum_{i \in \Omega q} i \cdot p_i \Big/ \sum_{i \in \Omega q} p_i = \frac{1}{p_{\Omega q}} \sum_{i \in \Omega q} i \cdot p_i \tag{6}$$

Similarly, the cumulative probability w_k and mean gray level μ_k for each cluster C_k $(k = 0, 1, \cdots, K - 1)$ can be respectively estimated as:

$$w_k = \sum_{\Omega q \in C_k} p_{\Omega q} \quad \text{and} \quad \mu_k = \sum_{\Omega q \in C_k} i_{\Omega q} \cdot p_{\Omega q} / \omega_k \tag{7}$$

Therefore, the mean intensity of the whole image μ_T and the between-class variance σ_B^2 can be respectively determined as:

$$\mu_T = \sum_{q=0}^{Mz-1} i_{\Omega q} \cdot p_{\Omega q} = \sum_{k=0}^{K-1} \mu_k \omega_k \tag{8}$$

and

$$\sigma_B^2 = \sum_{k=0}^{K-1} \omega_k \left(\mu_k - \mu_T\right)^2 = \sum_{k=0}^{K-1} \omega_k \mu_k^2 - \mu_T^2 \tag{9}$$

In the first stage of the TSMO method, the sets with the numbers $\left\{q_0^*, q_1^*, \cdots, q_{K-2}^*\right\}$ can be determined using multilevel Otsu's thresholding method. That is:

$$\left\{q_0^*, q_1^*, \ldots, q_{K-2}^*\right\} = \underset{0 \le q_0 < \ldots < q_{K-2} < Mz-1}{\arg} \max \left\{\sigma_B^2(q_0, q_1, \ldots, q_{K-2})\right\} \tag{10}$$

Therefore, the possible optimal threshold for each set should fall into the chosen sets $\left\{\Omega_{q_0^*}, \Omega_{q_1^*}, \cdots, \Omega_{q_{K-2}^*}\right\}$. Then, in stage two of the TSMO method, bilevel Otsu's thresholding method is used to determine the optimal threshold t_k^* for each chosen set $\Omega_{q_k^*}$ $(k = 0, 1, \cdots, K - 2)$ as:

$$t_k^* = \underset{t_k \in \Omega_{q_k^*}}{\arg} \max \left\{\sigma_B^2(t_k)\right\}, \quad k = 0, 1, \cdots, K - 2 \tag{11}$$

3. **The Proposed Method for Determining the Number of Clusters and Multilevel Thresholds.** A flowchart of the proposed method is shown in Figure 1. The histogram of input gray levels is first statistically computed. The method of valley estimation is then used to determine the number of clusters for the image to be properly segmented. Furthermore, a modified version of the TSMO method is proposed to provide the same set of thresholds as those obtained using the conventional Otsu method. The high computation complexity of Otsu's method is completely avoided, especially when the number of clusters is relatively high. The valley estimation method is described in Section 3.1, and the modified version of the TSMO method is given in Section 3.2.

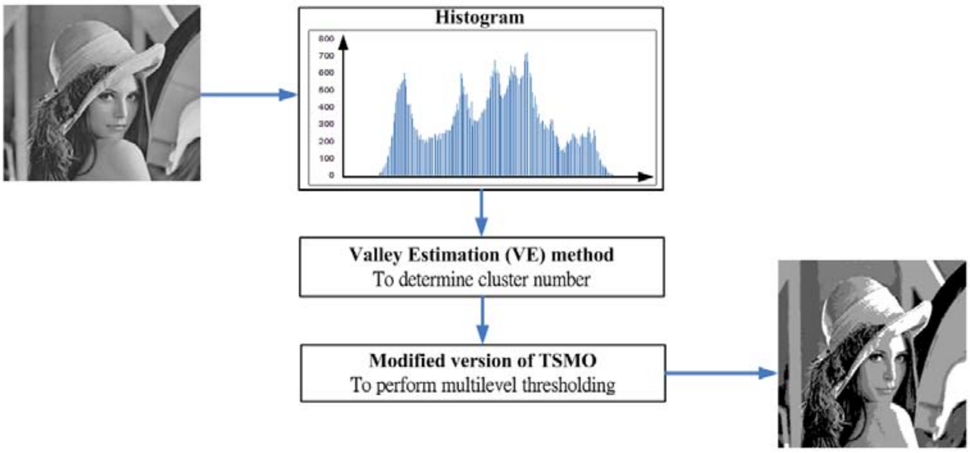

Histogram
800
700
600
500
400
300
200
100
0
Valley Estimation (VE) method
To determine cluster number
Modified version of TSMO
To perform multilevel thresholding

Figure 1. Flowchart of the proposed method for multilevel thresholding

3.1. **Determination of the number of clusters using the histogram-based valley estimation method.** In this paper, a histogram-based valley estimation method (see Figure 2) is proposed for determining the number of clusters for an image to be properly segmented. The approach consists of the following three procedures: (1) normalized histogram binning, (2) probability estimation and (3) valley estimation.

To complete the bin grouping statistically (see Figure 2(b)), 32 counters containing 8 pixels, denoted as C_0 to C_{31}, are used. Hence, when the gray levels are in the ranges of 0 to 7, 8 to 15, ..., and 248 to 255, they will be grouped into counters C_0, C_1, ..., and C_{31}, respectively. Therefore, the normalized histogram binning process is used to estimate the probability of occurrence in bin group C_i as:

$$H_i = \frac{h_i}{\max(h_i)} \times 100, \quad i \in \{0, 1, \cdots, 31\}, \quad \text{where } h_i = \sum_{i \in C_i} f_i \tag{12}$$

where f_i represents the number of pixels at gray level i, and H_i is called a normalized histogram of bin group C_i, which has a range of values from 0 to 100. Normalized histogram binning reduces noise since bin grouping operates like a smoothing spatial filter. Bin grouping also greatly increases the stability of the valley estimation process.

