THESIS APPROVAL

GRADUATE SCHOOL, KASETSART UNIVERSITY

Master of Engineering (Computer Engineering)

DEGREE

Computer Engineering | Computer Engineering

FIELD | **DEPARTMENT**

TITLE: Multilevel and Multi-dimensional Thresholding

NAME: Mr. Puthipong Sthitpattanapongsa

THIS THESIS HAS BEEN ACCEPTED BY

_____ **THESIS ADVISOR**

(Assistant Professor Thitiwan Srinark, Ph.D.)

_____ **THESIS CO-ADVISOR**

(Associate Professor Punpiti Piamsa-nga, D.Sc.)

_____ **DEPARTMENT HEAD**

(Assistant Professor Putchong Uthayopas, Ph.D.)

APPROVED BY THE GRADUATE SCHOOL ON _____

_____ **DEAN**

(Associate Professor Gunjana Theeragool, D.Agr.)

THESIS

MULTILEVEL AND MULTI-DIMENSIONAL THRESHOLDING

PUTHIPONG STHITPATTANAPONGSA

A Thesis Submitted in Partial Fulfillment of
the Requirements for the Degree of
Master of Engineering (Computer Engineering)
Graduate School, Kasetsart University
2012

Puthipong Sthitpattanapongsa 2012: Multilevel and Multi-dimensional Thresholding. Master of Engineering (Computer Engineering), Major Field: Computer Engineering, Department of Computer Engineering. Thesis Advisor: Assistant Professor Thitiwan Srinark, Ph.D. 79 pages.

Thresholding is a low-level image segmentation method. Otsu's method is one of the most popular thresholding method. In order to increase its efficiency, Otsu's method is thus extended to use a multi-dimensional histogram to handle images that contain noise, and it is extended to support multiple levels to analyze complex images. These two extended methods have different properties. However, real world images mostly contain noise and are complicated. In this thesis, we combine these two methods into a single one so that the new proposed method can take advantages from both methods, for thresholding real world images.

However, there are some drawbacks in multi-dimensional methods as follows. The region division of the multi-dimensional histogram is not reasonable; and ignoring edge and noise regions in calculation can cause unsuccessful segmentation results. So we propose methods to improve the multi-dimensional method to overcome these drawbacks before combining. From the experimental results, they show that our methods including 2D and 3D ones perform faster, more resist to noise, and give better or comparable results than the others. Moreover, our 3D method gives better results and less computational time than our 2D one.

Finally, we combine our proposed 3D method with Eichmann's multilevel thresholding method for the threshold selection. From the experimental results, they show that our proposed multilevel method is more resist to noise and also give better or comparable results than the other methods but it requires longer execution time.

_____ _____ ___ / ___ / _____

Student's signature Thesis Advisor's signature

ACKNOWLEDGEMENTS

I would like to gratefully thank and deeply indebted to Assist. Prof. Dr. Thitiwan Srinark, my thesis advisor, for advices, encouragements, and valuable suggestions to complete the thesis.

I would like to sincerely thank all faculty members in the department of computer engineering for teaching and guidance.

I would like to sincerely thank my parents for consistent encouragements and support.

I would like to thank the officers in the computer engineering graduate school for providing good services.

I would like to thank my friends in the department of computer engineering for encouragements, advices, and assistance.

Puthipong Sthitpattanapongsa

April 2012

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

Figure **Page**

MULTILEVEL AND MULTI-DIMENSIONAL THRESHOLDING

INTRODUCTION

Image segmentation is the technique that extracts objects or regions of interest in an image from the background. The ultimate goal is to simplify the image so that further analysis can be performed. Many computer applications, such as object recognition, scene understanding and analysis, automatic traffic control system, locating objects in satellite images, document analysis, and many applications in medical imagining, are based on segmentation. Many segmentation techniques are available, however, it still remains a challenge for computer vision to satisfy goal needs. Therefore, the selection of proper techniques is very important as per application. Different traditional and common segmentation techniques consist of threshold-based techniques, histogram-based techniques, edge detection techniques, region-based techniques, watershed transformation techniques, and graph partitioning techniques. Threshold-based techniques are generally used for gray scaled images. It is one of the most popular technique as it is very simple to implement. (Raut *et al.*, 2009)

Thresholding is considered as one low-level segmentation method since it uses only pixel information. This kind of methods is typically simple and operationally efficient. Most thresholding methods are bi-level and one-dimensional (1D) thresholding methods. These methods select a threshold from a 1D histogram of gray levels which separates the pixels into two classes. One of the most well-known thresholding method is Otsu's due to its fast computation and reasonable results in various applications

For complex image analysis, some 1D thresholding methods are extended from a bi-level threshold selection to a multilevel threshold selection. These methods select multiple thresholds, which separate pixels into several classes. Moreover, 1D thresholding method uses only a 1D histogram of an image, which cannot express the spatial relation between image pixels. It is difficult to obtain accurate results when images contain noise. It is thus extended to multi-dimensional thresholding methods which select a threshold vector from either two-dimensional (2D) or three-dimensional (3D) histogram. These methods can give better results because of the spatial information in the histogram. However, real world images are complicated and contain noise. There are still no methods that can handle both difficulties.

In this thesis, we propose a new method that combines the multi-dimensional Otsu's thresholding method with a multilevel threshold selection. This method can take advantage of complex image analysis from the multilevel method and advantage of noise resistance from the multi-dimensional method. However, multi-dimensional method has some drawbacks. The drawbacks consist of long execution time, ignoring some regions of the histogram in calculation of threshold, and unreasonable histogram division in the 2D Otsu's method. We thus improve a multi-dimensional Otsu's thresholding method by using histogram projections before extending the method from a bi-level to a multilevel threshold selection.

OBJECTIVES

1. To improve the muti-dimensional Otsu's thresholding methods in order to reduce computational time, increase noise resistance and support for extension of the method to multilevel thresholding

2. To extend the improved multi-dimensional Otsu's thresholding method to the multilevel method

LITERATURE REVIEW

1. Thresholding

Thresholding is one well-known image segmentation methods. The objective of thresholding is to extract objects or regions of interest based on the selected threshold from the image histogram. In general, there are three steps of thresholding method as shown in Fig. 1. The first step is to generate one or more histograms from an input image. The histogram can be either one-dimensional (1D), two-dimensional (2D), or three-dimensional (3D) histogram. The second step is to select one or more thresholds from the histogram by a selection technique. The number of thresholds is based on the number of separated classes. Threshold selection techniques can be classified into parametric and non-parametric approaches (Horng, 2010). In the parametric approach, the gray level distribution of each class has a probability density function that is assumed to obey a Gaussian distribution. The approaches attempt to find an estimation of parameters of the distribution that will best fit the given histogram. For example, Ramírez-Ortegón *et al.* (2010) use the transition set to calculate the mean and variance of each distribution which are used to binarize the image. In non-parametric approach, the thresholds are selected by optimizing (either maximization or minimization) some criterion function defined from images. For example, Otsu (1979) selects an optimal threshold that maximizes the between-class variance. The last step is to classify the image pixels into classes based on the selected thresholds. Different thresholding methods are described and compared based on different error measurements in Sezgin and Sankur (2004).

2. Otsu's thresholding method

Otsu's thresholding method was proposed by Otsu (1979). This method considers the 1D histogram of an image to select an optimal threshold that maximizes the between-class variance. The details of this method is as follows.

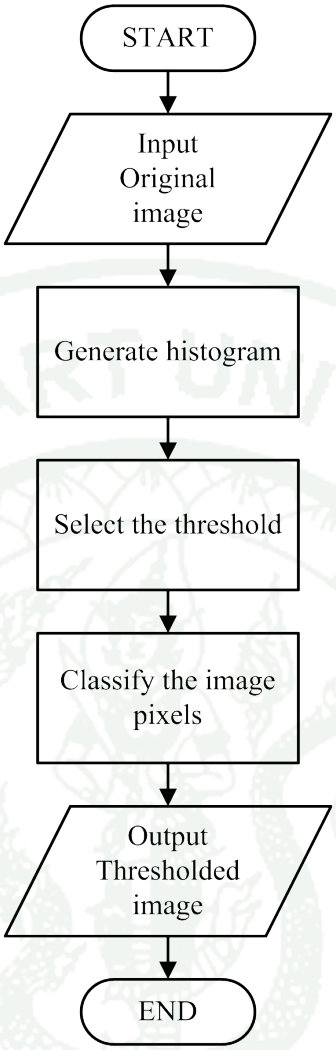


Figure 1 The flowchart of the general thresholding method

Given an image $f(x, y)$ represented by L gray levels. Let N be the number of pixels in the image, c_i denote the frequency of the gray level i and its joint probability can be expressed as

$$p_i = \frac{c_i}{N} \tag{1}$$

where $\sum_{i=0}^{L} p_i = 1$ and $i = 0, 1, 2, \ldots, L-1$

Given an arbitrary threshold t. Let two classes C_0 and C_1 represent the object and the background, or vice versa. ω_x and μ_x represent the probability and mean value of the class C_x, respectively. The probabilities of C_0 and C_1 can be denoted as

$$\omega_0 = \sum_{i=0}^{t} p_i \tag{2}$$

$$\omega_1 = \sum_{i=t+1}^{L-1} p_i \tag{3}$$

The mean values of C_0 and C_1 can be denoted as

$$\mu_0 = \sum_{i=0}^{t} \frac{ip_i}{\omega_0} \tag{4}$$

$$\mu_1 = \sum_{i=t+1}^{L-1} \frac{ip_i}{\omega_1} \tag{5}$$

The total mean value of the histogram is

$$\mu_T = \sum_{i=0}^{L-1} ip_i \tag{6}$$

The between-class variance is defined as

$$\sigma_B^2(t) = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 \tag{7}$$

The optimal threshold is thus defined as

$$t^* = \arg \max_{0 \le t \le L-1} \sigma_B^2(t) \tag{8}$$

The binary image $f_T(x, y)$ can be obtained by

$$f_T(x, y) = \begin{cases} 0 & \text{if } f(x, y) \le t^* \\ 1 & \text{if } f(x, y) > t^* \end{cases} \tag{9}$$

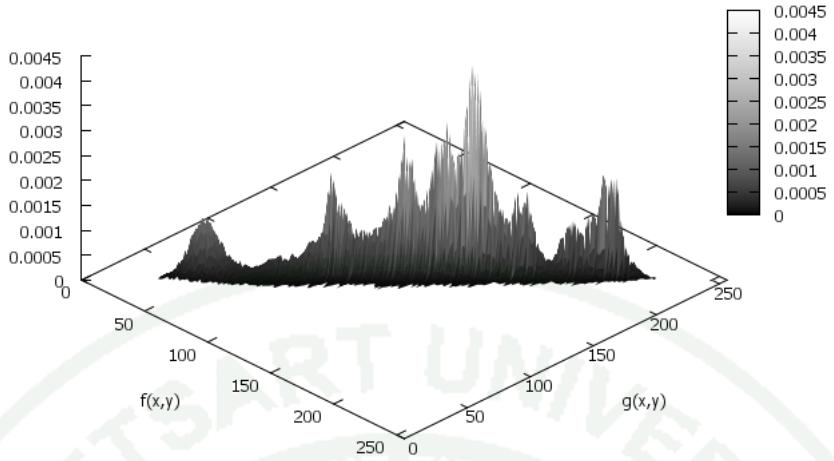


Figure 2 2D histogram

This method is good for real world images with regard to uniformity and shape measure and is one of the most popular thresholding method (Sahoo *et al.*, 1988; Sezgin and Sankur, 2004).

3. Multi-dimensional Otsu's thresholding method

Due to the fact that 1D Otsu's thresholding method considers only the gray levels of pixels, which cannot express the spatial relation between image pixels, the method cannot give satisfactory results when images contain noise. This method thus was extended to multi-dimensional Otsu's thresholding methods. Multi-dimensional Otsu's thresholding methods include two-dimensional (2D) and three-dimensional (3D) Otsu's thresholding methods.

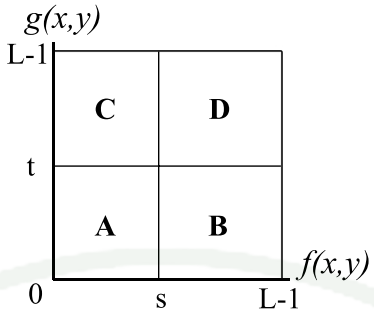


Figure 3 Region division of the 2D histogram

3.1 Two-dimensional (2D) Otsu's thresholding method

Liu *et al.* (1991) proposed a 2D Otsu's thresholding method which selects a threshold vector from the 2D histogram. This 2D histogram as shown in Fig. 2 consists of the gray levels of image pixels ($f(x, y)$) and the mean value of their neighborhood ($g(x, y)$). The detail of this method is explained as follows.

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

$$g(x,y) = \frac{1}{k^2} \sum_{i=-k/2}^{k/2} \sum_{j=-k/2}^{k/2} f(x+i, y+j) \tag{10}$$

where, k=3, in this thesis. For each pixel in the image, we can obtain a pair (i, j), where i is the original gray level appeared in $f(x, y)$, and j is the gray level of the mean value appeared in $g(x, y)$. Let c_{ij} denote the frequency of pair (i, j), and its joint probability can be expressed as

$$p_{ij} = \frac{c_{ij}}{N} \tag{11}$$

where $\sum_{i=0}^{L-1}\sum_{j=0}^{L-1} p_{ij} = 1, i$ and $j = 0, 1, \ldots, L-1$.

Given an arbitrary threshold vector (s, t). This threshold vector divides the 2D histogram into four regions as shown in Fig. 3. The regions A and D represent the object and the background, respectively. The regions B and C represent both edges and noise. Let two classes C_0 and C_1 represent the object and the background, or vice versa; and ω_x and μ_x represent the probability and mean value vector of the class C_x, respectively. The probabilities of C_0 and C_1 can be denoted as

$$\omega_0 = \sum_{i=0}^{s}\sum_{j=0}^{t} p_{ij} \tag{12}$$

$$\omega_1 = \sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1} p_{ij} \tag{13}$$

The mean value vectors of C_0 and C_1 can be expressed as

$$\begin{aligned} \mu_0 &= (\mu_{0i}, \mu_{0j})^T \\ &= \left(\sum_{i=0}^{s}\sum_{j=0}^{t}\frac{ip_{ij}}{\omega_0}, \sum_{i=0}^{s}\sum_{j=0}^{t}\frac{jp_{ij}}{\omega_0}\right)^T \end{aligned} \tag{14}$$

$$\begin{aligned} \mu_1 &= (\mu_{1i}, \mu_{1j})^T \\ &= \left(\sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}\frac{ip_{ij}}{\omega_1}, \sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}\frac{jp_{ij}}{\omega_1}\right)^T \end{aligned} \tag{15}$$

The total mean vector of the 2D histogram is

$$\begin{aligned} \mu_T &= (\mu_{Ti}, \mu_{Tj})^T \\ &= \left(\sum_{i=0}^{L-1}\sum_{j=0}^{L-1} ip_{ij}, \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} jp_{ij}\right)^T \end{aligned} \tag{16}$$

In most cases, the probability that is away from the diagonal can be negligible. It can easily verify the following relations:

$$\omega_0 + \omega_1 \approx 1 \tag{17}$$

$$\omega_0\mu_0 + \omega_1\mu_1 \approx \mu_T \tag{18}$$

The between-class discrete matrix is defined as

$$S_B(s,t) = \omega_0[(\mu_0 - \mu_T)(\mu_0 - \mu_T)^T] + \omega_1[(\mu_1 - \mu_T)(\mu_1 - \mu_T)^T] \tag{19}$$

The trace of discrete matrix could be expressed as

$$\begin{aligned} tr(S_B(s,t)) &= \omega_0[(\mu_{0i} - \mu_{Ti})^2 + (\mu_{0j} - \mu_{Tj})^2] + \\ &\quad \omega_1[(\mu_{1i} - \mu_{Ti})^2 + (\mu_{1j} - \mu_{Tj})^2] \end{aligned} \tag{20}$$

The optimal threshold is thus defined as

$$(s^*, t^*) = \arg \max_{0 \le s,t \le L-1} TrS_b(s,t) \tag{21}$$

The binary image $f_T(x,y)$ can be obtained by

$$f_T(x,y) = \begin{cases} 0 & \text{if } f(x,y) \le s^* \text{ and } g(x,y) \le t^* \\ 1 & \text{if } f(x,y) > s^* \text{ and } g(x,y) > t^* \end{cases} \tag{22}$$

Note that pixels that satisfy conditions $f(x,y) \le s^*$ and $g(x,y) > t^*$; and $f(x,y) > s^*$ and $g(x,y) \le t^*$ are ignored and are set to either 0 or 1. In this thesis, we set them to 1.

