An Equivalent 3D Otsu's Thresholding Method

Puthipong Sthitpattanapongsa and Thitiwan Srinark*

Graphics Innovation and Vision Engineering (GIVE) Laboratory
Department of Computer Engineering, Faculty of Engineering
Kasetsart University, Bangkok, Thailand
puthi.sthit@gmail.com, thitiwan.s@ku.ac.th

Abstract. Due to unsatisfactory segmentation results when images contain noise by the Otsu's thresholding method. Two-dimensional (2D) and three-dimensional (3D) Otsu's methods thus were proposed. These methods utilize not only grey levels of pixels but also their spatial informations such as mean and median values. The 3D Otsu's methods use both kinds of spatial information while 2D Otsu's methods use only one. Consequently the 3D Otsu's methods more resist to noise, but also require more computational time than the 2D ones. We thus propose a method to reduce computational time and still provide satisfactory results. Unlike the 3D Otsu's methods, our method selects each threshold component in the threshold vector independently instead of one threshold vector. The experimental results show that our method is more robust against noise, and its computational time is very close to that of the 2D Otsu's methods.

Keywords: Image segmentation, Thresholding, 3D Otsu's method, Three-dimensional histogram.

1 Introduction

Thresholding is considered as one low-level segmentation method since it uses only pixel information. The method is typically simple and computationally efficient. Different thresholding methods are described and compared based on different error measurements in [1]. One popular thresholding method is Otsu's [2] due to its fast computation and reasonable results in many applications. However, it uses only a one-dimensional (1D) histogram of an image, which cannot express spatial relation between image pixels, it is difficult to obtain accurate results when images contain noise. Lui *et al.* [3] thus proposed two-dimensional (2D) Otsu's method. This method selects an optimal threshold vector on a 2D histogram. The 2D histogram consists of the gray levels of the image pixels and the mean values of their neighborhood. Since the 2D histogram represents the relation of the original and mean-filtered images, this method gives more satisfactory results. However this method uses an exhaustive search to find the optimal threshold vector, the time complexity of this method is $O(L^4)$, where L

* Thanks to Kasetsart University Research and Development Institute for funding.

Y.-S. Ho (Ed.): PSIVT 2011, Part I, LNCS 7087, pp. 358–369, 2011.

is the number of gray levels. Gong *et al.*[4] thus proposed a fast recursive method of the 2D Otsu's method which can reduce the time complexity from $O(L^4)$ to $O(L^2)$. Ningbo *et al.* [5] proposed a method, which projects a 2D histogram onto a diagonal line to compose a new 1D histogram. The method uses a 1D Otsu's method to select a point that splits this histogram into object and background regions, and applies a 2D Otsu's method to select an optimal threshold vector. This method can enhance execution time, but it requires a large space for three look-up tables. Yue *et al.* [6] proposed a decomposition of the 2D Otsu's method that calculates the optimal threshold by using two 1D Otsu's computations instead of one 2D Otsu's computation. This method is robust against noise, and the time complexity is reduced from $O(L^2)$ to $O(L)$. Chen *et al.* [7] pointed out the weakness of region division by a threshold vector in the 2D Otsu's method that some object and background regions are assigned to edge and noise regions, and vice versa. They proposed the 2D Otsu's method on a gray level-gradient histogram, however, an appropriate initialization is required.

In addition to 2D Otsu's methods, Jing *et al.* [8] proposed a three-dimensional (3D) Otsu's method that selects an optimal threshold vector on a 3D histogram. This 3D histogram contains the median values of neighborhood pixels as the third feature. The 3D Otsu's method provides better results than the 2D Otsu's methods, but its time complexity is $O(L^3)$. Wang *et al.* [9] proposed a group of new recurrence formula of the 3D Otsu's method. This method thus removes redundant computation and calculates a look-up table by iteration. The method has the same thesholding results as the traditional 3D Otsu's method, however, its time complexity is still $O(L^3)$. Dongju *et al.* [10] proved that the objective function of K-means is equivalent to that of the Otsu's method, K-means thus can be extended to 2D and 3D thesholding methods. and performs more efficiently than Otsu's.

Notice that the time complexity of the 2D's Otsu methods can be reduced from $O(L^4)$ to $O(L)$ while the time complexity of the 3D Otsu's methods is still at $O(L^3)$. Even though K-means can be used instead of Otsu's methods, its execution time depends on the number of iterations. In this paper, we propose a fast and robust thresholding method, which selects and uses three optimal thresholds independently instead of one threshold vector of 3D's Otsu methods. Our method can reduce the time complexity from $O(L^3)$ to $O(L)$, and it still provides satisfactory results in noisy conditions.