The purpose of probability estimation is to determine the probability of one bin becoming a valley location in a histogram distribution. In this stage, the bin groups are scanned from C_1 to C_{30} (see Figure 2(c)). The scanning counter, C_i, is assigned a probability $\Pr(C_i)$ of 0%, 25%, 75%, 100% or $\Pr(C_{i-1})$ according to the normalized histogram distribution (H_{i-1}, H_i, H_{i+1}) of the counters (C_{i-1}, C_i, C_{i+1}) (see Figure 3). Figure 3 shows the 9 possible combinations of the bin group C_i between C_{i-1} and C_{i+1}, where in

3(a)-3(e), the bin group C_i is not the candidate of a valley location since the probability of occurrence of C_i is greater than that of either C_{i-1} or C_{i+1}, and where in 3(f)-3(h) whether the bin group C_i is a valley location cannot be determined. The probabilities for the bin group to become a valley location are set to 25% for Figure 3(f) and to 75% for Figure 3(h), as determined from experience. In Figure 3(i), the bin group C_i is a valley location because the probability of occurrence of C_i is less than those of both C_{i-1} and C_{i+1}.

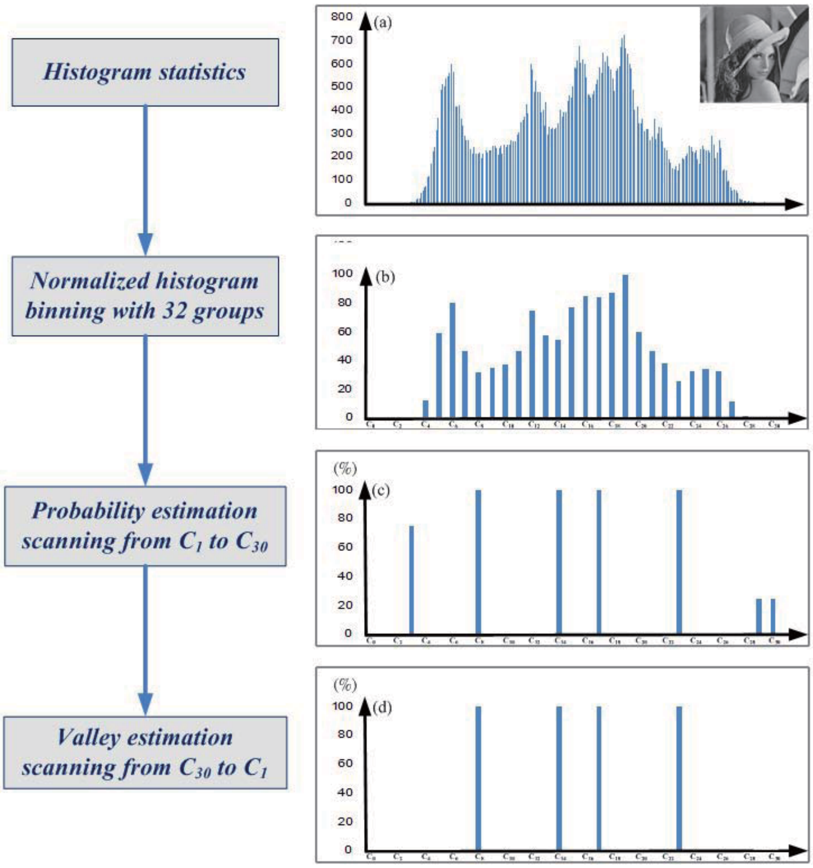

Histogram statistics
Normalized histogram binning with 32 groups
Probability estimation scanning from C_1 to C_{30}
Valley estimation scanning from C_{30} to C_1
(a)
800
700
600
500
400
300
200
100
0
(b)
100
80
60
40
20
0
(%)
(c)
100
80
60
40
20
0
(%)
(d)
100
80
60
40
20
0

Figure 2. Flowchart of the proposed valley estimation method: (a) Histogram statistics; (b) normalized histogram binning with 32 sets; (c) probability estimation and (d) valley estimation

Therefore, the probabilities for the 9 combinations of bin group C_i can be given as follows:

$$\begin{cases} \Pr(C_i) = 0\% & \text{if } (H_i > H_{i-1}) \text{ or } (H_i > H_{i+1}) \\ \Pr(C_i) = 25\% & \text{if } (H_i < H_{i-1}) \text{ and } (H_i = H_{i+1}) \\ \Pr(C_i) = 75\% & \text{if } (H_i = H_{i-1}) \text{ and } (H_i < H_{i+1}) \\ \Pr(C_i) = 100\% & \text{if } (H_i < H_{i-1}) \text{ and } (H_i < H_{i+1}) \\ \Pr(C_i) = \Pr(C_{i-1}) & \text{if } (H_i = H_{i-1}) \text{ and } (H_i = H_{i+1}) \\ \Pr(C_0) = \Pr(C_{31}) = 0\% & \end{cases} \quad i \in \{1, 2, \cdots, 30\} \tag{13}$$

where $\Pr(C_i) = 0\%$ means that the probability of bin group C_i to be a valley location is zero, which corresponds to the cases in Figures 3(a)-3(e). The probabilities of $\Pr(C_i)$=25%, 75% and 100% represents the cases in Figures 3(f), 3(h) and 3(i), respectively. However, if the probabilities of occurrence of the three counters (C_{i-1}, C_i, C_{i+1}) are the same, which corresponds to the case of Figure 3(g), the probability of C_i is then