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 $O(L^4)$. Gong *et al.* (1998) thus proposed a fast recursive method of the 2D Otsu's method which can reduce the time complexity from $O(L^4)$ to $O(L^2)$. Notice that this method gives the same thresholding results because of changing only the search technique. Yue *et al.* (2009) 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 $O(L)$.

Ningbo *et al.* (2009) 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 on the line, which is vertical to the diagonal and pass the selected point, to select an optimal threshold vector. This method can enhance execution time, but it requires a large space for three look-up tables. Qidan *et al.* (2010) used the average of two thresholds, which are selected by Yue's method, to form a threshold vector for image thresholding. Changhua *et al.* (2010) used the median values of pixels instead of the mean values of pixels for generating a 2D histogram. The results of this method are not different from the original one.

Chen *et al.* (2010a) showed that ignoring edge and noise regions in calculation of many existing 2D thresholding methods causes unsuccessful segmentation results, and they used a threshold line instead of a threshold vector. Chen *et al.* (2010b) 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.

3.2 Three-dimensional (3D) Otsu's thresholding method

Jing *et al.* (2003) proposed a three-dimensional (3D) Otsu's method that selects an optimal threshold vector on a 3D histogram. This 3D histogram uses the first and the second features similar to a 2D histogram and additionally uses the median values of neighboring pixels as the third feature. The detail of this method is as follows.

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. $g(x, y)$ is defined as (10) and $h(x, y)$ is defined as

$$h(x,y) = med\left\{ f(x+i, y+j) : i = -\frac{k}{2}, \ldots, \frac{k}{2};\ j = -\frac{k}{2}, \ldots, \frac{k}{2} \right\} \tag{23}$$

where, k=3, in this thesis. 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. 4(a). Let c_{ijk} denote the frequency of a triple (i, j, k). Its joint probability can be expressed as

$$p_{ijk} = \frac{c_{ijk}}{N} \tag{24}$$

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. 4(b)-4(d). Let C_0 and C_1 represent the object and the background, respectively, or vice versa. Let R_x represents the rectangular volume x, where x is the rectangular volume number. Let m_x, ω_x, and μ_x represent the summation vector, the probability, and the mean vector of R_x, respectively, and μ_T represent the total mean vector. m_x can be expressed as

$$\begin{aligned} m_x &= \omega_x \mu_x = (m_{xi}, m_{xj}, m_{xk})^T \\ &= \left(\sum_{(i,j,k)\in R_x} ip_{ijk}, \sum_{(i,j,k)\in R_x} jp_{ijk}, \sum_{(i,j,k)\in R_x} kp_{ijk} \right)^T \end{aligned} \tag{25}$$

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,

$$\omega_0 + \omega_1 \approx 1 \tag{26}$$

$$\omega_0 \mu_0 + \omega_1 \mu_1 \approx \mu_T \tag{27}$$

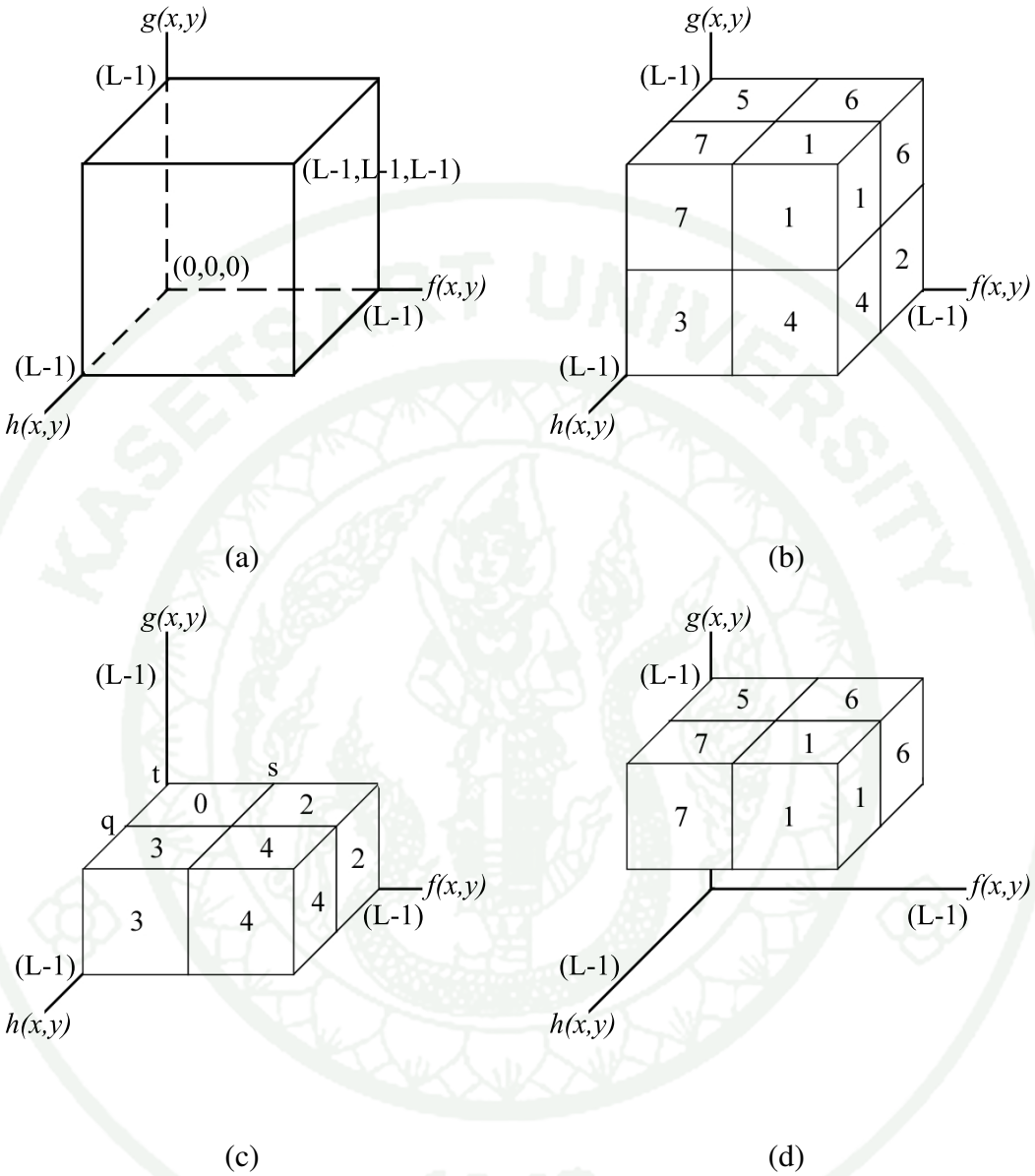


Figure 4 3D histogram

The probabilities of C_0 and C_1 thus can be denoted as

$$\begin{aligned}\omega_0 &= \sum_{(i,j,k)\in R_0} p_{ijk} \\ &= \sum_{i=0}^{s}\sum_{j=0}^{t}\sum_{k=0}^{q} p_{ijk} \qquad (28)\\ \omega_1 &= \sum_{(i,j,k)\in R_1} p_{ijk} \\ &= \sum_{i=s+1}^{L-1}\sum_{j=t+1}^{L-1}\sum_{k=q+1}^{L-1} p_{ijk} \qquad (29)\end{aligned}$$

The mean vectors of C_0 and C_1 can be expressed as

$$\begin{aligned}\mu_0 &= (\mu_{0i}, \mu_{0j}, \mu_{0k})^T \\ &= \left(\frac{m_{0i}}{\omega_0}, \frac{m_{0j}}{\omega_0}, \frac{m_{0k}}{\omega_0}\right)^T \\ &= \left(\sum_{(i,j,k)\in R_0} \frac{ip_{ijk}}{\omega_0}, \sum_{(i,j,k)\in R_0} \frac{jp_{ijk}}{\omega_0}, \sum_{(i,j,k)\in R_0} \frac{kp_{ijk}}{\omega_0}\right)^T \qquad (30)\\ \mu_1 &= (\mu_{1i}, \mu_{1j}, \mu_{1k})^T \\ &= \left(\frac{m_{1i}}{\omega_1}, \frac{m_{1j}}{\omega_1}, \frac{m_{1k}}{\omega_1}\right)^T \\ &= \left(\sum_{(i,j,k)\in R_1} \frac{ip_{ijk}}{\omega_1}, \sum_{(i,j,k)\in R_1} \frac{jp_{ijk}}{\omega_1}, \sum_{(i,j,k)\in R_1} \frac{kp_{ijk}}{\omega_1}\right)^T \qquad (31)\end{aligned}$$

The total mean vector of 3D histogram is

$$\begin{aligned}\mu_T &= (\mu_{Ti}, \mu_{Tj}, \mu_{Tk})^T \\ &= \left(\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} ip_{ijk}, \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} jp_{ijk}, \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} kp_{ijk}\right)^T \qquad (32)\end{aligned}$$

The between-class discrete matrix is defined as

$$S_B(s,t,q) = \omega_0[(\mu_0 - \mu_T)(\mu_0 - \mu_T)^T] + \omega_1[(\mu_1 - \mu_T)(\mu_1 - \mu_T)^T] \qquad (33)$$

The trace of discrete matrix can be expressed as

$$\begin{aligned} tr(S_B(s,t,q)) &= \omega_0[(\mu_{0i}-\mu_{Ti})^2+(\mu_{0j}-\mu_{Tj})^2+(\mu_{0k}-\mu_{Tk})^2]+ \\ &\quad \omega_1[(\mu_{1i}-\mu_{Ti})^2+(\mu_{1j}-\mu_{Tj})^2+(\mu_{1k}-\mu_{Tk})^2] \end{aligned} \tag{34}$$

The optimal threshold vector (s^*, t^*, q^*) is

$$(s^*, t^*, q^*) = \arg \max_{0 \le s,t,q \le L-1} (tr(S_B(s,t,q))) \tag{35}$$

The binary image $f_T(x,y)$ can be obtained by

$$f_T(x,y) = \begin{cases} 0 & \text{if } f(x,y) \le s^* \text{ and } g(x,y) \le t^* \text{ and } h(x,y) \le q^* \\ 1 & \text{if } f(x,y) > s^* \text{ and } g(x,y) > t^* \text{ and } h(x,y) > q^* \end{cases} \tag{36}$$

In case of otherwise, the pixels are ignored and are set to either 0 or 1. In this thesis, we set them to 1.

Since the 3D Otsu's method uses more spatial information than the 2D Otsu's method, the 3D Otsu's method can give better results with the time complexity of $O(L^3)$. Wang *et al.* (2008) 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 iterations. The method has the same thesholding results as the traditional 3D Otsu's method, however, its time complexity is still $O(L^3)$. Wang *et al.* (2010) thus used shuffled frog-leaping algorithm, which is meta-heuristic evolutionary algorithm, to speed up the 3D Otsu's method.

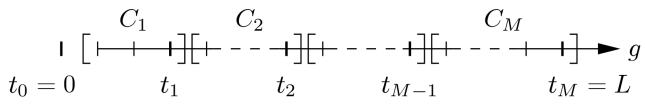


Figure 5 Separation of gray levels into classes

Source: Eichmann and Lüssi (2005)

4. Multilevel Otsu's thresholding method

Since the 1D bi-level Otsu's thresholding method is insufficient for the complex image analysis, it was extended to a 1D multilevel Otsu's thresholding method. The task of multilevel thresholding is to separate the pixels of the image, which are represented by L gray levels, into M classes, $C_1, C_2, \ldots, C_M$, by setting the thresholds $t_1, t_2, \ldots, t_{M-1}$ as shown in Fig. 5. The multilevel Otsu's thresholding method selects the thresholds, which maximize the between-class variance to separate the pixels into several classes. The between-class variance for multilevel Otsu's thresholding method is defined as

$$\sigma_B^2 = \sum_{x=1}^{M} \omega_x (\mu_x - \mu_T)^2 \tag{37}$$

where σ_B^2 is the between-class variance, ω_x is the probability of class x, μ_x is the mean value of class x, and μ_T is the total mean of the histogram. This method calculates the between-class variance for all possible thresholds so the computational time is exponentially increased when the number of classes is increased.

Liao *et al.* (2001) showed that the between-class variance can be modified which can be written as

$$\begin{aligned}\sigma_B^2 &= \sum_{x=1}^{M} \omega_x(\mu_x - \mu_T)^2 \\ &= \sum_{x=1}^{M} \omega_x(\mu_x^2 - 2\mu_x\mu_T + \mu_T^2) \\ &= \sum_{x=1}^{M} \omega_x\mu_x^2 - 2\mu_T \sum_{x=1}^{M} \omega_x\mu_x + \mu_T^2 \sum_{x=1}^{M} \omega_x \\ &= \sum_{x=1}^{M} \omega_x\mu_x^2 - \mu_T^2 \end{aligned} \tag{38}$$

Because $\sum_{x=1}^{M} \omega_x\mu_x = \mu_T$ and $\sum_{x=1}^{M} \omega_x = 1$, the term of μ_T^2 in (38) is independent for searching the optimal thresholds and can be omitted for an efficient computation. Therefore, the modified between-class variance is defined as

$$\sigma_B^2 = \sum_{x=1}^{M} \omega_x\mu_x^2 \tag{39}$$

The look-up tables are build to store the value of every possible combination of gray level p and q ($0 \leq p < q \leq L-1$) and are used to increase the speed of an exhaustive search. However, it still takes a long execution time. Huang and Wang (2009) thus split the searching space of optimal thresholds into two stages. In the first stage, the histogram is divided into groups to reduce the number of searching. Each group contain a certain range of gray levels. The gray value and the joint probability of each group are the average of gray levels and the summation of joint probability of gray levels in that group, respectively. The groups that maximize the between-class variance are selected. In the final stage, the optimal thresholds are searched by considering only gray levels in the selected groups from the first stage.

Eichmann and Lüssi (2005) used dynamic programming, which is a well-known and generic technique for solving optimization problems, to find the optimal thresholds. Dynamic programming algorithm known as the shortest path algorithm is used. This algorithm is implemented by using the trellis structure. The time complexity of this algorithm is $O(ML^2)$. Since the problem of finding the optimal paths to all the nodes in one stage in the trellis structure is equivalent to the problem of finding the row wise maxima in a lower triangular matrix, which is called *search matrix*; and the class cost of Otsu's method is proved that it fulfills the convex quadrangle inequality, which causes that the search matrix of Otsu's method is totally monotone. Therefore, it can use the SMAWK algorithm which is an efficient algorithm for finding the row maxima in totally monotone matrices. This algorithm can find the row wise maxima of $m \times n$ matrix ($m < n$) in $O(n)$ time. So the optimal thresholds can be found in $O(ML)$ time.

Dongju and Jian (2009) proved that the objective function of Otsu's method is equivalent to that of K-means in bi-level and multilevel thresholding. Moreover, K-means does not require a computation of a gray level histogram before running but Otsu's method does need to compute a gray level histogram. K-means thus can be extended to multilevel thesholding method and also can be extended to 2D and 3D thesholding methods. Evidently K-means can perform more efficiently than Otsu's.

Moreover, some research work (Hao *et al.*, 2010; Horng, 2010; Nabizadeh *et al.*, 2010; Shu-Chien, 2009) used an evolutionary algorithm to find the optimal thresholds. Hao *et al.* (2010) and Nabizadeh *et al.* (2010) used Otsu's method as a fitness function, while Shu-Chien (2009) and Horng (2010) used an entropy based method as a fitness function. Hammouche *et al.* (2010) compared research work within this group and found that their optimal thresholds are inconsistent for each searching.