2 3D Otsu's Method

Given an image $f(x,y)$ represented by L gray levels and the number of pixels in the image, N. The mean and the median of gray values of pixels in the $k \times k$ neighborhood regions centered at the coordinate (x,y) are denoted as $g(x,y)$ and $h(x,y)$, respectively, which are defined as

$$g(x,y) = \frac{1}{k^2} \sum_{i=-k/2}^{k/2} \sum_{j=-k/2}^{k/2} f(x+i, y+j) \tag{1}$$

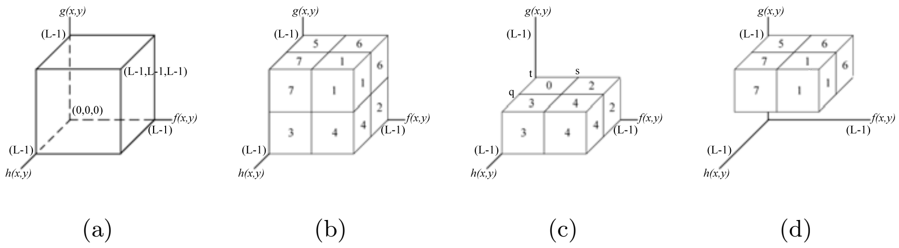


Fig. 1. Three-dimensional histogram

$$h(x,y) = med\left\{f(x+i,y+j) : i = -\frac{k}{2},\dots,\frac{k}{2};\ j = -\frac{k}{2},\dots,\frac{k}{2}\right\} \tag{2}$$

In this paper, we use $k = 3$. For each pixel in the image, we can obtain a triple (i, j, k), where i is the original gray level appeared in $f(x, y)$, j is the grey level of the mean value appeared in $g(x, y)$, and k is the gray level of the median value appeared in $h(x, y)$. All the triples of the image define a 3D histogram within a cube of $L \times L \times L$ as shown in Fig.1(a). Let c_{ijk} denote the frequency of a triple (i, j, k). Its joint probability can be expressed as

$$p_{ijk} = \frac{c_{ijk}}{N}, \tag{3}$$

where $0 \leq i, j, k \leq L-1$ and $\sum_i^{L-1}\sum_j^{L-1}\sum_k^{L-1} p_{ijk} = 1$

Given an arbitrary threshold vector (s, t, q). This threshold vector divides the 3D histogram into eight rectangular volumes as shown in Fig. 1(b)-1(d). Let C_0 and C_1 represent the object and the background, respectively, or vice versa; m_x, ω_x, and μ_x represent the summation vector, the probability, and the mean vector of the rectangular volume x (R_x), respectively, where x is the rectangular volume number; and μ_T represent the total mean vector. m_x can be expressed as

$$\begin{aligned} m_x &= \omega_x \mu_x = (m_{xi}, m_{xj}, m_{xk})^T \\ &= \left(\sum_{(i,j,k)\in R_x} i p_{ijk}, \sum_{(i,j,k)\in R_x} j p_{ijk}, \sum_{(i,j,k)\in R_x} k p_{ijk}\right)^T \end{aligned} \tag{4}$$

The three elements in the triple are very close to each other for the interior pixels of either the object or the background regions while they are very different for the pixels that are edges and noise. Therefore, the rectangular volumes 2-7 can be considered as noise and edges; and rectangular volumes 0 and 1 can be considered as object and background regions, respectively, or vice versa. In most cases, the edge and noise pixels are very small fraction of the overall pixels in an image, hence the probabilities of the rectangular volumes 2-7 can be negligible. It can easily verify the relations,

$$\omega_0 + \omega_1 \approx 1 \qquad \omega_0\mu_0 + \omega_1\mu_1 \approx \mu_T \tag{5}$$

The probabilities of C_0 and C_1 thus can be denoted as

$$\omega_0 = \sum_{(i,j,k)\in R_0} p_{ijk} = \sum_{i=0}^{s}\sum_{j=0}^{t}\sum_{k=0}^{q} p_{ijk} \tag{6}$$

$$\omega_1 = \sum_{(i,j,k)\in R_1} p_{ijk} = \sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}\sum_{k=q+1}^{L-1} p_{ijk} \tag{7}$$