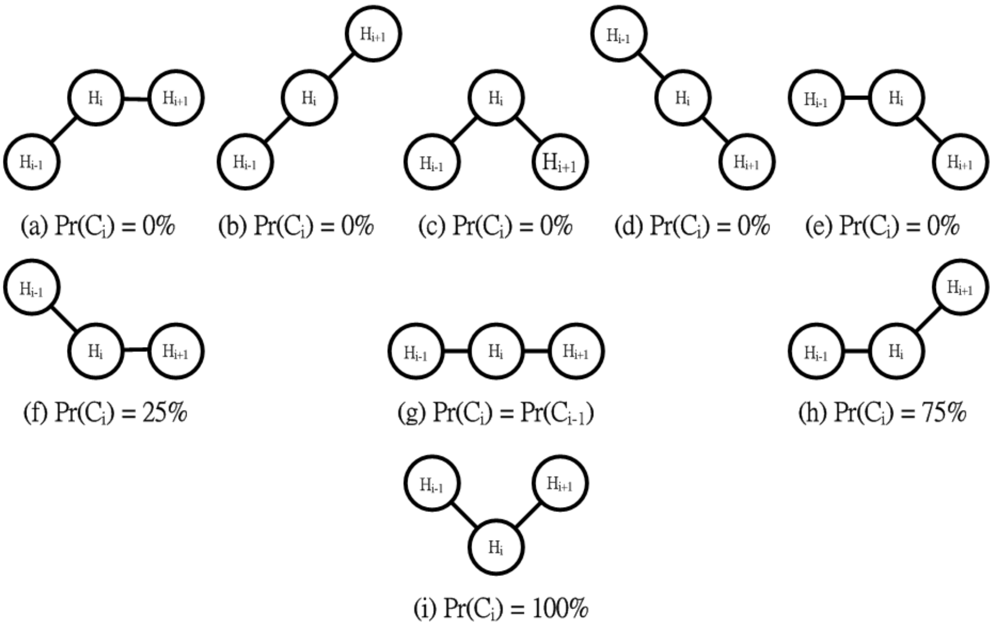


Figure 3. Nine combinations of the normalized histogram distribution (H_{i-1}, H_i, H_{i+1}) and their corresponding probability of the counter C_i when scanning from C_1 to C_{30}

set to that of C_{i-1}. Additionally, the probability of 0% is used for bin groups C_0 and C_{31} because these two locations are not the valleys.

For the valley estimation stage, the bin groups are scanned in the opposite direction, from C_{30} to C_1, as shown in Figure 2(d). The probabilities ($\Pr(C_{i-1})$, $\Pr(C_i)$ and $\Pr(C_{i+1})$) determined in the previous stage (i.e., probability estimation) are then added together only if the probability of the scanning counter C_i is not equal to zero. If the sum of probabilities is higher than or equal to 100%, the probability of C_i is reset to 100%; otherwise, it is 0%. This is given by:

$$\begin{cases} \Pr(C_i) = 0\% & \text{if} \quad (\Pr(C_{i-1}) + \Pr(C_i) + \Pr(C_{i+1})) < 100\% \\ \Pr(C_i) = 100\% & \text{if} \quad (\Pr(C_{i-1}) + \Pr(C_i) + \Pr(C_{i+1})) \geq 100\% \quad i \in \{30, 29, \cdots, 1\} \\ \Pr(C_0) = \Pr(C_{31}) = 0 & \end{cases} \tag{14}$$

where the bin group C_i with $\Pr(C_i) = 100\%$ is the valley location. After all the valley locations are summed, the number of clusters is determined.

3.2. **Proposed method for multilevel thresholding.** As mentioned in Section 2.2, the TSMO method first generates a coarse partition of the input gray levels into Mz sets, and estimates a set of thresholds for these sets in the first stage. Then, the TSMO method searches for each threshold within the set that contains each threshold in the second stage (see Equation (11)). This leads to a difference of thresholds found by the TSMO method and Otsu's method. In this paper, a modified version of the TSMO method that can provide the same set of thresholds as those obtained using the original Otsu method but without the high computational efforts is presented.

As illustrated in Figure 4, the multilevel thresholds selected by the TSMO method are always close to the one decided by bilevel Otsu's thresholding. This indicates that the distribution of between-class variance σ_B^2 for bilevel thresholding forms approximately a single mode (see Figure 4). In this case, the bilevel threshold of 133 is obtained by the conventional Otsu method. However, the 5-level thresholds determined by the TSMO method are 79, 119, 160 and 200, which fall into the sets C_9 (72~79), C_{14} (112~119), C_{20} (160~167) and C_{25} (200~207), respectively, where the values in parentheses represent the range of gray levels in group C_i. Clearly, the values of 79 in group C_9, 119 in group C_{14}, 160 in group C_{20} and 200 in group C_{25}, are the closest values to 133 for each bin group, respectively. This occurs because the TSMO method searches for each threshold using bilevel Otsu's method confined to each set that is chosen by the first stage.

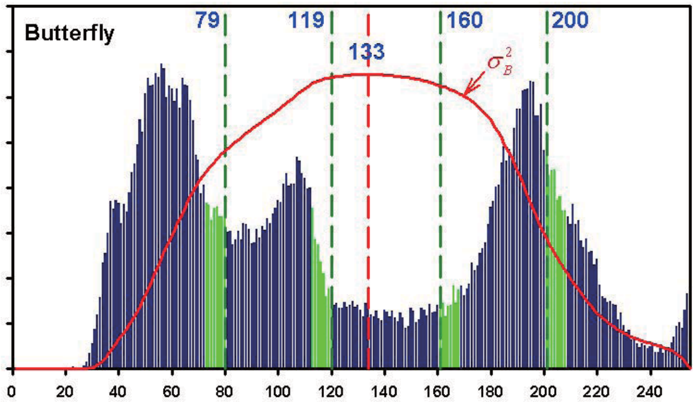


FIGURE 4. Thresholds of 79, 119, 160 and 200 chosen by the TSMO method ($K = 5$) (green dashed lines) for each bin group and bilevel threshold of 133 determined by Otsu's method (red dashed line). The solid red line represents the distribution of between-class variance for bilevel Otsu's thresholding.