5. Misclassification error (ME) and Modified Hausdorff distance (MHD)

The misclassification error (ME) is used to present the number of background pixels wrongly assigned to the foreground, and vice versa; and the modified Hausdorff distance (MHD) is used to measure the shape distortion of each result image compared with its corresponding ground truth. ME and MHD are defined as (Sezgin and Sankur, 2004)

$$\mathsf{ME} = 1 - \frac{|B_O \cap B_T| + |F_O \cap F_T|}{|B_O| + |F_O|} \tag{40}$$

$$\mathsf{MHD} = \max(d_{\mathsf{MHD}}(F_O, F_T), d_{\mathsf{MHD}}(F_T, F_O)) \tag{41}$$

$$d_{\mathsf{MHD}}(F_O, F_T) = \frac{1}{|F_O|} \sum_{f_O \in F_O} \min_{f_T \in F_T} \|f_O - f_T\| \tag{42}$$

$$d_{\mathsf{MHD}}(F_T, F_O) = \frac{1}{|F_T|} \sum_{f_T \in F_T} \min_{f_O \in F_O} \|f_T - f_O\| \tag{43}$$

where F_i and B_i denote the foreground and background pixels, respectively, of images i, which include 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).

For multilevel thresholding evaluation, we have modified ME as

$$\mathsf{MME} = 1 - \frac{\sum_{k=1}^{M} |C_{kO} \cap C_{kT}|}{\sum_{k=1}^{M} |C_{kO}|} \tag{44}$$

where C_{ki} denotes the pixels of class k of an image i, which includes the ground truth (O) and thresholded (T) images. $|.|$ is the cardinality of the set. M is the number of classes.

METERIALS AND METHODS

Metarials

1. Hardware

    1) Computer notebook with Intel(R) Core(TM)2 Duo CPU T7300 @ 2.00GHz, 2.99 GB of RAM

2. Software

    1) Microsoft Window XP
    2) Microsoft Visual Studio 2010
    3) OpenCV Library
    4) Scilab

Methods

There are two approaches to combine multilevel thresholding with the multi-dimensional thresholding method. The first approach is to extend a 1D multilevel thresholding method to either 2D or 3D multilevel thresholding method, but it is inefficient because the computation of threshold cannot take advantage of the recursive structure of the objective function (Liao *et al.*, 2001). Remind that both of them require a long execution time. When we combine them together by this approach, the computational time of combined method will be increased dramatically. The second approach is to use histogram projections in the multi-dimensional with bi-level thresholding method to form 1D histograms and then apply the 1D multilevel thresholding method on them. This approach is more efficient than the first one because histogram projections can reduce the search space for thresholds in the multi-dimensional thresholding method, and also apply 1D multilevel thresholding method without any modification.

In this thesis, we use the second approach to combine multilevel thresholding with the multi-dimensional thresholding method, however, there are still some problems in multi-dimensional thresholding methods that should be taken care of before combining. These problems are as follows. Ignoring edge and noise regions in calculation causes unsuccessful segmentation results (Chen *et al.*, 2010a). The region division by a threshold vector in 2D Otsu's method is not reasonable, which some object and background regions are wrongly assigned to edge and noise regions, or vice versa (Chen *et al.*, 2010b).

1. Improving multi-dimensional Otsu's thresholding method

We have improved both the 2D Otsu's thresholding method (Sthitpattanapongsa and Srinark, 2011) and the 3D Otsu's thresholding method (Sthitpattanapongsa and Srinark, 2012) to reduce computational time, increase noise resistance and increase accuracy in segmentation.

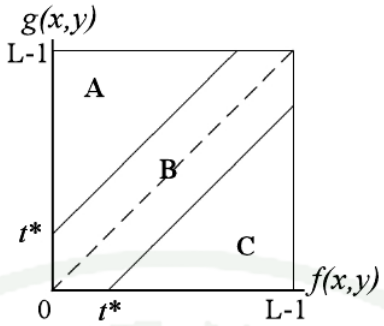


Figure 6 Region division by t^* on the 2D histogram of the proposed method

1.1 Proposed 2D Otsu's thresholding method

We observe that pixels of the object, the background, edges, and noise have their own unique characteristics. Interior pixels of the object have their original gray levels close to their average gray levels; and this characteristic is also applied with interior pixels of the background. However, pixels representing edges and noise have their original gray levels far different from their average gray levels.

We use the gradient gray level to reflect this gray level dissimilarity values. It can be noticed that in the 2D histogram, the gradient magnitude values along the diagonal are zero, and are increased, where coordinates (i, j) are far away from the diagonal. In the same way, the small gradient magnitude values likely appears on pixels representing the object and the background while the large gradient magnitude values likely appears on pixels representing edges and noise. Therefore, the probability of classes near the diagonal can be approximated to 1, while the probability of classes far away from the diagonal is approximated to 0. It is obvious that the region division of the traditional 2D's Otsu method does not suitably separate the region. We can thus take advantage from the observed characteristics such that the 2D histogram can be divided into regions that separate the object and background from edges and noise . We can compute a gradient gray level histogram from the 2D histogram as follow

$$H_{gradient}(x) = \sum_{x=|i-j|} p_{ij} \tag{45}$$

where i and $j = 0, 1, \ldots, L-1$.

In the next step, we apply Otsu's method on the 1D histogram, $H_{gradient}$, to get an optimal threshold of the gradient gray level, t^*, for a region division of the 2D histogram. We apply t^* to generate two lines that are parallel to the diagonal so that the 2D histogram can be divided into three regions as shown in Fig. 6. The region B contains coordinates that have lower gradient values than t^*, so it is the area close along the diagonal (dashed line). The coordinates in the region B highly represent either the object or the background. The regions A and C contain coordinates that have higher gradient values than t^*, so they are far away from the diagonal. The coordinates in these two regions possibly represent either edges or noise.

We then project the 2D histogram based on each region's property to generate a new 1D histogram. Coordinates in the region B are projected by their original gray levels. Coordinates in the regions A and C are projected by their average gray levels instead of their original gray levels, since they are possibly either edge or noise pixels and ignoring them causes unsatisfactory results. In this case, it is the same as the gray level of the edges and noise pixels in an image $f(x, y)$ is replaced by its average gray level in an neighborhood average image $g(x, y)$ for noise removal before creating the 1D histogram of $f(x, y)$ for selecting the optimal threshold. The projected 1D histogram can be thus expressed as

$$H_{projected}(y) = \sum_{|y-j| \leq t^*} p_{yj} + \sum_{|i-y| > t^*} p_{iy} \tag{46}$$

where y, i and $j = 0, 1, \ldots, L-1$. The first term represents the projection by original gray levels. The second term represents the projection by average gray levels.

Finally, we apply the 1D bi-level Otsu's method on $H_{projected}$ to select the optimal threshold, s^*. Currently, we have two thresholds, s^* and t^*, for classification of image pixels. t^* is the gradient gray level, which decides whether the gray level of each pixel for classification comes from either $f(x,y)$ or $g(x,y)$. s^* is the gray level, which decides whether the gray level of each pixel is either the object or background. Therefore, the binary image $f_T(x,y)$ can be obtained by

$$f_T(x,y) = \begin{cases} 0 & \text{if } |f(x,y) - g(x,y)| \leq t^* \text{ and } f(x,y) \leq s^* \\ 0 & \text{if } |f(x,y) - g(x,y)| > t^* \text{ and } g(x,y) \leq s^* \\ 1 & \text{otherwise.} \end{cases} \tag{47}$$

1.2 Proposed 3D Otsu's thresholding method

From (26)-(27), we can see that the probabilities and the summation vectors of volume 0 and 1 are approximated to the probabilities and the summation of all volumes, respectively. So we can conclude that

$$\omega_x \approx 0 \tag{48}$$

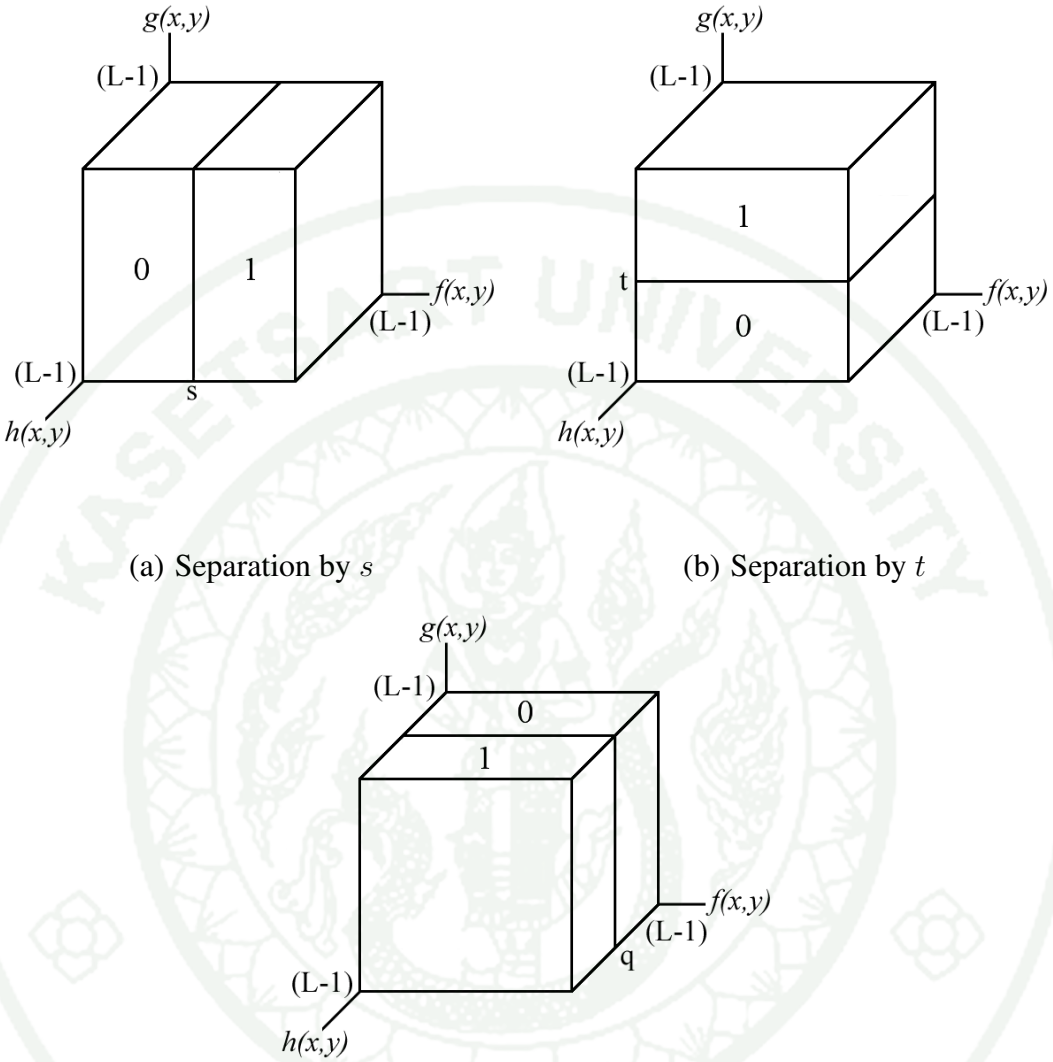
$$m_{xi} \approx 0 \tag{49}$$

$$m_{xj} \approx 0 \tag{50}$$

$$m_{xk} \approx 0 \tag{51}$$

where $x = 2, \ldots, 7$.

For each element of the threshold vector (s,t,q), it creates a plane that separates the 3D histogram into two volumes as shown in Fig. 7. Let ω_{xy} be the probability of the volume x, which is separated by the threshold y. m_{xy} and μ_{xy} are the summation and the mean of gray levels that are on the same axis as the threshold y and are in the volume x which is separated by the threshold y.



(a) Separation by s

(b) Separation by t

(c) Separation by q

Figure 7 3D histogram separation by each element of the threshold vector (s, t, q)

For the threshold s that separates the histogram in Fig. 7(a), ω_{xy} and m_{xy} for this threshold can be expressed as

$$\omega_{0s} = \sum_{i=0}^{s}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} p_{ijk} \tag{52}$$

$$\omega_{1s} = \sum_{i=s+1}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} p_{ijk} \tag{53}$$

$$m_{0s} = \sum_{i=0}^{s}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} ip_{ijk} \tag{54}$$

$$m_{1s} = \sum_{i=s+1}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} ip_{ijk} \tag{55}$$

Let $\sum_{j=0}^{L-1}\sum_{k=0}^{L-1} p_{ijk} = P_i$. So we can rewrite (52)-(55) and μ_{Ti} in (32) as

$$\omega_{0s} = \sum_{i=0}^{s} P_i \tag{56}$$

$$\omega_{1s} = \sum_{i=s+1}^{L-1} P_i \tag{57}$$

$$m_{0s} = \sum_{i=0}^{s} iP_i \tag{58}$$

$$m_{1s} = \sum_{i=s+1}^{L-1} iP_i \tag{59}$$

$$\mu_{Ti} = \sum_{i=0}^{L-1} iP_i \tag{60}$$

From Fig. 4, we can write the relation between the volumes that are separated by the threshold s and the volumes that are separated by the threshold vector (s, t, q) as

$$\omega_{0s} = \omega_0 + \omega_3 + \omega_5 + \omega_6 \tag{61}$$

$$\omega_{1s} = \omega_1 + \omega_2 + \omega_4 + \omega_7 \tag{62}$$

$$m_{0s} = m_{0i} + m_{3i} + m_{5i} + m_{6i} \tag{63}$$

$$m_{1s} = m_{1i} + m_{2i} + m_{4i} + m_{7i} \tag{64}$$

From (48)-(51), we can conclude that

$$\omega_{0s} \approx \omega_0 \tag{65}$$

$$\omega_{1s} \approx \omega_1 \tag{66}$$

$$m_{0s} \approx m_{0i} \tag{67}$$

$$m_{1s} \approx m_{1i} \tag{68}$$

$$\mu_{0s} = \frac{m_{0s}}{\omega_{0s}} \approx \frac{m_{0i}}{\omega_0} = \mu_{0i} \tag{69}$$

$$\mu_{1s} = \frac{m_{1s}}{\omega_{1s}} \approx \frac{m_{1i}}{\omega_1} = \mu_{1i} \tag{70}$$

For the threshold t that separates the histogram in Fig. 7(b), ω_{xy} and m_{xy} for this threshold can be expressed as

$$\omega_{0t} = \sum_{i=0}^{L-1} \sum_{j=0}^{t} \sum_{k=0}^{L-1} p_{ijk} \tag{71}$$

$$\omega_{1t} = \sum_{i=0}^{L-1} \sum_{j=t+1}^{L-1} \sum_{k=0}^{L-1} p_{ijk} \tag{72}$$

$$m_{0t} = \sum_{i=0}^{L-1} \sum_{j=0}^{t} \sum_{k=0}^{L-1} j p_{ijk} \tag{73}$$

$$m_{1s} = \sum_{i=0}^{L-1} \sum_{j=t+1}^{L-1} \sum_{k=0}^{L-1} j p_{ijk} \tag{74}$$