The mean vectors of C_0 and C_1 can be expressed as

$$\begin{aligned}\mu_0 &= (\mu_{0i}, \mu_{0j}, \mu_{0k})^T = \left(\frac{m_{0i}}{\omega_0}, \frac{m_{0j}}{\omega_0}, \frac{m_{0k}}{\omega_0}\right)^T \\ &= \left(\sum_{(i,j,k)\in R_0} \frac{ip_{ijk}}{\omega_0}, \sum_{(i,j,k)\in R_0} \frac{jp_{ijk}}{\omega_0}, \sum_{(i,j,k)\in R_0} \frac{kp_{ijk}}{\omega_0}\right)^T \end{aligned} \tag{8}$$

$$\begin{aligned}\mu_1 &= (\mu_{1i}, \mu_{1j}, \mu_{1k})^T = \left(\frac{m_{1i}}{\omega_1}, \frac{m_{1j}}{\omega_1}, \frac{m_{1k}}{\omega_1}\right)^T \\ &= \left(\sum_{(i,j,k)\in R_1} \frac{ip_{ijk}}{\omega_1}, \sum_{(i,j,k)\in R_1} \frac{jp_{ijk}}{\omega_1}, \sum_{(i,j,k)\in R_1} \frac{kp_{ijk}}{\omega_1}\right)^T \end{aligned} \tag{9}$$

The total mean vector of 3D histogram is

$$\begin{aligned}\mu_T &= (\mu_{iT}, \mu_{jT}, \mu_{kT})^T \\ &= \left(\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} ip_{ijk}, \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} jp_{ijk}, \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} kp_{ijk}\right)^T \end{aligned} \tag{10}$$

The between-class discrete matrix is defined as

$$S_B(s,t,q) = \omega_0[(\mu_0 - \mu_T)(\mu_0 - \mu_T)^T] + \omega_1[(\mu_1 - \mu_T)(\mu_1 - \mu_T)^T] \tag{11}$$

The trace of discrete matrix can be expressed as

$$\begin{aligned} tr(S_B(s,t,q)) = \omega_0[(\mu_{0i} - \mu_{Ti})^2 + (\mu_{0j} - \mu_{Tj})^2 + (\mu_{0k} - \mu_{Tk})^2] + \\ \omega_1[(\mu_{1i} - \mu_{Ti})^2 + (\mu_{1j} - \mu_{Tj})^2 + (\mu_{1k} - \mu_{Tk})^2] \end{aligned} \tag{12}$$

The optimal threshold vector (s', t', q') is

$$(s', t', q') = arg \max_{0 \leq s,t,q \leq L-1} (tr(S_B(s,t,q))) \tag{13}$$

3 Proposed Method

From (5), we can see that $\omega_x \approx 0$ and $\omega_x\mu_x = m_x \approx 0$, where $x = 2, \ldots, 7$. From these conditions, we can conclude as follows.

$$\omega_0 \approx \omega_{0i} = \omega_0 + \omega_3 + \omega_5 + \omega_7 = \sum_{i=0}^{s}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} p_{ijk} = \sum_{i=0}^{s} P_i \tag{14}$$

$$\omega_1 \approx \omega_{1i} = \omega_1 + \omega_2 + \omega_4 + \omega_6 = \sum_{i=s+1}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} p_{ijk} = \sum_{i=s+1}^{L-1} P_i \tag{15}$$

$$\omega_0 \approx \omega_{0j} = \omega_0 + \omega_2 + \omega_3 + \omega_4 = \sum_{i=0}^{L-1}\sum_{j=0}^{t}\sum_{k=0}^{L-1} p_{ijk} = \sum_{j=0}^{t} P_j \tag{16}$$

$$\omega_1 \approx \omega_{1j} = \omega_1 + \omega_5 + \omega_6 + \omega_7 = \sum_{i=0}^{L-1}\sum_{j=t+1}^{L-1}\sum_{k=0}^{L-1} p_{ijk} = \sum_{j=t+1}^{L-1} P_j \tag{17}$$

$$\omega_0 \approx \omega_{0k} = \omega_0 + \omega_2 + \omega_5 + \omega_6 = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{q} p_{ijk} = \sum_{k=0}^{q} P_k \tag{18}$$

$$\omega_1 \approx \omega_{1k} = \omega_1 + \omega_3 + \omega_4 + \omega_7 = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=q+1}^{L-1} p_{ijk} = \sum_{k=q+1}^{L-1} P_k \tag{19}$$