The TSMO method uses bin grouping to speed up the searching process of sets (or bin groups) that contain each threshold in the first stage. For bilevel thresholding, TSMO (using 32 bins) finds the set (say Ω_q) that has the maximum between-class variance (see Figure 5(a)). However, the optimal threshold determined by Otsu's method (using 256 gray levels) may fall into the set Ω_{q+1}, which is immediately to the right of Ω_q (see Figure 5(b)). This result can be extended to the cases of multilevel thresholding. Therefore, in the second stage of the TSMO method, a bigger bin group extending the set $\{\Omega_q\}$ to $\{\Omega_q + \Omega_{q+1}\}$ should be considered to find the thresholds. A method that provides the same set of thresholds as those obtained using the original Otsu method, where only the second stage of the TSMO method is modified, is as follows:

$$\left\{t_0^*, t_1^*, \cdots, t_{K-2}^*\right\} = \underset{t_0\in\left\{\Omega_{q_0^*}+\Omega_{q_0^*+1}\right\},\cdots,t_{K-2}\in\left\{\Omega_{q^*_{(K-2)}}+\Omega_{q^*_{(K-2)}+1}\right\}}{\arg\max}\left\{\sigma_B^2(t_0, t_1, \cdots, t_{K-2})\right\} \tag{15}$$

The computation complexity of the proposed method is estimated as $(Mz - K + 1)^{K-1} + (2 \times Nz)^{K-1}$ (note that $Mz = 32$ and $Nz = 8$ in this work). The first term $(M_z - K + 1)^{K-1}$ involves the possible combinations of the search in the first stage for finding the sets $\left\{\Omega_{q_0^*}, \Omega_{q_1^*}, \cdots, \Omega_{q_{K-2}^*}\right\}$ into which the possible optimal thresholds fall, and the second term $(2 \times Nz)^{K-1}$ is required for the second stage to determine the optimal thresholds $\left\{t_0^*, t_1^*, \cdots, t_{K-2}^*\right\}$. The computation complexities for the TSMO method and Otsu's method are $(Mz - K + 1)^{K-1} + (Nz - 2 + 1) \times (K - 1)$ and $(L - K + 1)^{K-1}$ ($L = 256$), respectively. Hence, the difference of computation complexity between the proposed method and the TSMO method is in the second term. For example, the number

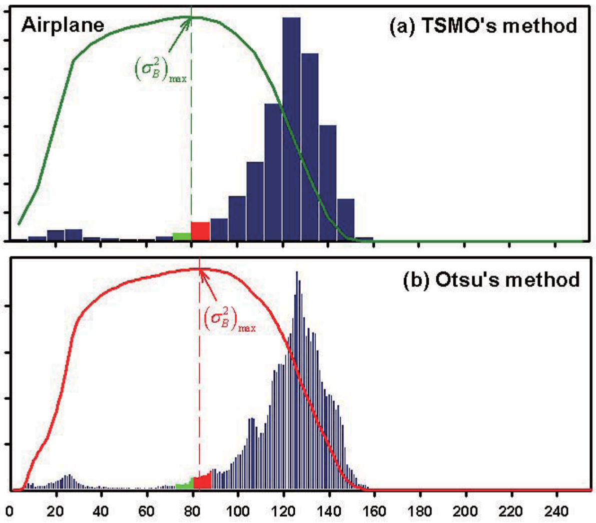


Figure 5. (a) Bilevel thresholding obtained using the TSMO method and (b) bilevel thresholding obtained using Otsu's method

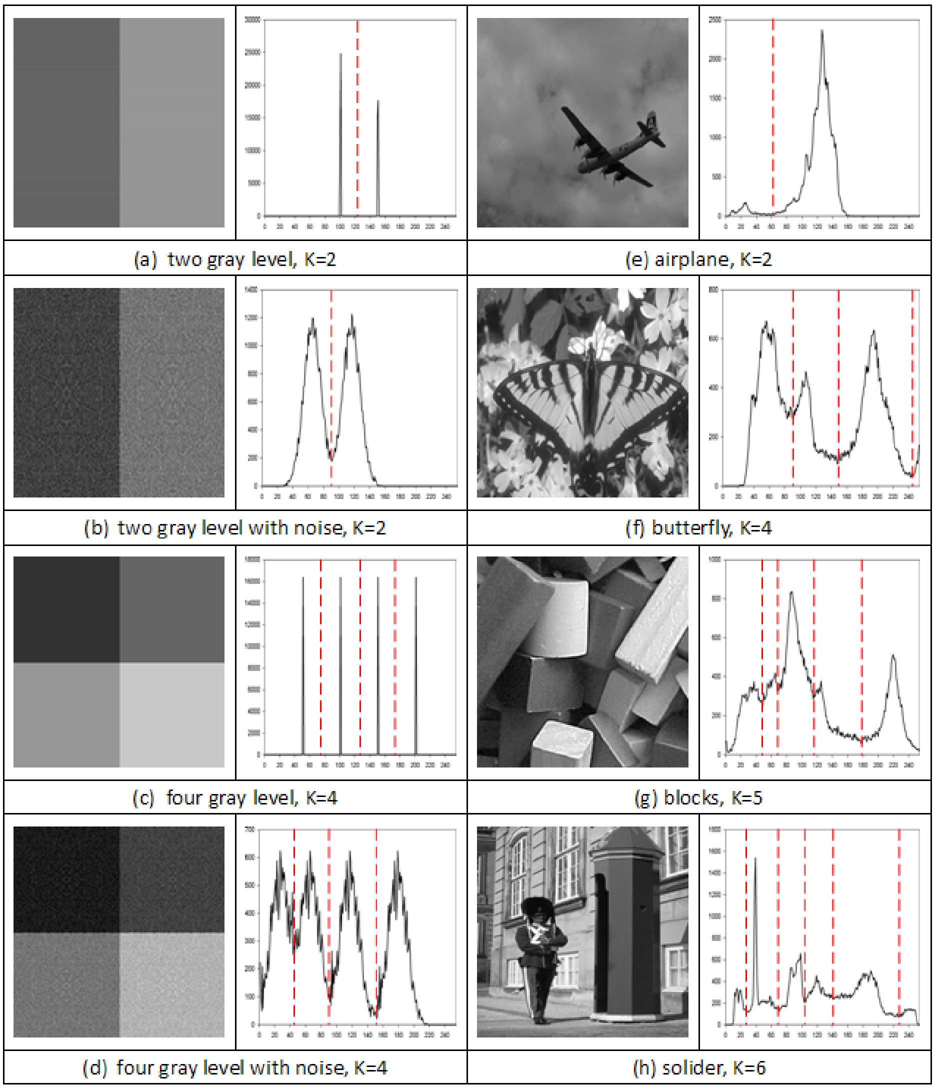
of required iterations for the TSMO method and the proposed method in the second stage are 42 and 16,777,216, respectively, if the cluster number K is set to 7.