Let $\sum_{i=0}^{L-1} \sum_{k=0}^{L-1} p_{ijk} = P_j$. So we can rewrite (71)-(74) and μ_{Tj} in (32) as

$$\omega_{0t} = \sum_{j=0}^{t} P_j \tag{75}$$

$$\omega_{1t} = \sum_{j=t+1}^{L-1} P_j \tag{76}$$

$$m_{0t} = \sum_{j=0}^{t} j P_j \tag{77}$$

$$m_{1t} = \sum_{j=t+1}^{L-1} jP_j \tag{78}$$

$$\mu_{Tj} = \sum_{j=0}^{L-1} jP_j \tag{79}$$

From Fig. 4, we can write the relation between the volumes that are separated by the threshold t and the volumes that are separated by the threshold vector (s, t, q) as

$$\omega_{0t} = \omega_0 + \omega_2 + \omega_3 + \omega_4 \tag{80}$$

$$\omega_{1t} = \omega_1 + \omega_5 + \omega_6 + \omega_7 \tag{81}$$

$$m_{0t} = m_{0j} + m_{2j} + m_{3j} + m_{4j} \tag{82}$$

$$m_{1t} = m_{1j} + m_{5j} + m_{6j} + m_{7j} \tag{83}$$

From (48)-(51), we can conclude that

$$\omega_{0t} \approx \omega_0 \tag{84}$$

$$\omega_{1t} \approx \omega_1 \tag{85}$$

$$m_{0t} \approx m_{0j} \tag{86}$$

$$m_{1t} \approx m_{1j} \tag{87}$$

$$\mu_{0t} = \frac{m_{0t}}{\omega_{0t}} \approx \frac{m_{0j}}{\omega_0} = \mu_{0j} \tag{88}$$

$$\mu_{1t} = \frac{m_{1t}}{\omega_{1t}} \approx \frac{m_{1j}}{\omega_1} = \mu_{1j} \tag{89}$$

For the threshold q that separates the histogram in Fig. 7(c), ω_{xy} and m_{xy} for this threshold can be expressed as

$$\omega_{0q} = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \sum_{k=0}^{q} p_{ijk} \tag{90}$$

$$\omega_{1q} = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \sum_{k=q+1}^{L-1} p_{ijk} \tag{91}$$

$$m_{0q} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=0}^{q} kp_{ijk} \tag{92}$$

$$m_{1q} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\sum_{k=q+1}^{L-1} kp_{ijk} \tag{93}$$

Let $\sum_{i=0}^{L-1}\sum_{j=0}^{L-1} p_{ijk} = P_k$. So we can rewrite (90)-(93) and μ_{Tk} in (32) as

$$\omega_{0q} = \sum_{k=0}^{q} P_k \tag{94}$$

$$\omega_{1q} = \sum_{k=q+1}^{L-1} P_k \tag{95}$$

$$m_{0q} = \sum_{k=0}^{q} kP_k \tag{96}$$

$$m_{1q} = \sum_{k=q+1}^{L-1} kP_k \tag{97}$$

$$\mu_{Tk} = \sum_{k=0}^{L-1} kP_k \tag{98}$$

From Fig. 4, we can write the relation between the volumes that are separated by the threshold q and the volumes that are separated by the threshold vector (s, t, q) as

$$\omega_{0q} = \omega_0 + \omega_2 + \omega_5 + \omega_6 \tag{99}$$

$$\omega_{1q} = \omega_1 + \omega_3 + \omega_4 + \omega_7 \tag{100}$$

$$m_{0q} = m_{0k} + m_{2k} + m_{5k} + m_{6k} \tag{101}$$

$$m_{1q} = m_{1k} + m_{3k} + m_{4k} + m_{7k} \tag{102}$$

From (48)-(51), we can conclude that

$$\omega_{0q} \approx \omega_0 \tag{103}$$

$$\omega_{1q} \approx \omega_1 \tag{104}$$

$$m_{0q} \approx m_{0k} \tag{105}$$

$$m_{1q} \approx m_{1k} \tag{106}$$

$$\mu_{0q} = \frac{m_{0q}}{\omega_{0q}} \approx \frac{m_{0k}}{\omega_0} = \mu_{0k} \tag{107}$$

$$\mu_{1q} = \frac{m_{1q}}{\omega_{1q}} \approx \frac{m_{1k}}{\omega_1} = \mu_{1k} \tag{108}$$

Notice that P_i, P_j, and P_k are equivalent with the 1D histogram of the original, mean-filtered, and median-filtered images, respectively. From (65), (69)-(70), (84), (88)-(89), (103), and (107)-(108), we can rewrite (34) as

$$\begin{aligned} tr(S_B(s,t,q)) \approx\ & \overbrace{\left[\omega_{0s}(\mu'_{0s}-\mu_{Ti})^2+\omega_{1s}(\mu'_{1s}-\mu_{Ti})^2\right]}^{A}+ \\ & \overbrace{\left[\omega_{0j}(\mu'_{0j}-\mu_{Tj})^2+\omega_{1j}(\mu'_{1j}-\mu_{Tj})^2\right]}^{B}+ \\ & \overbrace{\left[\omega_{0k}(\mu'_{0k}-\mu_{Tk})^2+\omega_{1k}(\mu'_{1k}-\mu_{Tk})^2\right]}^{C} \end{aligned} \tag{109}$$

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

$$\sigma^2_{Bi}(s) = \omega_{0s}(\mu_{0s}-\mu_{Ti})^2+\omega_{1s}(\mu_{1s}-\mu_{Ti})^2 \tag{110}$$

$$\sigma^2_{Bj}(t) = \omega_{0t}(\mu_{0t}-\mu_{Tj})^2+\omega_{1t}(\mu_{1t}-\mu_{Tj})^2 \tag{111}$$

$$\sigma^2_{Bk}(q) = \omega_{0q}(\mu_{0q}-\mu_{Tk})^2+\omega_{1q}(\mu_{1q}-\mu_{Tk})^2 \tag{112}$$

The optimal threshold (s^*,t^*,q^*) is

$$\begin{aligned} (s^*,t^*,q^*) &= arg \max_{0\le s,t,q\le L-1}(tr(S_B(s,t,q))) \\ &\approx arg \max_{0\le s,t,q\le L-1}(\sigma^2_{Bi}(s)+\sigma^2_{Bj}(t)+\sigma^2_{Bk}(q)) \end{aligned} \tag{113}$$

which can be splited into

$$s^* = arg \max_{0\le s\le L-1}\sigma^2_{Bi}(s) \tag{114}$$

$$t^* = arg \max_{0\le t\le L-1}\sigma^2_{Bj}(t) \tag{115}$$

$$q^* = arg \max_{0 \le q \le L-1} \sigma_{Bk}^2(q) \tag{116}$$

Equations (114), (115), and (116) are in the same form as (8), which is the threshold selection of 1D Otsu's method. So 1D Otsu's method can be used to select the optimal threshold of the original, mean-filtered, and median-filtered images. 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 $O(L^3)$. The flowchart of this method is shown as Fig. 8.

In the classification step, combining three thresholds (s^*, t^*, and q^*) to a 3D threshold vector and classifying the image pixels as 3D Otsu's thresholding method cause unsuccessful segmentation results because there are ignored pixels in classification. For pixels at (x, y), we thus use s^* to classify $f(x, y)$, t^* to classify $g(x, y)$, and q^* to classify $h(x, y)$. Now, we have class results from each classification and use the mostly selected class as the class of binary image $f_T(x, y)$ as

$$f_T(x,y) = \begin{cases} 0 & \text{if } (f(x,y) \le s^* \text{ and } (g(x,y) \le t^* \text{ or } h(x,y) \le q^*)) \text{ or} \\ & (g(x,y) \le t^* \text{ and } h(x,y) \le q^*) \\ 1 & \text{otherwise.} \end{cases} \tag{117}$$

2. Multilevel and multi-dimensional Otsu's thresholding method

Based on the experimental results, which can be seen in the Results and Discussion Section, they show that the proposed 3D Otsu's method has more noise resistance and less execution time than the 2D one. We thus apply a 1D multilevel Otsu's thresholding method with the proposed 3D Otsu's method in order to gain the property of noise resistance from the multi-dimensional method, and the property of complex image analysis from the multilevel method.

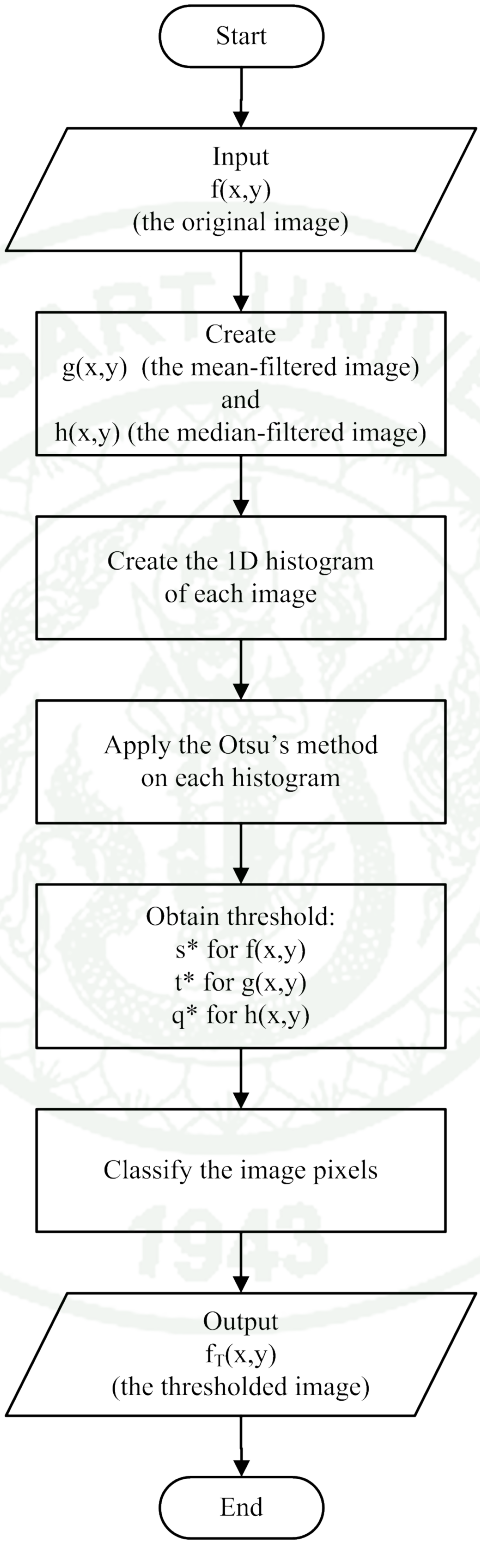


Figure 8 The flowchart of the proposed 3D Otsu's thresholding method

For the multilevel and multi-dimensional Otsu's thresholding method, we replace the 1D bi-level threshold selection of the proposed 3D Otsu's method with the 1D multilevel threshold selection to select the optimal thresholds of each 1D histogram. The flowchart of this method is shown in Fig. 9. The reason of using Eichmann's method is that the time complexity of this method is $O(ML)$ while the time complexity of Liao's method is $O((L-M)^{M-1})$ and the time complexity of K-means method depends on the number of iteration.

Eichmann's method uses the shortest path algorithm to find the optimal thresholds. A partial sum up to the gray level l for the first m classes is defined as

$$J_m(l) = \sum_{k=1}^{m} c(t_{k-1}, t_k], \qquad 1 \leq t_1 < t_2 < \ldots < t_{m-1} < l \tag{118}$$

where $c(t_{k-1}, t_k]$ is called *class cost*. For Otsu's method, the class cost $c(p, q]$ is

$$c(p, q] = \omega(p, q](\mu(p, q])^2 \tag{119}$$

The optimal solution to a subproblem is given by

$$J_m^*(l) = \max_{m-1 \leq t_{m-1} < l} \{J_m(l)\} \tag{120}$$

The recursive optimal cost is defined as

$$J_m^*(l) = \begin{cases} \max_{m-1 \leq t_{m-1} < l} \{J_{m-1}^*(t_{m-1}) + c(t_{m-1}, l]\} & \text{if } m > 1 \\ c(0, l] & \text{if } m = 1 \end{cases} \tag{121}$$

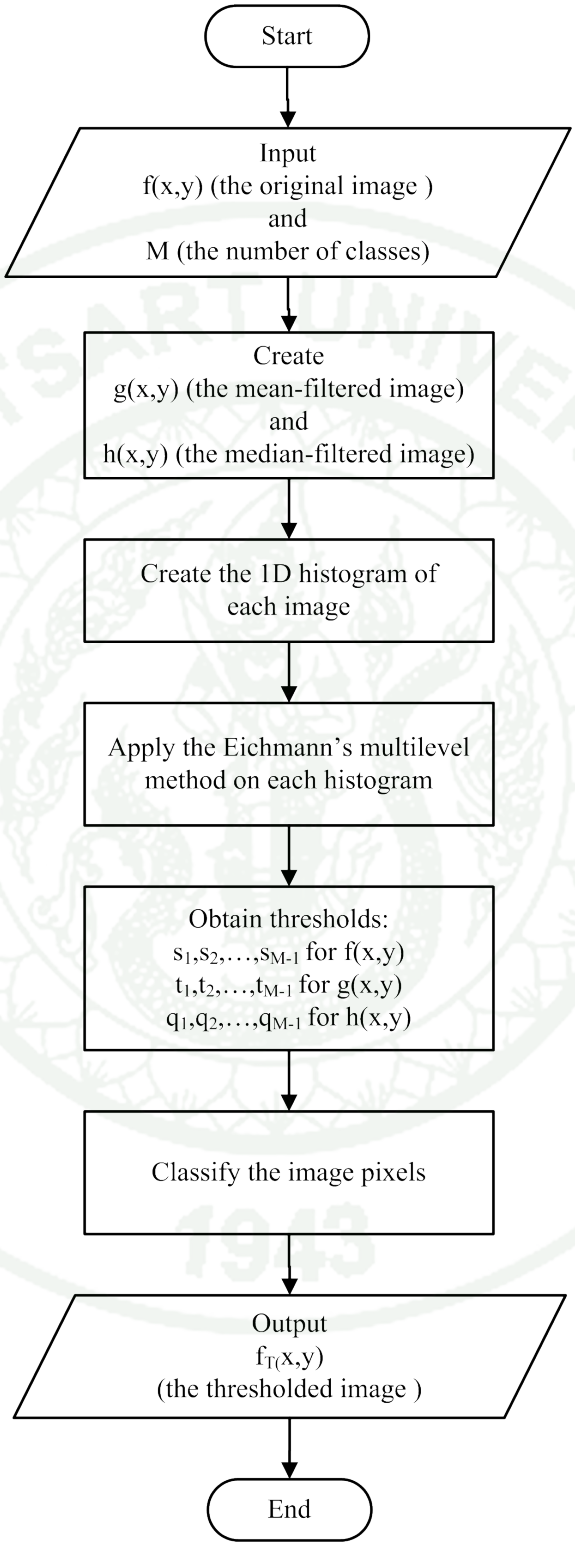


Figure 9 The flowchart of the proposed multilevel and multi-dimensional Otsu's thresholding method

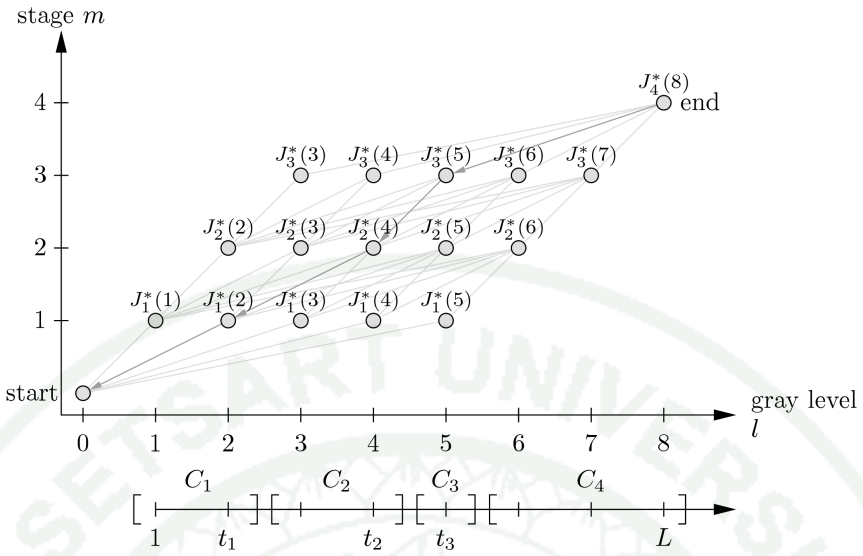


Figure 10 Trellis structure

Source: Eichmann and Lüssi (2005)

A trellis structure is used to implement this algorithm. Fig. 10 shows the trellis structure, where the gray dots are nodes and the gray lines are the path. The x-axis represents the gray levels l from 0 to L, where L is the number of gray levels. The y-axis represents the stage (m) of the structure from 1 to M, where M is the number of classes. In the start stage, there is only one node that has the path to all node in stage 1. In stages 1 to $M-1$, each stage has $L-M+1$ nodes, where each node has the paths from all lower left nodes. In the stage M, there is one node that has the path from all nodes in the stage $M-1$. The algorithm proceeds from the bottom to the top of the trellis structure. For each node, it selects a path which has the optimal value, stores the optimal value, and sets its backpointer to the lower left node, whose path is selected. The optimal thresholds can be found by following the backpointer from $J_M^*(L)$, which $J_M^*(L)$ is $J_4^*(8)$ and the backpointers is shown in dark gray arrow in Fig. 10.