$$m'_{0i} = m_{0i} + m_{3i} + m_{5i} + m_{7i} = \sum_{i=0}^{s}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} ip_{ijk} = \sum_{i=0}^{s} iP_i \tag{20}$$

$$m'_{1i} = m_{1i} + m_{2i} + m_{4i} + m_{6i} = \sum_{i=s+1}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} ip_{ijk} = \sum_{i=s+1}^{L-1} iP_i \tag{21}$$

$$m'_{0j} = m_{0j} + m_{2j} + m_{3j} + m_{4j} = \sum_{i=0}^{L-1}\sum_{j=0}^{t}\sum_{k=0}^{L-1} jp_{ijk} = \sum_{j=0}^{t} jP_j \tag{22}$$

$$m'_{1j} = m_{1j} + m_{5j} + m_{6j} + m_{7j} = \sum_{i=0}^{L-1}\sum_{j=t+1}^{L-1}\sum_{k=0}^{L-1} jp_{ijk} = \sum_{j=t+1}^{L-1} jP_j \tag{23}$$

$$m'_{0k} = m_{0k} + m_{2k} + m_{5k} + m_{6k} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{q} kp_{ijk} = \sum_{k=0}^{q} kP_k \tag{24}$$

$$m'_{1k} = m_{1k} + m_{3k} + m_{4k} + m_{7k} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=q+1}^{L-1} kp_{ijk} = \sum_{k=q+1}^{L-1} kP_k \tag{25}$$

where

$$m_{0i} \approx m'_{0i}, m_{1i} \approx m'_{1i}, m_{0j} \approx m'_{0j}, m_{1j} \approx m'_{1j}, m_{0k} \approx m'_{0k}, m_{1k} \approx m'_{1k}.$$

Thus, we can define the new mean vectors as

$$\begin{aligned}\mu_0 \approx \mu'_0 = (\mu'_{0i}, \mu'_{0j}, \mu'_{0k})^T &= \left(\frac{m'_{0i}}{w_{0i}}, \frac{m'_{0j}}{w_{0j}}, \frac{m'_{0k}}{w_{0k}}\right)^T \\ &= \left(\frac{\sum_{i=0}^{s} iP_i}{\sum_{i=0}^{s} P_i}, \frac{\sum_{j=0}^{t} jP_j}{\sum_{j=0}^{t} P_j}, \frac{\sum_{k=0}^{q} kP_k}{\sum_{k=0}^{q} P_k}\right)^T \end{aligned} \tag{26}$$

$$\begin{aligned}\mu_1 \approx \mu'_1 = (\mu'_{1i}, \mu'_{1j}, \mu'_{1k})^T &= \left(\frac{m'_{1i}}{w_{1i}}, \frac{m'_{1j}}{w_{1j}}, \frac{m'_{1k}}{w_{1k}}\right)^T \\ &= \left(\frac{\sum_{i=s+1}^{L-1} iP_i}{\sum_{i=s+1}^{L-1} P_i}, \frac{\sum_{j=t+1}^{L-1} jP_j}{\sum_{j=t+1}^{L-1} P_j}, \frac{\sum_{k=q+1}^{L-1} kP_k}{\sum_{k=q+1}^{L-1} P_k}\right)^T \end{aligned} \tag{27}$$

where $P_i = \sum_{j=0}^{L-1} \sum_{k=0}^{L-1} p_{ijk}$, $P_j = \sum_{i=0}^{L-1} \sum_{k=0}^{L-1} p_{ijk}$, and $P_k = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} p_{ijk}$. Notice that P_i, P_j, and P_k are equivalent with the 1D histogram of of original, mean-filtered, and median-filtered images, respectively. From (14)-(19) and (26)-(27), we can rewritten (12) as

$$\begin{aligned} tr(S_B(s,t,q)) \approx\ & \overbrace{[\omega_{0i}(\mu'_{0i} - \mu_{Ti})^2 + \omega_{1i}(\mu'_{1i} - \mu_{Ti})^2]}^{A} + \\ & \overbrace{[\omega_{0j}(\mu'_{0j} - \mu_{Tj})^2 + \omega_{1j}(\mu'_{1j} - \mu_{Tj})^2]}^{B} + \\ & \overbrace{[\omega_{0k}(\mu'_{0k} - \mu_{Tk})^2 + \omega_{1k}(\mu'_{1k} - \mu_{Tk})^2]}^{C} \end{aligned} \tag{28}$$