4. **Results and Discussion.**

To evaluate the efficiency of the proposed method, the results are compared with those of Otsu's method and the TSMO method. The efficiency evaluations for the three methods were carried out on a computer with a Pentium 3.4 GHz Core Duo processor and 4 GB DDR II memory. In this section, the accuracy of the determined number of clusters using the valley estimation method is first discussed. The efficiency of the proposed method is then compared with those of the TSMO method and Otsu's method.

To evaluate the accuracy of the valley estimation method for determining the optimal number of clusters, two images that have two and four gray levels, shown in Figures 6(a) and 6(c), respectively, were used. To verify the robustness of the proposed method, some Gaussian noise with a standard deviation $\sigma = 10$ was added to the two images, as shown in Figures 6(b) and 6(d), respectively. In Figure 6, the numbers of clusters, denoted as K, for the four images were correctly determined by the proposed method. The proposed method was also used to segment images *Airplane*, *Butterfly*, *Blocks* and *Soldier*, which were acquired from [22], as shown in Figures 6(e)-6(h), respectively. The method found optimal numbers of clusters of 2, 4, 5 and 6, respectively. The thresholds estimated for $K = 2$ to 7 clusters are shown in Figure 7, in which the proposed method and Otsu's method yield the same results.

In the proposed method, the determination of the number of clusters highly depends on the shape of the histogram of gray level images. For example, if 32 bin groups are used, the histogram of Figure 8(a) is shown in Figure 8(b). Clearly, no valleys in Figure 8(b) can be found, indicating that the proposed valley estimation method cannot be applied any more. If this is the case, 32 bin groups should be increased to 64 ones. In Figure 8(c), several valleys in the histogram can be found when 64 bin groups are used, indicating that the proposed valley estimation method can be properly utilized. In our experiments, the test images of which have the same histogram as that of Figure 8(b) are quite few. However, if the case really happens, the use of 64 bin groups can provide us a reasonable estimation of the number of clusters.