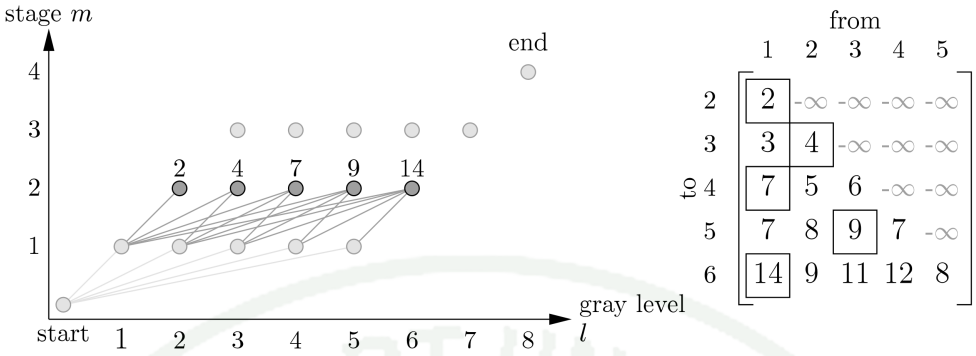


Figure 11 Equivalence to matrix search problem

Source: Eichmann and Lüssi (2005)

The problem of finding the optimal paths to all the nodes in one stage in the trellis structure is equivalent to the problem of finding the row wise maxima in a lower triangular matrix, which is shown in Fig. 11. The column of the matrix represents the source nodes of the path in lower stage. The row of the matrix represents the destination nodes of the path in a current stage. The value of each element in the matrix is the summation between the optimal value in the source node and the class cost of the path from the source node to the destination node of that element.

The class cost of Otsu's method fulfills the convex quadrangle inequality. It causes that the search matrix of Otsu's method is totally monotone. The matrix is called totally monotone if $Mat(i_1, k_1) < Mat(i_1, k_2)$ then $Mat(i_2, k_1) < Mat(i_2, k_2)$, where Mat is the $m \times n$ matrix, $0 \leq i_1 < i_2 < m$ and $0 \leq k_1 < k_2 < n$. From this property of the search matrix, SMAWK, which is an efficient algorithm for finding the row maxima in totally monotone matrices, can be used to find the row wise maxima in the search matrix.

The SMAWK function is called recursively. The input of this function is a search matrix. REDUCE function is called for each recursion to delete the columns of the matrix that are not contain the row wise maxima. The returned matrix from REDUCE is always a square matrix. If this returned matrix is 1×1 matrix, the recursion stops and SMAWK returns. Otherwise, the matrix with only the even-numbered rows of this returned matrix is used as the input in the next recursion of SMAWK. After the recursive call of SMAWK returns, the position of the maximum in even-numbered rows of the returned matrix from REDUCE are already known. MFILL is called to finds maxima in odd-numbered rows of this matrix. It has only to search the elements which are in the columns between the position of the maximum in the previous row and that in the next row.

Fig. 12 shows the operation of the SMAWK algorithm. The initial call is on a 7×7 matrix. The first recursive call is on a 3×7 matrix with only even-numbered rows of the initial matrix. After REDUCE has deleted four columns, the matrix becomes a square matrix. In the second recursive call, the matrix has the size of 1×3, REDUCE has deleted two columns, and the matrix becomes 1×1. At this point, the recursion stops. Note that the element of the last matrix is the maximum in the fourth row of the initial matrix. After the recursive call of SMAWK returns, MFILL finds the maxima in odd-numbered rows of the matrix. From the last recursive call, the position of the maximum in even-numbered rows are already known (black border) and MFILL has only to search the elements which in the gray background. After this, all the row wise maxima of the matrix have been found.

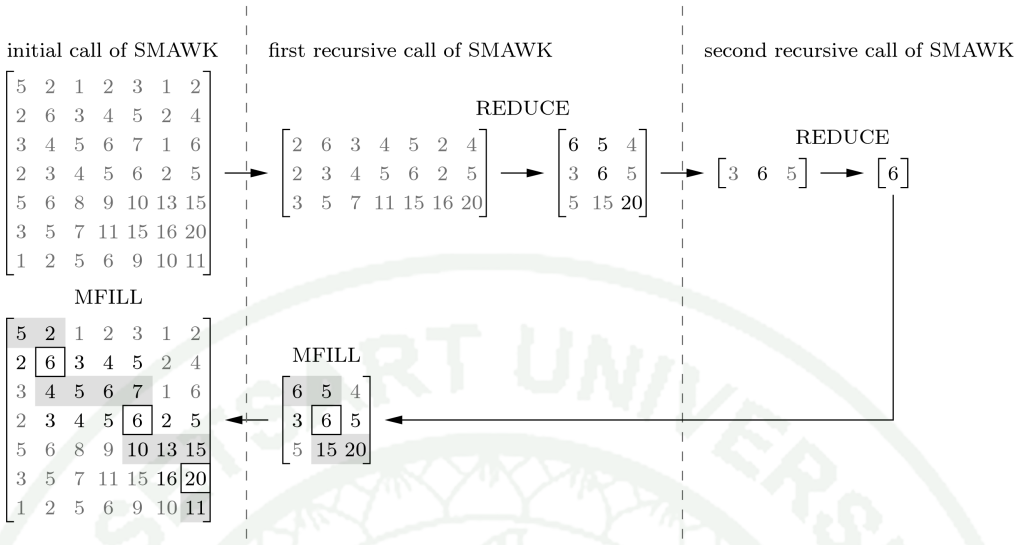


Figure 12 The operation of SMAWK algorithm

Source: Eichmann and Lüssi (2005)

The SMAWK algorithm can find the row wise maxima of $m \times n$ matrix in $O(m)$ time. In this case, the search matrix has the size, which is not over $L \times L$. Therefore, the row wise maxima can be found in $O(L)$ time for one stage. There are only M stages. The time complexity to find the optimal thresholds is $O(ML)$. We use Eichmann's method three times to select the optimal thresholds on each histogram of the original, mean-filtered, and median-filtered images. So the optimal thresholds of all histograms can be found in $O(ML)$ time in our proposed method.

In the classification step, we still use $s_1, s_2, \ldots, s_{M-1}$ to classify $f(x, y)$, $t_1, t_2, \ldots, t_{M-1}$ to classify $g(x, y)$, and $q_1, q_2, \ldots, q_{M-1}$ to classify $h(x, y)$ for pixels at (x, y). However, we cannot use the mostly selected class from all classification results as a class of thresholded image $f_T(x, y)$ because there are M classes, which M can be more than two, and all classification results can be different from each other. To solve this problem, we use the median class from all classification results as the class of $f_T(x, y)$.

RESULTS AND DISCUSSION

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 methods 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 μ= 0.

1. Results of improving multi-dimensional Otsu's thresholding method

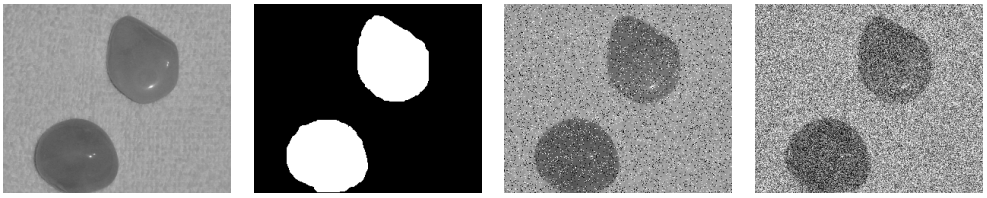
We compared our methods for both 2D and 3D methods with 1D Otsu's method (Otsu, 1979), 2D Otsu's method (Gong *et al.*, 1998), 3D Otsu's method (Wang *et al.*, 2008), K-means (Dongju and Jian, 2009) based methods for both 2D and 3D ones, Ningbo's method (Ningbo *et al.*, 2009), and Yue's method (Yue *et al.*, 2009) because they are based on Otsu's. For each experiment that the ground truth is available, we use misclassification error (ME) and modified Hausdorff distance (MHD) to measure the error of each resulting image compared with its corresponding ground truth.

(a) Original (b) Salt&Pepper ($\delta = 0.1$) (c) Gaussian ($\sigma^2 = 0.005$)

Figure 13 Lena images w/o noise added

Table 1 Resulting optimal thresholds of Lena images w/o noise added

Methods	Fig.		
	13(a)	13(b)	13(c)
1D Otsu's	117	117	119
2D Otsu's	(123,117)	(117,192)	(123,126)
3D Otsu's	(130,125,117)	(130,126,117)	(132,122,121)
2D K-means	(117,117)	(117,117)	(118,118)
3D K-means	(117,117,117)	(117,117,117)	(118,118,118)
Ningbo's	(117,117)	(117,117)	(117,119)
Yue's	(117,117)	(117,117)	(119,118)
Proposed 2D Otsu's	(117,7)	(117,9)	(118,17)
Proposed 3D Otsu's	(117,117,117)	(117,117,117)	(117,117,117)



(a) Original (b) Ground truth (c) Salt&Pepper ($\delta = 0.1$) (d) Gaussian ($\sigma^2 = 0.05$)

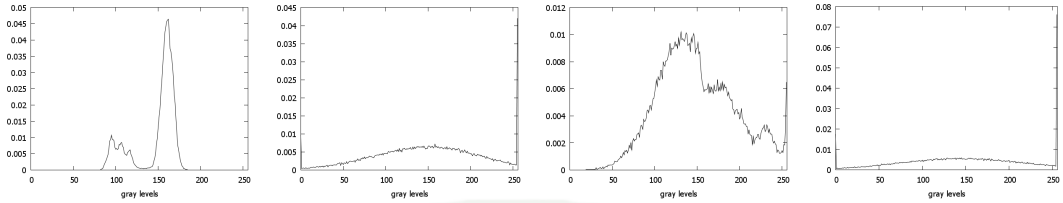
Figure 14 The first image set with sample noise added images

(a) Original (b) Ground truth (c) Salt&Pepper ($\delta = 0.1$) (d) Gaussian ($\sigma^2 = 0.05$)

Figure 15 The second image set with sample noise added images

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 δ = 0.01 to the original image, and the other one was generated by adding Gaussian noise with σ^2 = 0.005 to the original image as shown in Fig. 13. The resulting optimal thresholds are shown in Table 1.

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* (Alpert *et al.*, 2007). Fig. 14(a) and 14(b) show the first test image and its ground truth, respectively. Fig. 15(a) and 15(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 0.05. Fig. 14(c) and 14(d) show example noise added images of the first test image. Fig. 15(c) and 15(d) show example noise added images of the second test image.

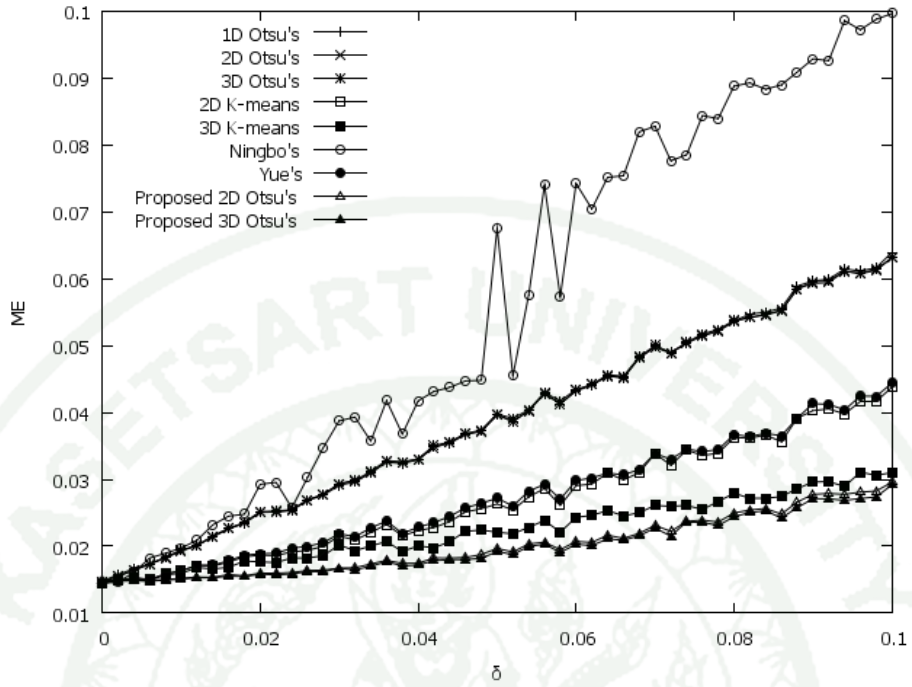


(a) Histogram of Fig. 14(a) (b) Histogram of Fig. 14(d) (c) Histogram of Fig. 15(a) (d) Histogram of Fig. 15(d)

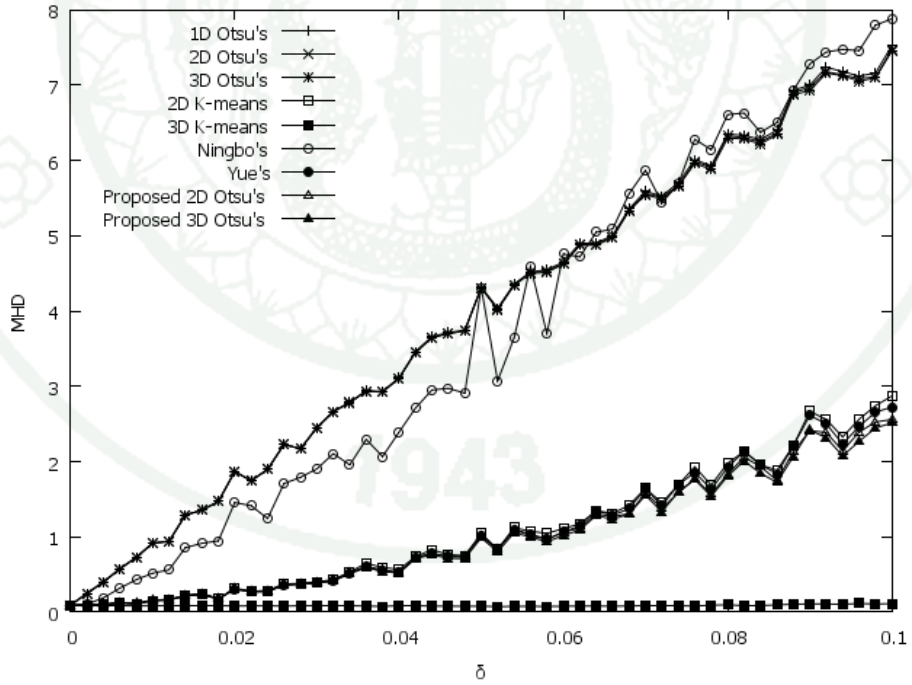
Figure 16 Histograms of the test images

Table 2 The average computational time (ms) on all noise added images

Methods	$\overline{T}$ (Gaussian noise)	$\overline{T}$ (Salt&Pepper noise)	$\overline{T}$ (Total)
1D Otsu's	0	0	0
2D Otsu's	13.32	8.03	10.67
3D Otsu's	2762.89	1790.25	2276.57
2D K-means	71.58	18.90	45.24
3D K-means	2502.81	1124.18	1813.50
Ningbo's	16.87	6.12	11.50
Yue's	5.55	1.82	3.69
Proposed 2D Otsu's	7.71	3.20	5.46
Proposed 3D Otsu's	7.56	3.5	5.53