The values of terms A, B, and C depend on the values of s, t, and q, respectively. We can define each term as

$$\sigma_{Bi}(s) = \omega_{0i}(\mu'_{0i} - \mu_{Ti})^2 + \omega_{1i}(\mu'_{1i} - \mu_{Ti})^2 \tag{29}$$

$$\sigma_{Bj}(t) = \omega_{0j}(\mu'_{0j} - \mu_{Tj})^2 + \omega_{1j}(\mu'_{1j} - \mu_{Tj})^2 \tag{30}$$

$$\sigma_{Bk}(q) = \omega_{0k}(\mu'_{0k} - \mu_{Tk})^2 + \omega_{1k}(\mu'_{1k} - \mu_{Tk})^2 \tag{31}$$

The optimal threshold (s', t', q') is

$$\begin{aligned}(s', t', q') &= arg \max_{0 \le s,t,q \le L-1} (tr(S_B(s,t,q))) \\ &\approx arg \max_{0 \le s,t,q \le L-1} (\sigma_{Bi}(s) + \sigma_{Bj}(t) + \sigma_{Bk}(q)) \end{aligned} \tag{32}$$

which can be splited into

$$s' = arg \max_{0 \le s \le L-1} \sigma_{Bi}(s) \tag{33}$$

$$t' = arg \max_{0 \le t \le L-1} \sigma_{Bj}(t) \tag{34}$$

$$q' = arg \max_{0 \le q \le L-1} \sigma_{Bk}(q) \tag{35}$$

Equations (33), (34), and (35) are 1D Otsu's methods that select the optimal threshold of the original, mean-filtered, and median-filtered images, respectively. Notice that we select the optimal threshold from three 1D histograms instead of one 3D histogram. Therefore, the time complexity of this method is only $O(L)$ instead of $O(L^3)$. We then apply each threshold element as a classifier to classify each image pixel into either the object or the background independently. A pixel (x, y) is assigned to the class, which is mostly selected by the thresholds s', t', and q' in the original, mean-filtered, and median-filtered images, respectively.

4 Experimental Results

We performed all experiments on a personal computer with 2.0 GHz Intel(R) Core(TM)2 Duo CPU and 4 GB DDR II memory. We implemented the proposed method in Visual C++ with OpenCV. Scilab was used to generate noised added images for noise tolerant tests. We tested on two kinds of noise including Salt&Pepper noise and Gaussian noise. Salt&Pepper noise is represented by noise density (δ), the probability of swapping a pixel. Gaussian noise is represented by mean (μ) and variance (σ^2). In our experiments, we used only $\mu = 0$.

We compared our method with the 1D Otsu's method [2], Gong's method [4] as the 2D Otsu's method, Wang's method [9] as the 3D Otsu's method, K-means [10] based methods for both 2D and 3D ones, Ningbo's method [5], and Yue's method [6] because they are based on Otsu's. For each experiment that the ground truth is available, we use misclassification error (ME) to present the number of background pixels wrongly assigned to the foreground, and vice versa; and we use modified Hausdorff distance (MHD) to measure the shape distortion of each result image compared with its corresponding ground truth. ME and MHD are defined as [1]

$$\mathsf{ME} = 1 - \frac{|B_O \cap B_T| + |F_O \cap F_T|}{|B_O| + |F_O|}, \tag{36}$$

$$\mathsf{MHD} = max(d_{\mathsf{MHD}}(F_O, F_T), d_{\mathsf{MHD}}(F_T, F_O)), \tag{37}$$

where

$$d_{\mathsf{MHD}}(F_O, F_T) = \frac{1}{|F_O|} \sum_{f_O \in F_O} \min_{f_T \in F_T} \|f_O - f_T\|,$$

$$d_{\mathsf{MHD}}(F_T, F_O) = \frac{1}{|F_T|} \sum_{f_T \in F_T} \min_{f_O \in F_O} \|f_T - f_O\|.$$

F_i and B_i denote the foreground and background pixels, respectively, of an image i, which includes the ground truth (O) and thresholded (T) images. $|.|$ is the cardinality of the set. $\|f_O - f_T\|$ is the Euclidean distance between the two corresponding pixels of the ground truth and thresholded images. Notice that ME varies from 0 (a perfectly classified image) to 1 (a totally incorrect binarized image).