(a) two gray level, K=2

(e) airplane, K=2

(b) two gray level with noise, K=2

(f) butterfly, K=4

(c) four gray level, K=4

(g) blocks, K=5

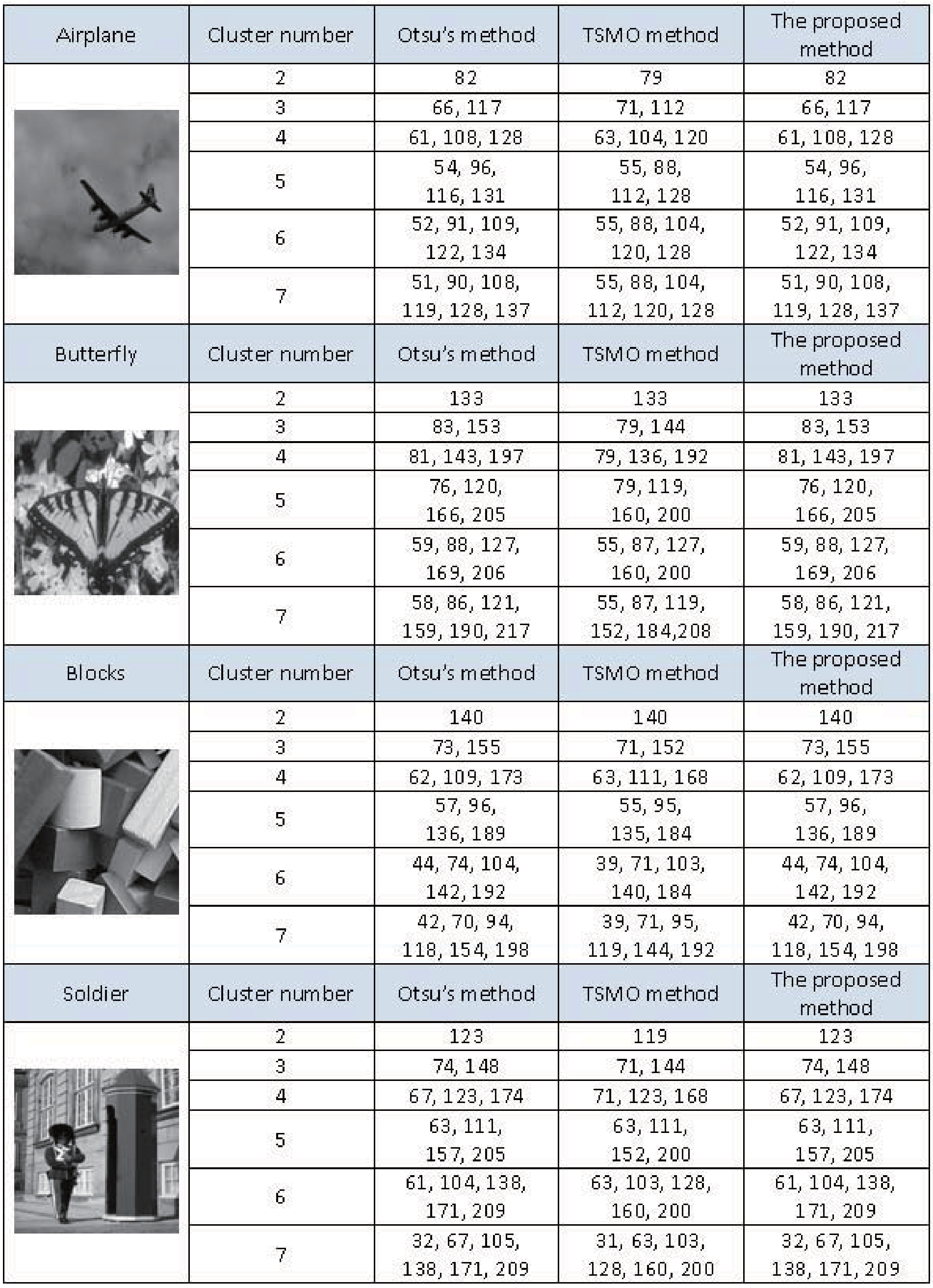
(d) four gray level with noise, K=4

(h) solider, K=6

Figure 6. Results of the proposed valley estimation method for determining the number of clusters for images to be properly segmented. The red dashed lines represent the thresholds determined. (a) Two gray levels; (b) two gray levels with noise; (c) four gray levels; (d) four gray levels with noise; (e) *Airplane*; (f) *Butterfly*; (g) *Blocks* and (h) *Soldier*.

As described earlier, the thresholds determined by the TSMO method are close to that estimated by bilevel Otsu's thresholding. For instance, the bilevel threshold for image *Blocks* is 140 for Otsu's method. The thresholds of 63, 111 and 168 chosen by the TSMO method for 4 clusters are the closest ones to 140 for each group, where the thresholds of 63, 111 and 168 fall into groups C_7 (56∼63), C_{13} (104∼111) and C_{21} (168∼175), respectively. Finally, the segmented results, optimal number of clusters, and thresholds determined by the proposed method are shown in Figure 9. As indicated in the figure, the segmentation results of image *Airplane* show the ability of the proposed method to completely isolate the object of interest (the airplane) from its background.

To evaluate the overall performance of the proposed method (i.e., the modified TSMO method) versus Otsu's method and the TSMO method, the four test images (i.e., *Airplane*, *Butterfly*, *Blocks* and *Soldier*) in Figure 7 are used. The runtimes of the number of clusters



Airplane	Cluster number	Otsu's method	TSMO method	The proposed method
	2	82	79	82
	3	66, 117	71, 112	66, 117
	4	61, 108, 128	63, 104, 120	61, 108, 128
	5	54, 96, 116, 131	55, 88, 112, 128	54, 96, 116, 131
	6	52, 91, 109, 122, 134	55, 88, 104, 120, 128	52, 91, 109, 122, 134
	7	51, 90, 108, 119, 128, 137	55, 88, 104, 112, 120, 128	51, 90, 108, 119, 128, 137
Butterfly	Cluster number	Otsu's method	TSMO method	The proposed method
	2	133	133	133
	3	83, 153	79, 144	83, 153
	4	81, 143, 197	79, 136, 192	81, 143, 197
	5	76, 120, 166, 205	79, 119, 160, 200	76, 120, 166, 205
	6	59, 88, 127, 169, 206	55, 87, 127, 160, 200	59, 88, 127, 169, 206
	7	58, 86, 121, 159, 190, 217	55, 87, 119, 152, 184,208	58, 86, 121, 159, 190, 217
Blocks	Cluster number	Otsu's method	TSMO method	The proposed method
	2	140	140	140
	3	73, 155	71, 152	73, 155
	4	62, 109, 173	63, 111, 168	62, 109, 173
	5	57, 96, 136, 189	55, 95, 135, 184	57, 96, 136, 189
	6	44, 74, 104, 142, 192	39, 71, 103, 140, 184	44, 74, 104, 142, 192
	7	42, 70, 94, 118, 154, 198	39, 71, 95, 119, 144, 192	42, 70, 94, 118, 154, 198
Soldier	Cluster number	Otsu's method	TSMO method	The proposed method
	2	123	119	123
	3	74, 148	71, 144	74, 148
	4	67, 123, 174	71, 123, 168	67, 123, 174
	5	63, 111, 157, 205	63, 111, 152, 200	63, 111, 157, 205
	6	61, 104, 138, 171, 209	63, 103, 128, 160, 200	61, 104, 138, 171, 209
	7	32, 67, 105, 138, 171, 209	31, 63, 103, 128, 160, 200	32, 67, 105, 138, 171, 209