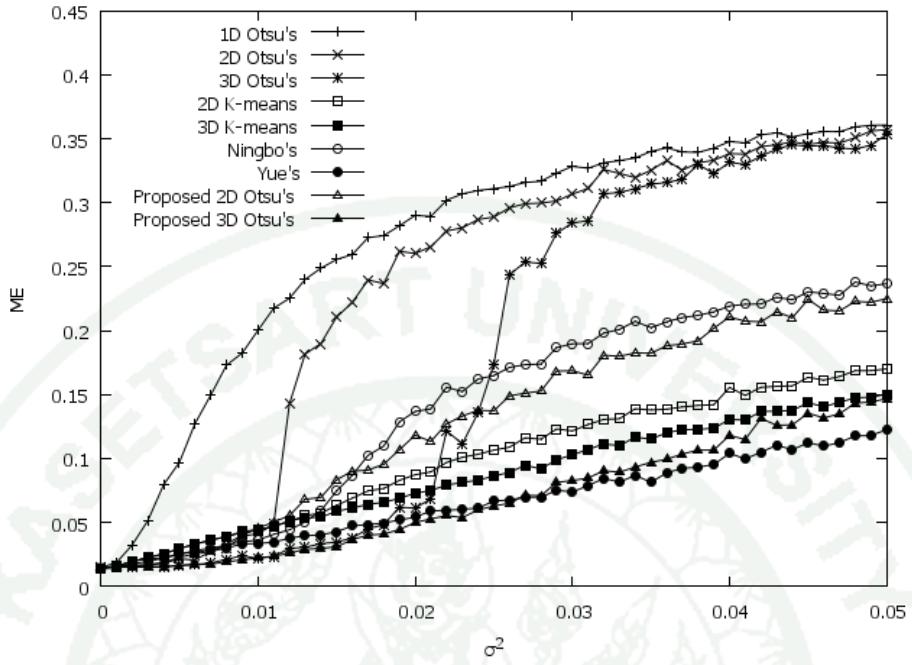


(a) ME

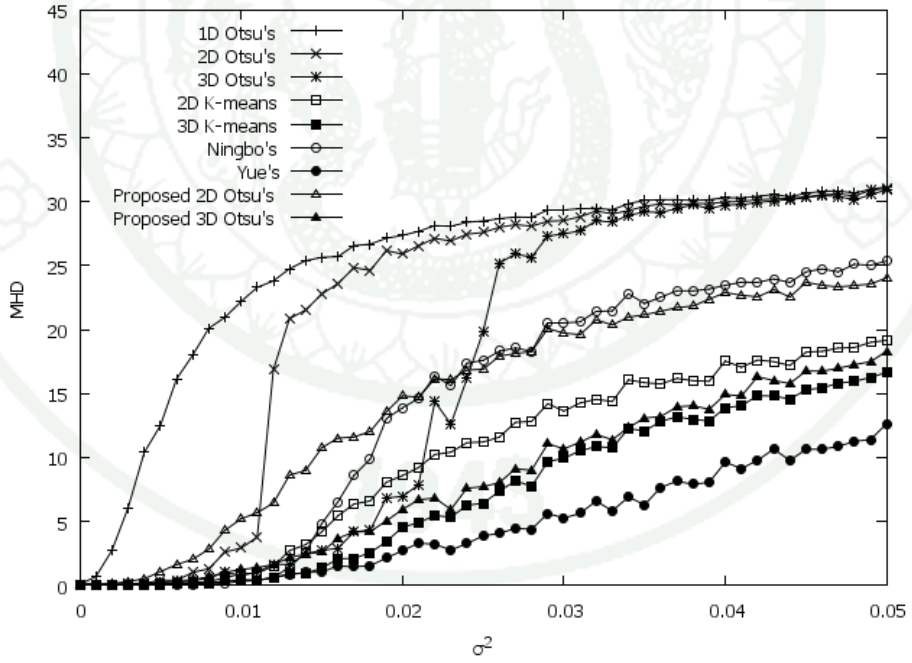


(b) MHD

Figure 17 Comparison of ME and MHD for the first test images with Salt&Pepper noise added at various δ

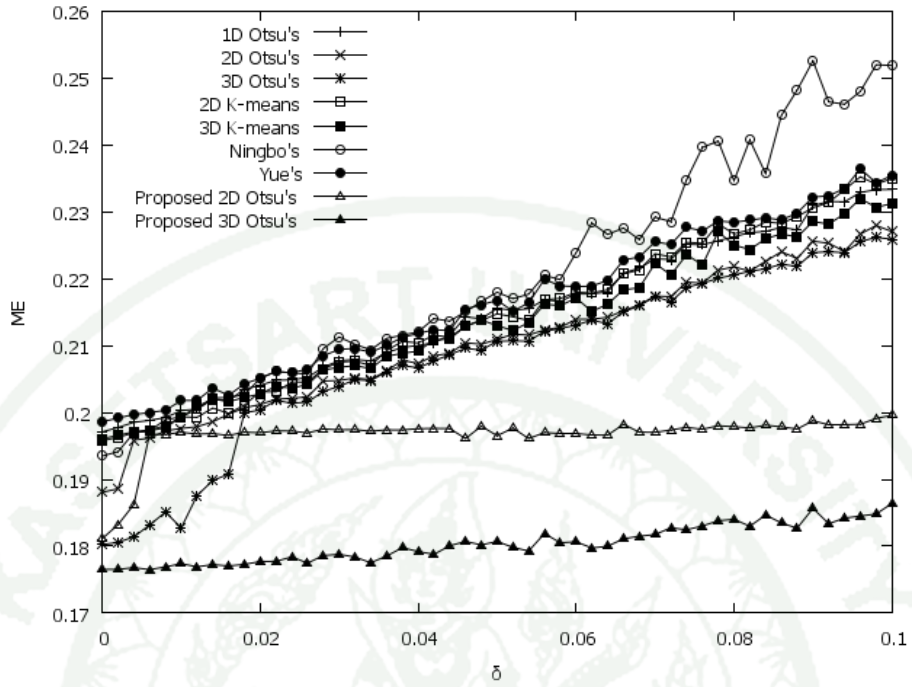


(a) ME

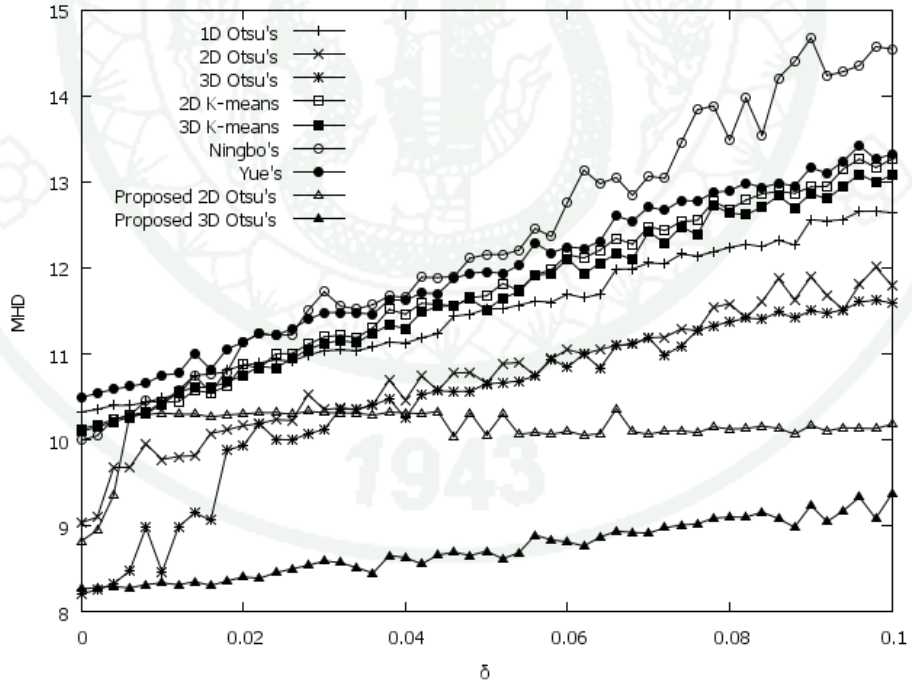


(b) MHD

Figure 18 Comparison of ME and MHD for the first test images with Gaussian noise added at various σ^2

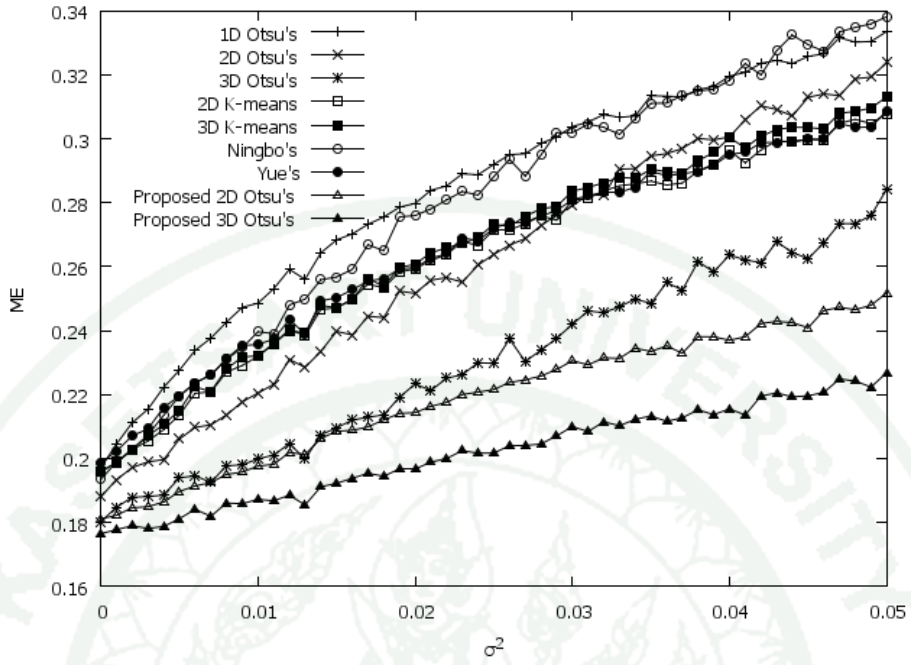


(a) ME

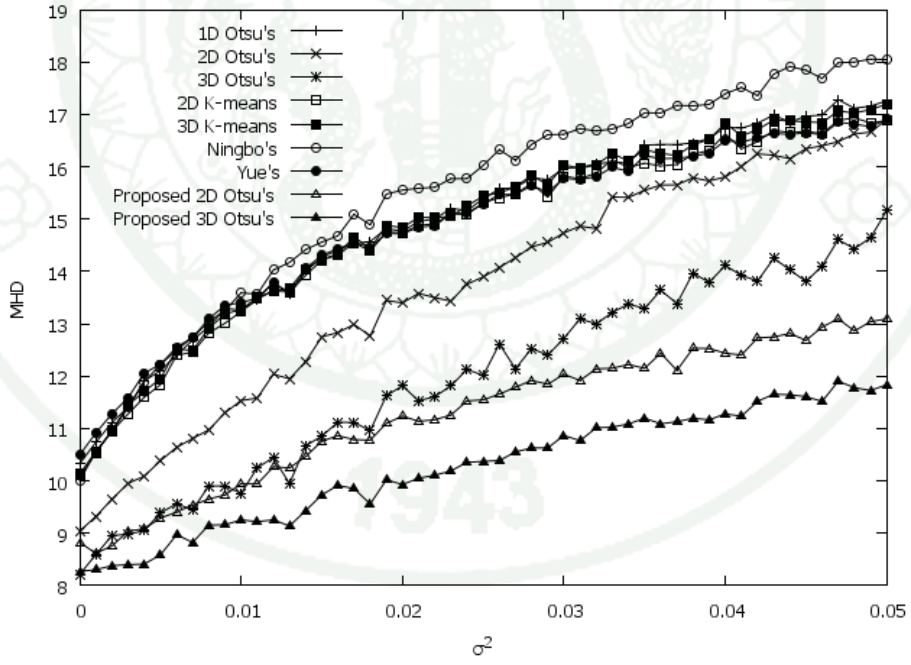


(b) MHD

Figure 19 Comparison of ME and MHD for the second test images with Salt&Pepper noise added at various δ



(a) ME



(b) MHD

Figure 20 Comparison of ME and MHD for the second test images with Gaussian noise added at various σ^2

Both test images show difficulties for thresholding when some amount of noise is added. Fig. 16(a) shows the histogram of the first test image that clearly presents bimodal, while Fig. 16(c) shows the histogram of the second test image that does not clearly presents bimodal. Fig. 16(b) and 16(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 segmented these 204 noise added images. We evaluated the performance of each method based on ME and MHD. Fig. 17 and 18 show the evaluation results of the first test images. Fig. 19 and 20 show the evaluation results of the second test images. The average computational time on all noise added images are shown in Table 2.

In the last experiment, we tested our method and the others with 200 real images from *Segmentation evaluation database* (Alpert *et al.*, 2007), where the ground truth of each image is provided. The average error measurements ($\overline{\mathsf{ME}}$ and $\overline{\mathsf{MHD}}$) and the average computational time ($\overline{\mathsf{T}}$) over 200 test images of each method are shown in Table 3. Fig. 21-26 show the sample thresholded images of each method. All segmentation results in this experiment is shown on *http://give.cpe.ku.ac.th/thresholding/MDT.php*

Table 3 $\overline{\mathrm{ME}}$, $\overline{\mathrm{MHD}}$, and $\overline{\mathrm{T}}$ over 200 real images

Methods	$\overline{\mathrm{ME}}$	$\overline{\mathrm{MHD}}$	$\overline{\mathrm{T}}$ (ms)
1D Otsu's	0.217102	19.545244	0.47
2D Otsu's	0.214391	19.580417	8.73
3D Otsu's	0.213022	19.569091	2275.07
2D K-means	0.228411	19.603090	15.61
3D K-means	0.228340	19.616353	1008.02
Ningbo's	0.214194	19.627783	8.28
Yue's	0.214962	19.408089	2.19
Proposed 2D Otsu's	0.212603	19.609405	2.89
Proposed 3D Otsu's	0.211341	19.199144	2.43