(a) Original (b) Gaussian (c) Salt&Pepper

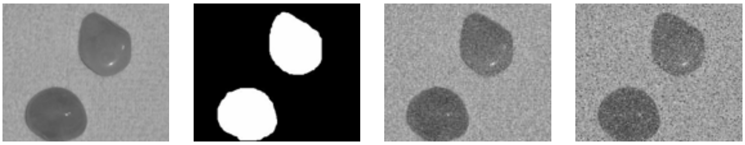
Fig. 2. Lena images w/o noise added

Table 1. Optimal thresholds of Lena images w/o noise added

Methods	Fig.		
	2(a)	2(b)	2(c)
1D Otsu's	117	119	117
2D Otsu's	(123,117)	(123,126)	(117,192)
3D Otsu's	(130,125,117)	(132,122,121)	(130,126,117)
2D K-means	(117,117)	(118,118)	(117,117)
3D K-means	(117,117,117)	(118,118,118)	(117,117,117)
Ningbo's	(117,117)	(117,119)	(117,117)
Yue's	(117,117)	(119,118)	(117,117)
Proposed	(117,117,117)	(117,117,117)	(117,117,117)

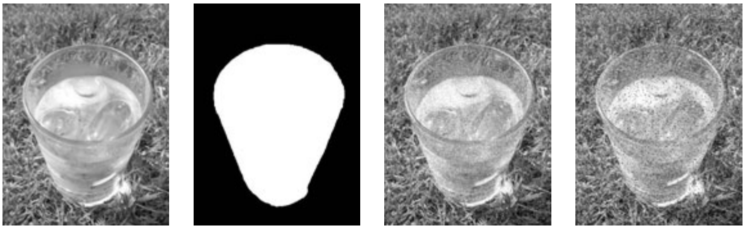
In the first experiment, we compared the optimal threshold selected by each method. We segmented Lena images consisting of the original one, and two noise added images. The first noise added image was generated by adding Salt&Pepper noise with $\delta = 0.01$ to the original image, and the other one was generated by adding Gaussian noise with $\sigma^2 = 0.005$ to the original image as shown in Fig. 2. Optimal thresholds are shown in Table 1. It can be seen that the optimal threshold of the proposed method is close to the optimal threshold of the other methods.

In the second experiment, we tested the robustness of each method in the presence of noise. We selected two images as our test images from *Segmentation evaluation database*[11]. Fig. 3(a) and 3(b) show the first test image and its ground truth, respectively. Fig. 4(a) and 4(b) show the second test image and its ground truth, respectively. We added noise to each test image to generate new 51 images with Salt&Pepper noise using δ that are vary from 0 to 0.1, and the other 51 images with Gaussian noise using σ^2 that are vary from 0 to 0.01. Fig. 3(c) and 3(d) show example noise added images of the first test image. Fig. 4(c) and 4(d) show example noise added images of the second test image. Both test images show difficulties for thresholding when some amount of noise is added. Fig. 5(a) shows the histogram of the first test image that clearly presents bimodal, while Fig. 5(c) shows the histogram of the second test image that does not clearly presents bimodal. Fig. 5(b) and 5(d) show histograms of two images with Gaussian noise added. Both of them present a single modal with a jagged curve. The second test image itself can be challenged to segment such that some background pixels present similar gray levels as the object. We



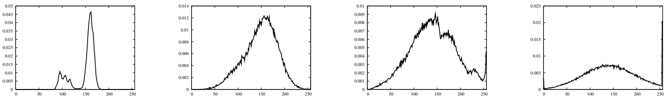
(a) Original (b) Ground truth (c) Gaussian ($\sigma^2 = 0.01$) (d) Salt&Pepper ($\delta = 0.1$)

Fig. 3. The first image set with sample noise added images in the second experiment



(a) Original (b) Ground truth (c) Gaussian ($\sigma^2 = 0.01$) (d) Salt&Pepper ($\delta = 0.1$)

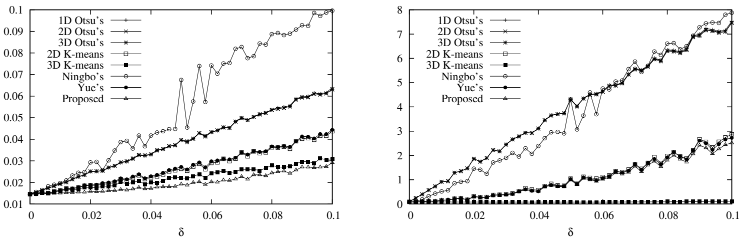
Fig. 4. The second image set with sample noise added images in the second experiment