FIGURE 7. Set of thresholds determined by the three methods for 2 to 7 clusters for the images *Airplane*, *Butterfly*, *Blocks* and *Soldier*

of 4, 5, 6 and 7 for the four test images are estimated, respectively. The mean value of a runtime T^K_{mean} for different clusters can be estimated as follows:

$$T^K_{mean} = \frac{N_T \times \left(T^K_{Airplane} + T^K_{Butterfly} + T^K_{Blocks} + T^K_{Soldier}\right)}{4 \times N_T} \tag{16}$$

where N_T denotes the total number of calculations, and T^K_X is the individual runtime of K clusters for image X, i.e., *Airplane*, *Butterfly*, *Blocks* and *Soldier*. In our experiments, N_T is taken as 1000 for the TSMO method and the proposed method. However, since the Otsu method is quite time consuming, N_T is taken as 1000, 20, 2 and 1 for $K = 4$, 5, 6 and 7, respectively. The runtimes for various numbers of clusters are listed in Table 1.

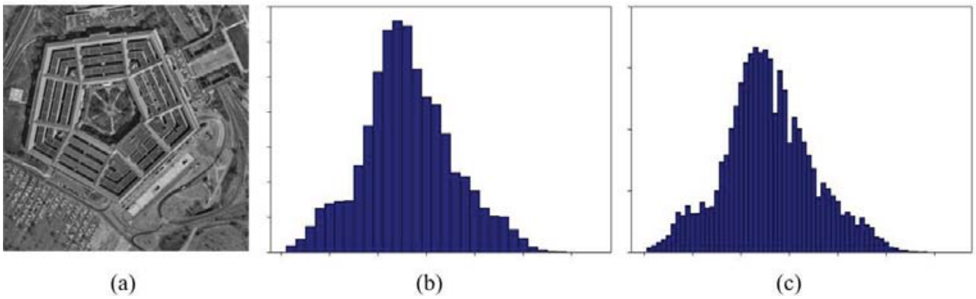


FIGURE 8. Illustration of the effect of different bin groups on the proposed valley estimation method for determining the number of clusters. (a) The image of *The Pentagon*; (b) histogram with 32 bin groups ($Mz = 32$); (c) histogram with 64 bin groups ($Mz = 64$).

The ratios of runtimes of the Otsu method to the proposed method are about 1580, 1760, 6290 and 19000 when the number of clusters is 4, 5, 6 and 7, respectively.

TABLE 1. Runtimes of the three methods for $K = 4$ to 7 (unit: seconds)

Methods	$K = 4$	$K = 5$	$K = 6$	$K = 7$
Otsu's method	1.29700	110.102	6098.64	299707
TSMO method	0.00023	0.03899	0.19500	0.89100
The proposed method	0.00082	0.06250	0.96900	15.7500

5. **Conclusions.**

Otsu's method is widely used in the fields of video surveillance, computer vision, and pattern recognition as a low-level image processing technique for isolating objects of interest from their backgrounds. It is one of the best thresholding methods for image segmentation with respect to uniformity and shape measures. However, a serious drawback of Otsu's method is high computational complexity when extended to a multilevel thresholding problem.

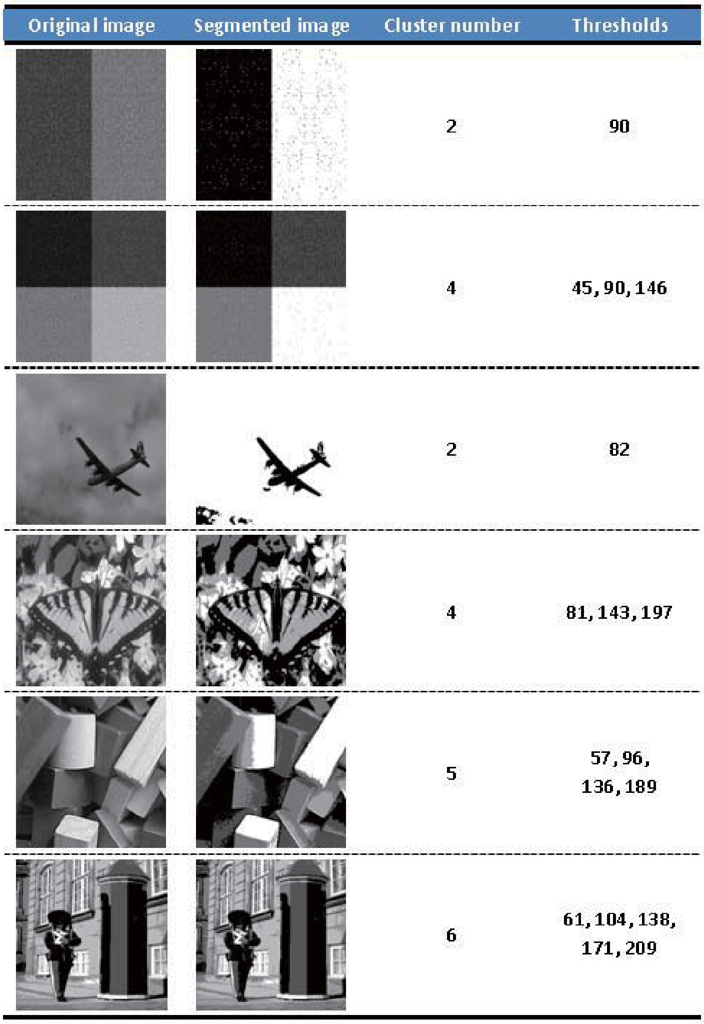
In this paper, a modified TSMO method based on a two-stage Otsu optimization approach for multilevel thresholding was proposed. The major contribution of this paper is to propose a method that can yield the same set of thresholds as those obtained using Otsu's method but without the high computational efforts when the number of clusters is large. The proposed method achieves real-time performance for multilevel thresholding when the number of clusters is fewer than 6.