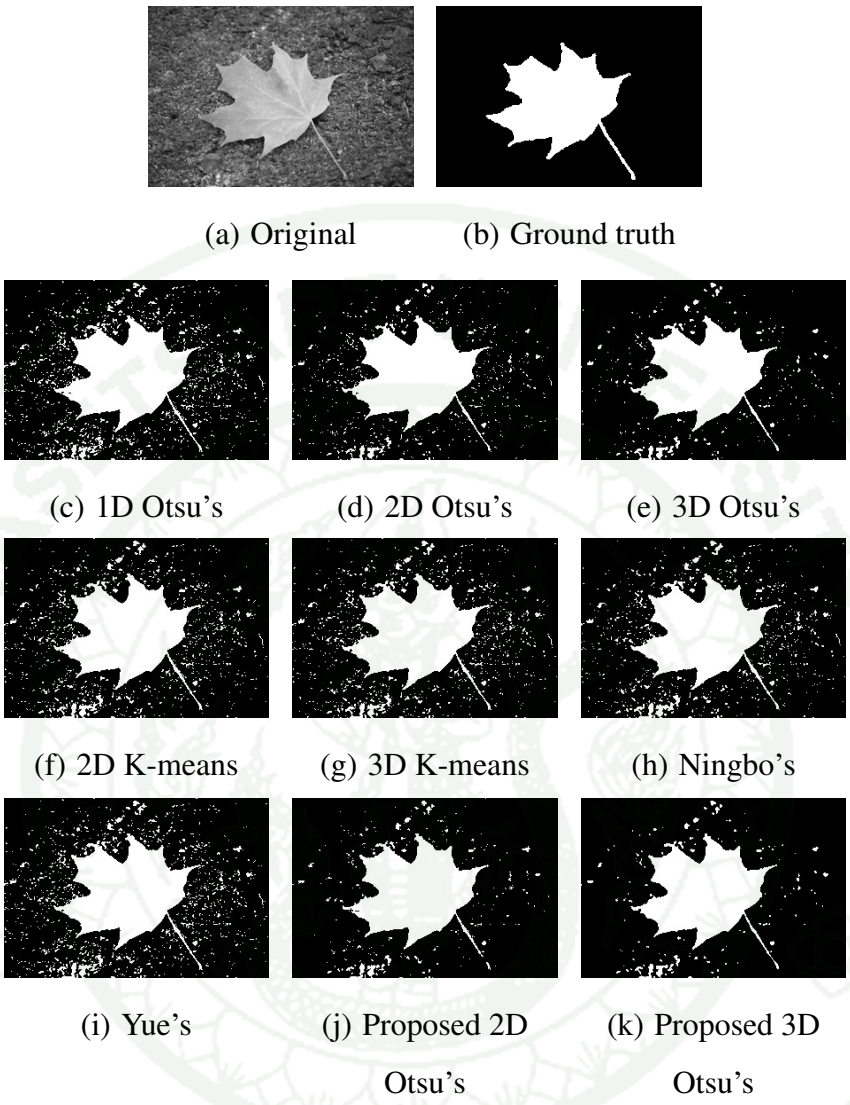


Figure 21 The first sample result images of each bi-level thresholding method

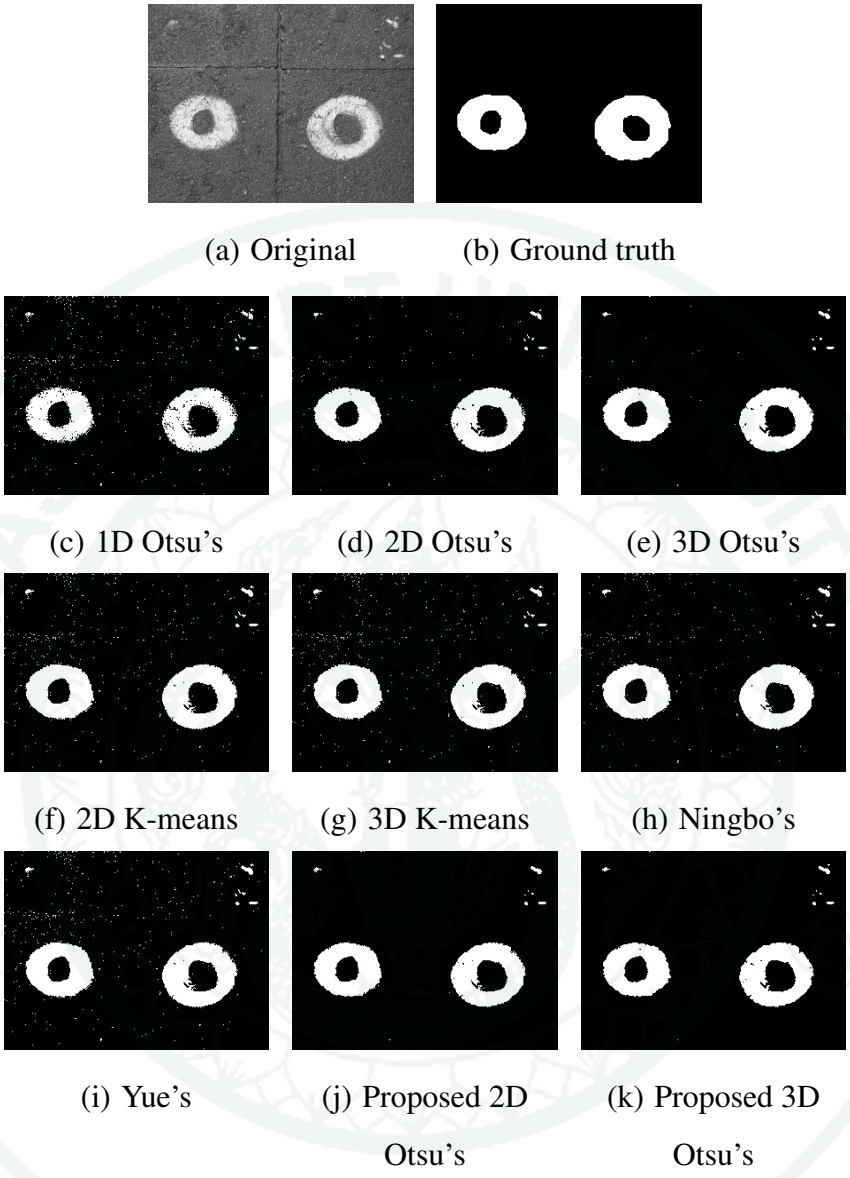


Figure 22 The second sample result images of each bi-level thresholding method

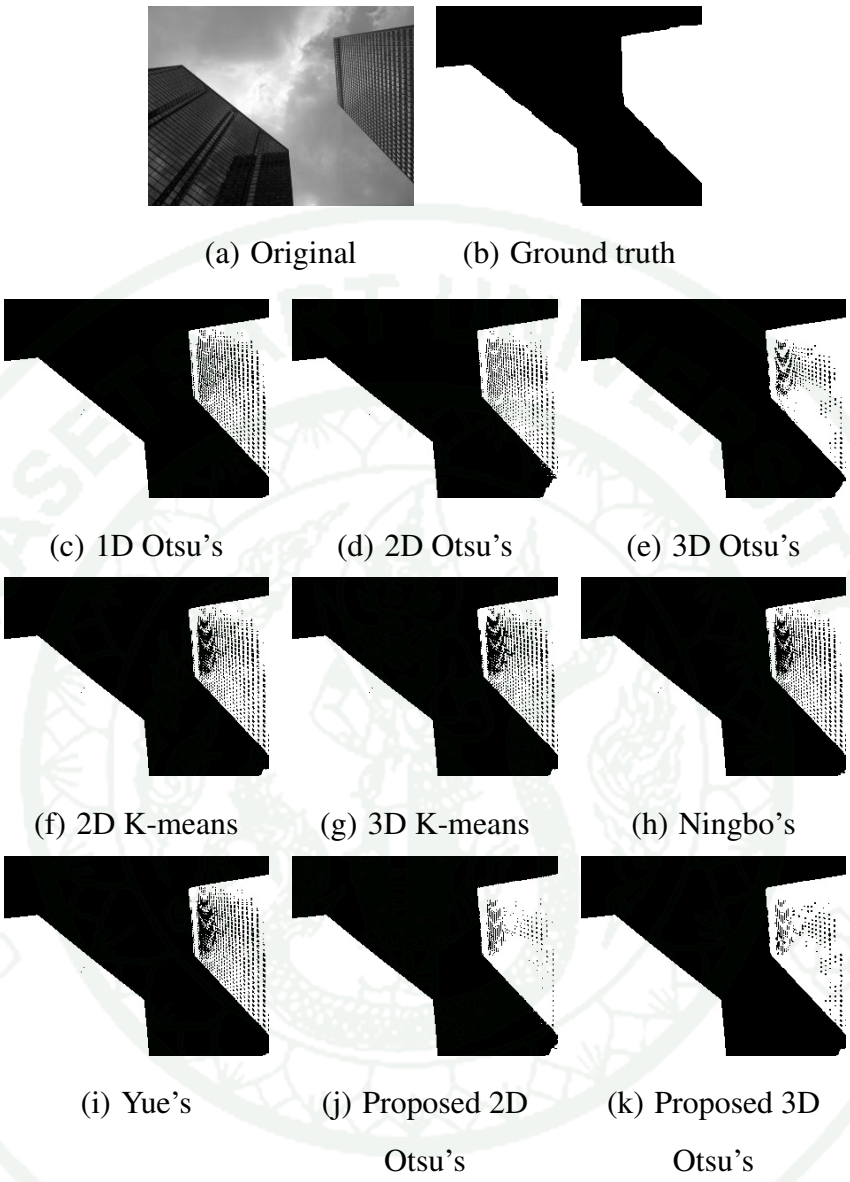


Figure 23 The third sample result images of each bi-level thresholding method

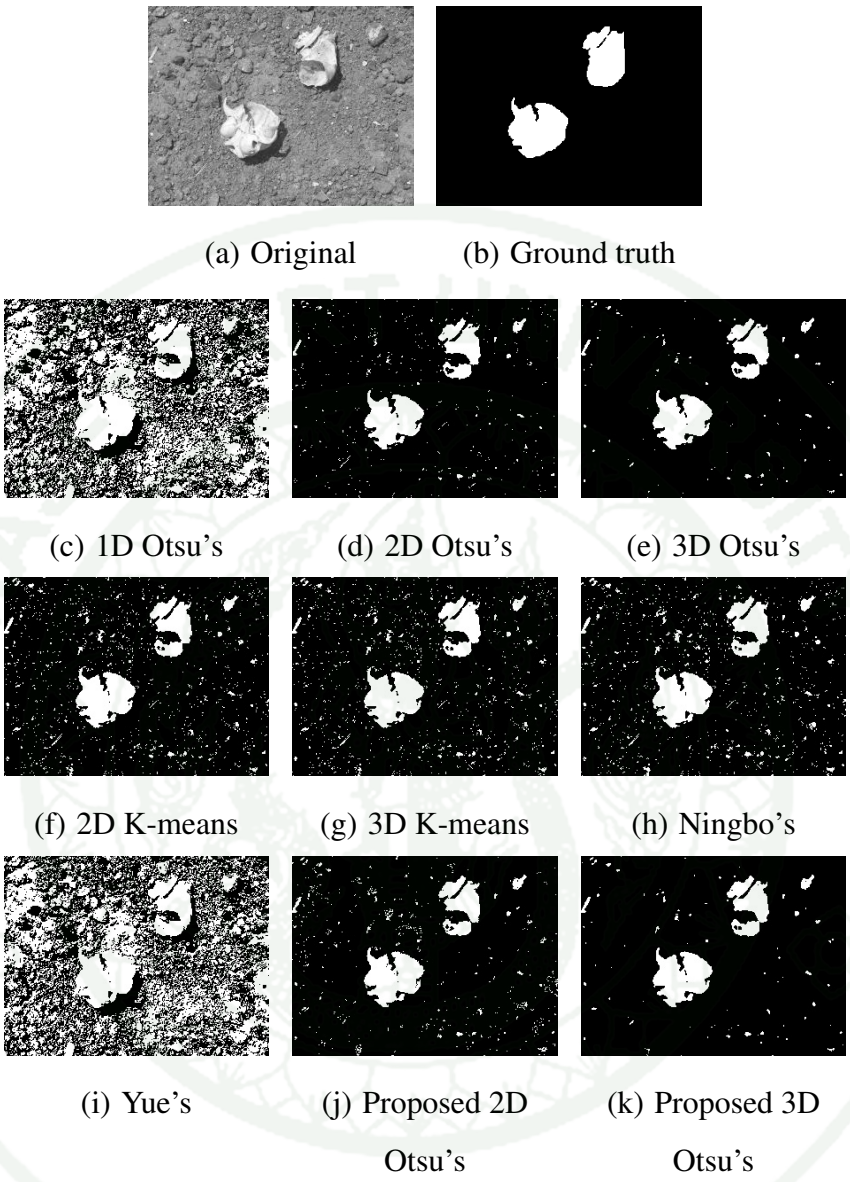


Figure 24 The forth sample result images of each bi-level thresholding method

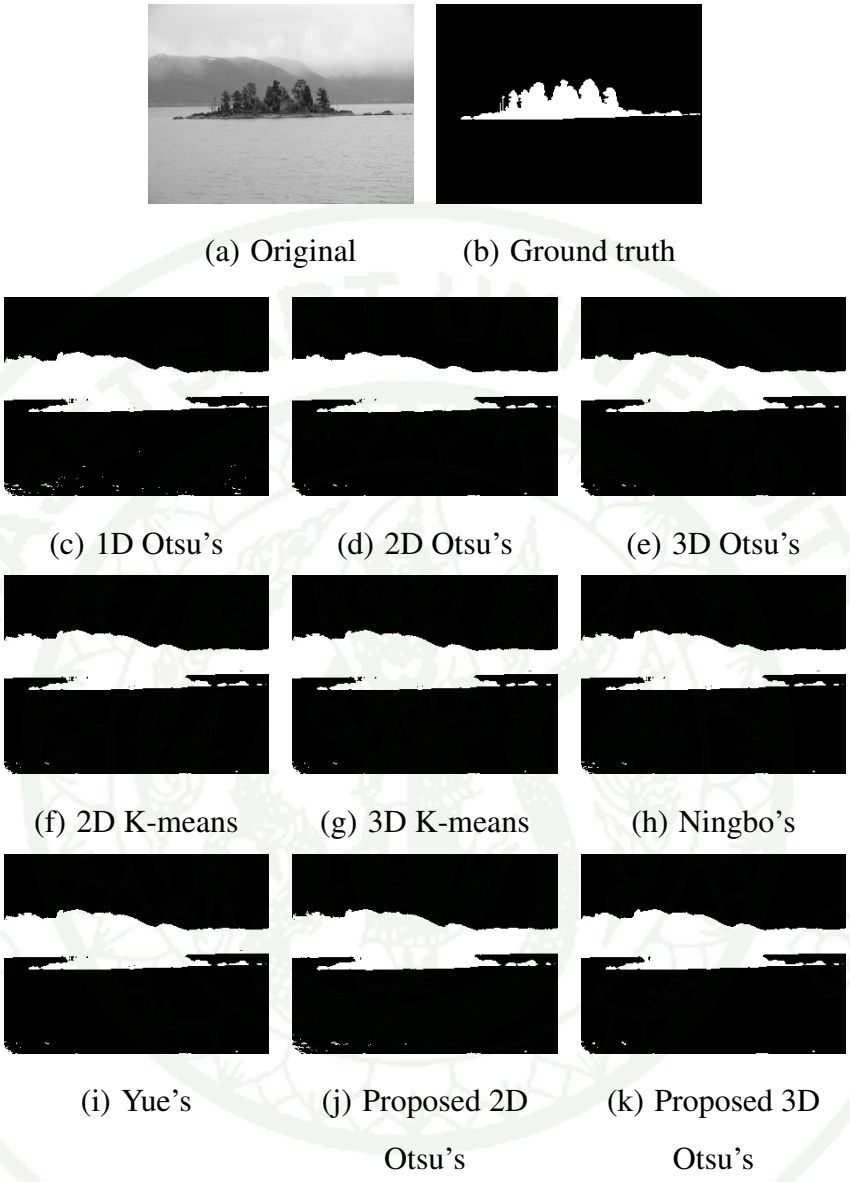


Figure 25 The fifth sample result images of each bi-level thresholding method

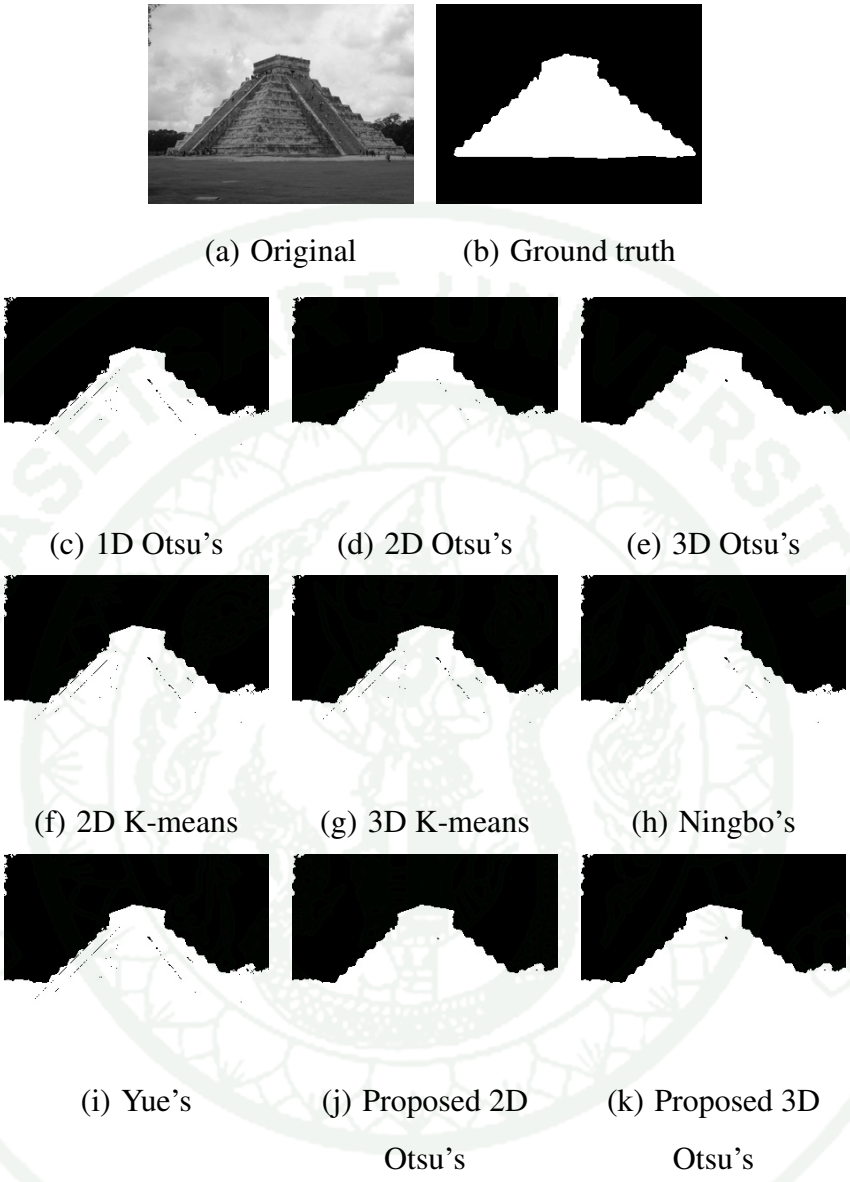


Figure 26 The sixth sample result images of each bi-level thresholding method

(a) Original (b) Ground truth

Figure 27 The sample image and its ground truth in the first test group

(a) Original (b) Ground truth

Figure 28 The sample image and its ground truth in the second test group

(a) Original (b) Ground truth

Figure 29 The sample image and its ground truth in the third test group

2. Results of multilevel and multi-dimensional Otsu's thresholding method

To evaluate our proposed multilevel method, we compared our method with Liao's method (Liao *et al.*, 2001), Eichmann's method (Eichmann and Lüssi, 2005), and K-means (Dongju and Jian, 2009) because they are multilevel methods that are based on Otsu's. There are two sets of experiments. The first one aims to test on real images with different complex conditions. The second one aims to test on noise robustness of each method.

In the first experiment, we selected the images from *Segmentation evaluation database* (Alpert *et al.*, 2007) and classify them into three test groups.

The first test group is used to test the performance of each method when the image pixels are classified into two classes . There are 49 images in this group which each image contains one object class and one background class. The sample image and its ground truth of this group are shown in Fig. 27. We use the ground truth that is provided in the database and use ME and MHD to measure the error between thresholded image and the corresponding ground truth. The average error measurements ($\overline{\mathsf{ME}}$ and $\overline{\mathsf{MHD}}$) and the average computational time ($\overline{\mathsf{T}}$) of each method are shown in Table 4. The sample thresholded images in the first test group of each method are shown in Fig. 30-33. All segmentation results of the first test group in this experiment is shown on *http://give.cpe.ku.ac.th/thresholding/MLT_11.php*

The second test group is used to test the performance of each method when the images are complex but they contain only one object class. There are 32 images in this group which each image contains one object class and many background classes that are ignored. We thus classify all background classes into only one class. The sample image and its ground truth of this group are shown in Fig. 28. We use the ground truth that is provided in the database, and we use ME and MHD to measure the error between thresholded image and the corresponding ground truth. The average error measurements ($\overline{\mathsf{ME}}$ and $\overline{\mathsf{MHD}}$) and the average computational time ($\overline{\mathsf{T}}$) of each method are shown in Table 5. The sample thresholded images in the second test group of each method are shown in Fig. 34-37. All segmentation results of the second test group in this experiment is shown on *http://give.cpe.ku.ac.th/thresholding/MLT_1D.php*

The third test group is used to test the performance of each method when the images are complex, but they can still be obviously separated into class regions. There are 28 images, where each image contains many classes. The sample image and its ground truth of this group are shown in Fig. 29. We generate the ground truth for the multilevel thresholding based on the ground truth that are provided in database and use modified ME to measure the error between thresholded image and the corresponding ground truth. The average error measurements ($\overline{\mathsf{MME}}$) and the average computational time ($\overline{\mathsf{T}}$) of each method for the third test group are shown in Table 6. The sample thresholded images in the third test group of each method are shown in Fig. 38-41. All segmentation results of the third test group in this experiment is shown on *http://give.cpe.ku.ac.th/thresholding/MLT_MM.php*