(a) Histogram of Fig.3(a) (b) Histogram of Fig.3(c) (c) Histogram of Fig.4(a) (d) Histogram of Fig.4(c)

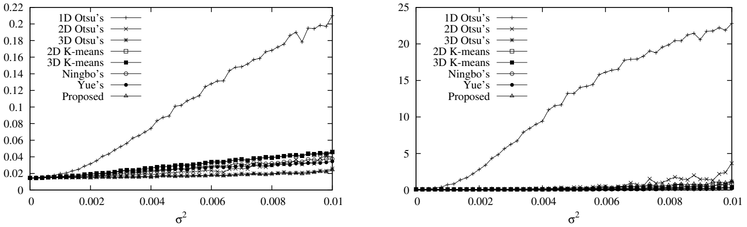
Fig. 5. Histograms of the test images in the second experiment

segmented these 204 noise added images. We evaluated the performance of each method based on ME and MHD. Fig. 6 and 7 show the evaluation results of the first test images. Fig. 8 and 9 show the evaluation results of the second test images. From the evaluation results in the presence of Salt&Pepper noise shown in Fig. 6 and 8, both ME and MHD values of our method are lower than those of the other methods except MHD values on the first test images, MHD values of our method are higher than the 3D K-means method. The thresholding results of the 3D K-means and our methods are shown in Fig. 10. The 3D K-means method gives higher number of mistaken pixels in the object region, and lower number of mistaken pixels in the background region, however, our method gives lower number of mistaken pixels in the object region, and higher number of pixels in the background region. MHD of our method is thus higher than of the 3D K-means. From the evaluation results in the presence of Gaussian noise shown in Fig. 7 and 9, both ME and MHD values of our method on the second test images are lower than those of the other methods. ME values of our method on the first test images are very close to that of the 3D Otsu's method and lower than those of the other methods. MHD values of our method on the first test images



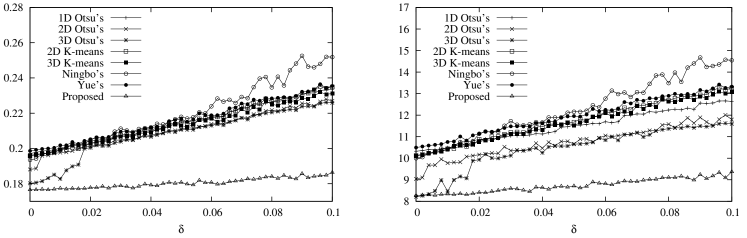
(a) ME (b) MHD

Fig. 6. Comparison of ME and MHD for thresholding of the first test images with Salt&Pepper noise added at various δ in the second experiment



(a) ME (b) MHD

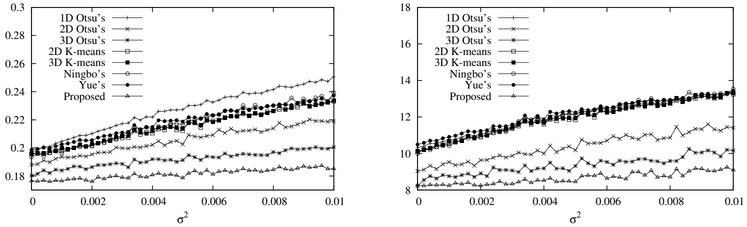
Fig. 7. Comparison of ME and MHD for thresholding of the first test images with Gaussian noise added at various σ^2 in the second experiment



(a) ME (b) MHD

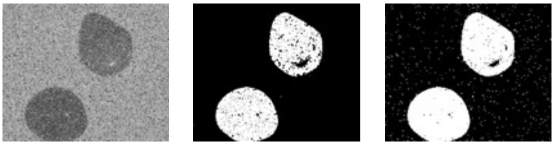
Fig. 8. Comparison of ME and MHD for thresholding of the second test images with Salt&Pepper noise added at various δ in the second experiment

are a little higher than those of the other methods except the 1D Otsu's and 2D Otsu's methods. The average computational time on all noise added images are 0.08, 12.05, 1891.63, 27.21, 1259.54, 11.39, 6.29, and 13.19 ms, for the 1D Otsu's, 2D Otsu's, 3D Otsu's, 2D K-means, 3D K-means, Ningbo's, Yue's, and our proposed methods, respectively. It can be seen that our method performs faster than the other 3D methods. Our average execution time is nearly the same as that of the other 2D methods except Yue's method. Our method always gives low error measurements in both classification and shape evaluations.