In addition, an effective method of histogram-based valley estimation was presented to determine the appropriate number of clusters for an image. To evaluate the performance of the proposed method, various real-world images were segmented. Experimental results show that the speed of computation for the proposed method is about 19000 times faster than that for the conventional Otsu method when the number of clusters is 7.

Acknowledgment. This work is partially supported by NSC 97-2221-E-212-035. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] K. Toyama, J. Krumm, B. Brumitt and B. Meyers, Wallflower: Principles and practice of background maintenance, *Proc. of the 7th IEEE Conf. on Computer Vision*, Kerkyra, Greece, pp.255-261, 1999.



Original image	Segmented image	Cluster number	Thresholds
		2	90
		4	45,90,146
		2	82
		4	81,143,197
		5	57,96, 136,189
		6	61,104,138, 171,209

FIGURE 9. Number of clusters and thresholds by the proposed method with histogram-based valley estimation. The segmentation results are shown in the second column for the tested images. The corresponding histogram for each test image is shown in Figure 6.

[2] S. Arseneau and J. R. Cooperstock, Real-time image segmentation for action recognition, *Proc. of IEEE Pacific Rim Conf. on Communications, Computers and Signal Processing*, Victoria, B.C., Canada, pp.86-89, 1999.

[3] A. T. Abak, U. Baris and B. Sankur, The performance evaluation of thresholding algorithms for optical character recognition, *Proc. of IEEE Int. Conf. Document Analysis and Recognition*, Ulm, Germany, pp.697-700, 1997.

[4] T. T. Zin, H. Hama and P. Tin, Spatial image retrieval based on dynamic thresholding, *International Journal of Innovative Computing, Information and Control*, vol.5, no.11(B), pp.4051-4059, 2009.

[5] S. Y. Chien, Y. W. Huang, B. Y. Hsieh, S. Y. Ma and L. G. Chen, Fast video segmentation algorithm with shadow cancellation, global motion compensation, and adaptive threshold techniques, *IEEE Trans. on Multimedia*, vol.6, no.5, pp.732-748, 2004.

[6] J. Pan, C. Zhang and Q. Guo, Image enhancement based on the shearlet transform, *ICIC Express Letters*, vol.3, no.3(B), pp.621-626, 2009.

[7] Y. Zhang, C. Zhang, J. Chi and R. Zhang, An algorithm for enlarged image enhancement, *ICIC Express Letters*, vol.3, no.3(B), pp.669-674, 2009.

[8] M. S. Atkins and B. T. Mackiewich, Fully automatic segmentation of the brain in MRI, *IEEE Trans. on Med. Imaging*, vol.17, no.1, pp.98-107, 1998.

[9] S.-G. Shu, H.-H. Lin, S.-W. Kuo and S.-S. Yu, Excluding background initial segmentation for radiographic image segmentation, *International Journal of Innovative Computing, Information and Control*, vol.5, no.11(A), pp.3849-3860, 2009.

[10] Y. Bazi, L. Bruzzone and F. Melgani, Image thresholding based on the EM algorithm and the generalized Gaussian distribution, *Pattern Recognit.*, vol.40, no.2, pp.619-634, 2007.

[11] N. Otsu, A threshold selection method from gray-level histograms, *IEEE Trans. on Syst., Man, Cybern.*, vol.9, pp.62-66, 1979.

[12] S. D. Zenzo, L. Cinque and S. Levialdi, Image thresholding using fuzzy entropies, *IEEE Trans. on Syst., Man, Cybern., Part B*, vol.28, no.1, pp.15-23, 1998.

[13] C. H. Li and P. K. S. Tam, An iterative algorithm for minimum cross entropy thresholding, *Pattern Recogn. Lett.*, vol.19, no.8, pp.771-776, 1998.

[14] P. S. Liao, T. S. Chen and P. C. Chung, A fast algorithm for multi-level thresholding, *J. Inf. Sci. Eng.*, vol.17, no.5, pp.713-727, 2001.

[15] L. Dong, G. Yu, P. Ogunbona and W. Li, An efficient iterative algorithm for image thresholding, *Pattern Recogn. Lett.*, vol.29, no.9, pp.1311-1316, 2008.

[16] D. Y. Huang and C. H. Wang, Optimal multi-level thresholding using a two-stage Otsu optimization approach, *Pattern Recognit. Lett.*, vol.30, no.3, pp.275-284, 2009.

[17] Y. Liu, B. Zhang, L. Zhang, D. Li, Z. Cai and L. Lei, Research on image segmentation based on fuzzy theory, *Proc. of the 2009 WRI World Congress on Computer Science and Information Engineering*, Los Angeles, CA, vol.4, pp.790-794, 2009.

[18] L. Grady and E. L. Schwartz, Isoperimetric graph partitioning for image segmentation, *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol.28, no.3, pp.469-475, 2006.

[19] Z. Li, C. Liu and Y. Cheng, A new image thresholding method based on isoperimetric ratios, *Proc. of IEEE International Conference on Cybernetics and Intelligent Systems*, Chengdu, China, vol.2, pp.724-727, 2008.

[20] P. K. Sahoo, S. Soltani, A. K. C. Wong and Y. Chen, A survey of thresholding techniques, *Computer Vision, Graphics, and Image Processing*, vol.41, no.2, pp.233-260, 1988.

[21] B. Sankur and M. Sezgin, Survey over image thresholding techniques and quantitative performance evaluation, *Journal of Electronic Imaging*, vol.13, no.1, pp.146-165, 2004.

[22] *The Berkeley Segmentation Dataset and Benchmark*, http://www.eecs.berkeley.edu/Research/Projec-ts/CS/vision/bsds.