(a) ME (b) MHD

Fig. 9. Comparison of ME and MHD for thresholding of the second test images with Gaussian noise added at various σ^2 in the second experiment



(a) Original (b) 3D K-means (c) proposed

Fig. 10. The original image and thresholded images of 3D K-means and proposed methods when σ^2=0.01

In the last experiment, we tested our method and the others with 200 real images from the *Segmentation evaluation database* [11], where the ground truth of each image is provided. The average error measurements ($\overline{\mathsf{ME}}$ and $\overline{\mathsf{MHD}}$) and the average compuational time ($\overline{\mathsf{T}}$) over 200 test images of each method are shown in Table 2. Segmentation results can be seen at `http://give.cpe.ku.ac.th/thresholding/equivalent-3D-thresholding.php`. From the results, it can be seen that the average computational time of our method is lower than that of the other 3D methods and is almost the same as that of the other 2D methods except the Yue's method. The average ME and MHD values of our method is lower than that of the other methods. It indicates that our method shows the best matching of the object and the background, and also gives the smallest amount of shape distortion.

Table 2. $\overline{\mathsf{ME}}$, $\overline{\mathsf{MHD}}$, and $\overline{\mathsf{T}}$ over 200 real images

Method	$\overline{\mathsf{ME}}$	$\overline{\mathsf{MHD}}$	$\overline{\mathsf{T}}$ (ms)
1D Otsu's	0.217102	19.545244	0.32
2D Otsu's	0.214391	19.580417	11.99
3D Otsu's	0.213022	19.569091	2151.03
2D K-means	0.228411	19.603090	20.60
3D K-means	0.228340	19.616353	1027.13
Ningbo's	0.214194	19.627783	12.92
Yue's	0.214962	19.408089	6.66
Proposed	0.211341	19.199144	12.59

5 Conclusions

We presented an improved thresholding method to overcome the shortcoming of the 1D, 2D, and 3D Otsu's method. The method calculates each optimal threshold from the original, mean-filtered, and median-filtered images independently; and uses the most selected class by each threshold on the corresponding images as the thresholding results. We tested our method on real images and images with noise added. The results show that our method gives satisfactory results, and it is robust against noise. Moreover, it requires less computational time than the other 3D methods, and also gives better or comparable results.

References

1. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. Jour. of Electronic Imaging 13(1), 146–168 (2004)
2. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. on Systems, Man and Cybernetics 9(1), 62–66 (1979)
3. Liu, J., Li, W., Tian, Y.: Automatic thresholding of gray-level pictures using two-dimension otsu method. In: Proc. of Intl. Conf. on Circuits and Systems, China, vol. 1, pp. 325–327 (1991)
4. Gong, J., Li, L., Chen, W.: Fast recursive algorithms for two-dimensional thresholding. Pattern Recognition 31(3), 295–300 (1998)
5. Ningbo, Z., Gang, W., Gaobo, Y., Weiming, D.: A fast 2d otsu thresholding algorithm based on improved histogram. In: Chinese Conf. on Pattern Recognition (CCPR), pp. 1–5 (2009)
6. Yue, F., Zuo, W.M., Wang, K.Q.: Decomposition based two-dimensional threshold algorithm for gray images. Zidonghua Xuebao/Acta Automatica Sinica 35(7), 1022–1027 (2009)
7. Chen, Y., Chen, D.r., Li, Y., Chen, L.: Otsu's thresholding method based on gray level-gradient two-dimensional histogram. In: 2nd Intl. Asia Conf. on Informatics in Control, Automation and Robotics (CAR), vol. 3, pp. 282–285 (2010)
8. Jing, X.J., Li, J.F., Liu, Y.L.: Image segmentation based on 3-d maximum between-cluster variance. Tien Tzu Hsueh Pao/Acta Electronica Sinica 31(9), 1281–1285 (2003)
9. Wang, L., Duan, H., Wang, J.: A fast algorithm for three-dimensional otsu's thresholding method. In: IEEE Intl. Sym. on IT in Medicine and Education (ITME), pp. 136–140 (2008)
10. Dongju, L., Jian, Y.: Otsu method and k-means. In: Ninth Intl. Conf. on Hybrid Intelligent Systems (HIS), vol. 1, pp. 344–349 (2009)
11. Alpert, S., Galun, M., Basri, R., Brandt, A.: Image segmentation by probabilistic bottom-up aggregation and cue integration. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1–8 (2007)