In the second experiment, we tested the robustness of each method in the presence of noise. We selected an images as our test image from the images of the third test group in the first experiment. Fig. 42(a) and 42(b) show the test image and its ground truth, respectively. We added noise to a 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 0.05. Fig. 42(c) and 42(d) show example noise added images of the test image.

Table 4 $\overline{\mathrm{ME}}$, $\overline{\mathrm{MHD}}$, and $\overline{\mathrm{T}}$ for the first test group

Multilevel methods	$\overline{\mathrm{ME}}$	$\overline{\mathrm{MHD}}$	$\overline{\mathrm{T}}$ (ms)
Liao's	0.063902	4.013095	2.84
Eichmann's	0.063902	4.013095	1.29
K-means	0.097367	6.753009	1.27
proposed	0.058984	3.441292	3.20

Table 5 $\overline{\mathrm{ME}}$, $\overline{\mathrm{MHD}}$, and $\overline{\mathrm{T}}$ for the second test group

Multilevel methods	$\overline{\mathrm{ME}}$	$\overline{\mathrm{MHD}}$	$\overline{\mathrm{T}}$ (ms)
Liao's	0.073685	6.936147	8778.28
Eichmann's	0.073685	6.936147	0.5
K-means	0.105336	11.048503	0.5
Proposed	0.068938	6.101892	3.88

Table 6 $\overline{\mathrm{MME}}$ and $\overline{\mathrm{T}}$ for the third test group

Multilevel methods	$\overline{\mathrm{MME}}$	$\overline{\mathrm{T}}$ (ms)
Liao's	0.169155	3.86
Eichmann's	0.169155	1.11
K-means	0.196328	0.54
Proposed	0.16149	3.93

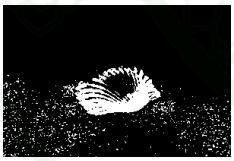

(a) Original (b) Ground truth




(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 30 The first sample result images in the first test group of each multilevel thresholding method


(a) Original (b) Ground truth




(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 31 The second sample result images in the first test group of each multilevel thresholding method

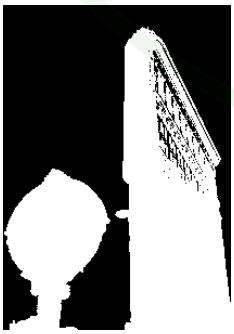
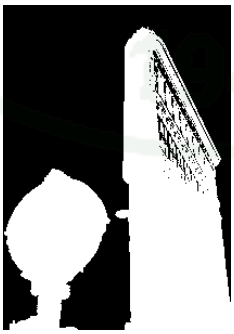
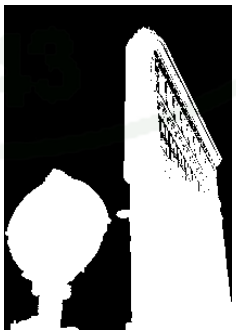
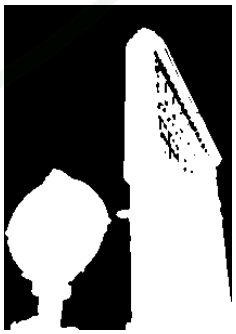

(a) Original (b) Ground truth




(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

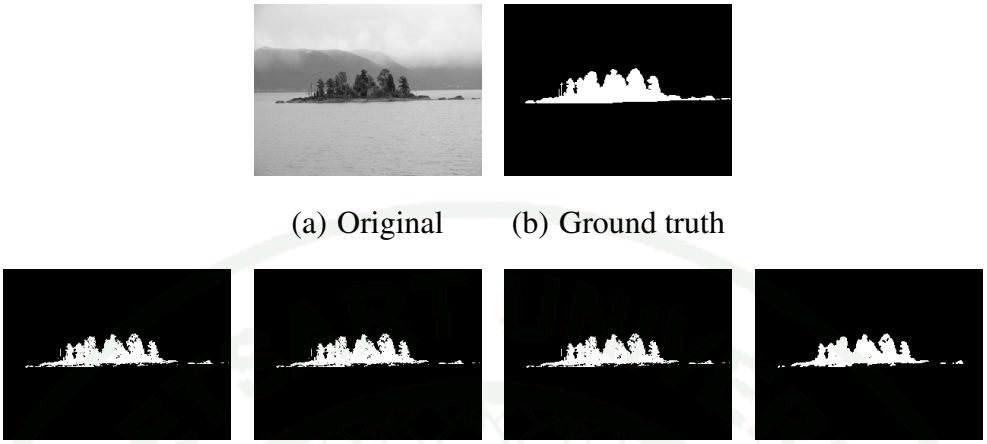
Figure 32 The third sample result images in the first test group of each multilevel thresholding method


(a) Original (b) Ground truth

(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

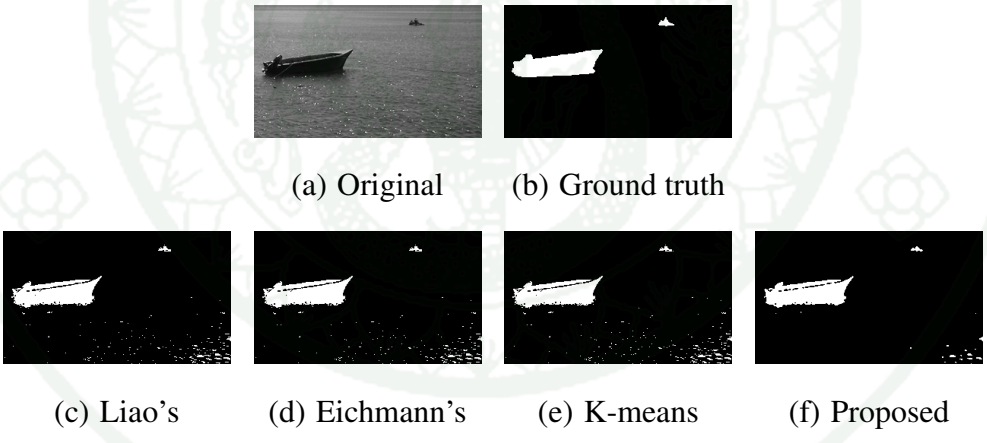
Figure 33 The forth sample result images in the first test group of each multilevel thresholding method



(a) Original (b) Ground truth

(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

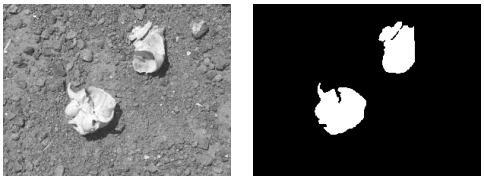
Figure 34 The first sample result images in the second test group of each multilevel thresholding method



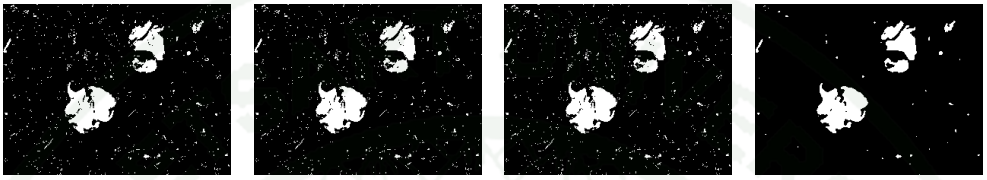
(a) Original (b) Ground truth

(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 35 The second sample result images in the second test group of each multi-level thresholding method

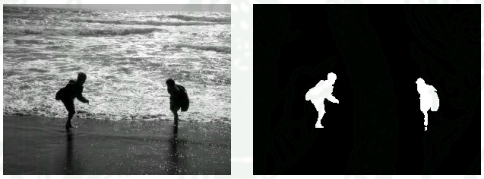


(a) Original (b) Ground truth



(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 36 The third sample result images in the second test group of each multilevel thresholding method



(a) Original (b) Ground truth

(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 37 The forth sample result images in the second test group of each multilevel thresholding method


(a) Original (b) Ground truth




(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 38 The first sample result images in the third test group of each multilevel thresholding method


(a) Original (b) Ground truth




(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 39 The second sample result images in the third test group of each multilevel thresholding method


(a) Original (b) Ground truth




(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

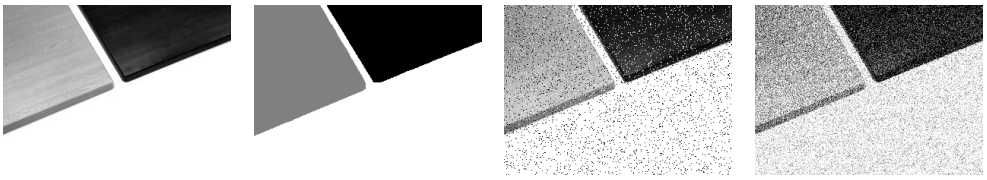
Figure 40 The third sample result images in the third test group of each multilevel thresholding method


(a) Original (b) Ground truth




(c) Liao's (d) Eichmann's (e) K-means (f) Proposed

Figure 41 The forth sample result images in the third test group of each multilevel thresholding method



(a) Original (b) Ground truth (c) Salt&Pepper ($\delta = 0.1$) (d) Gaussian ($\sigma^2 = 0.05$)

Figure 42 The image set with sample noise added images for multilevel thresholding

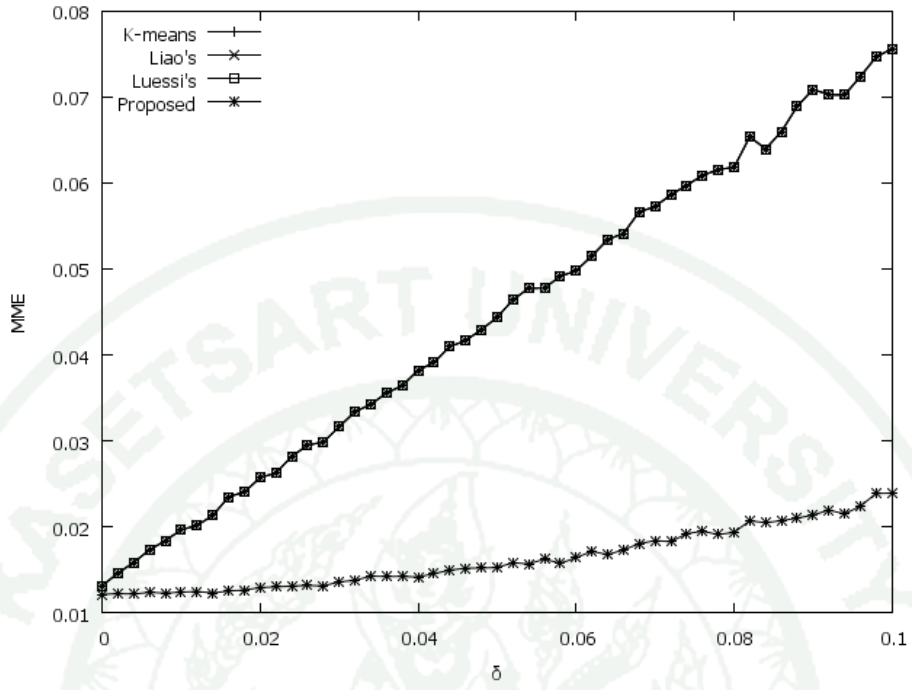
We segmented these 102 noise added images. We evaluated the performance of each method based on MME. Fig. 43(a) and 43(b) show the evaluation results of the test images.

Discussion

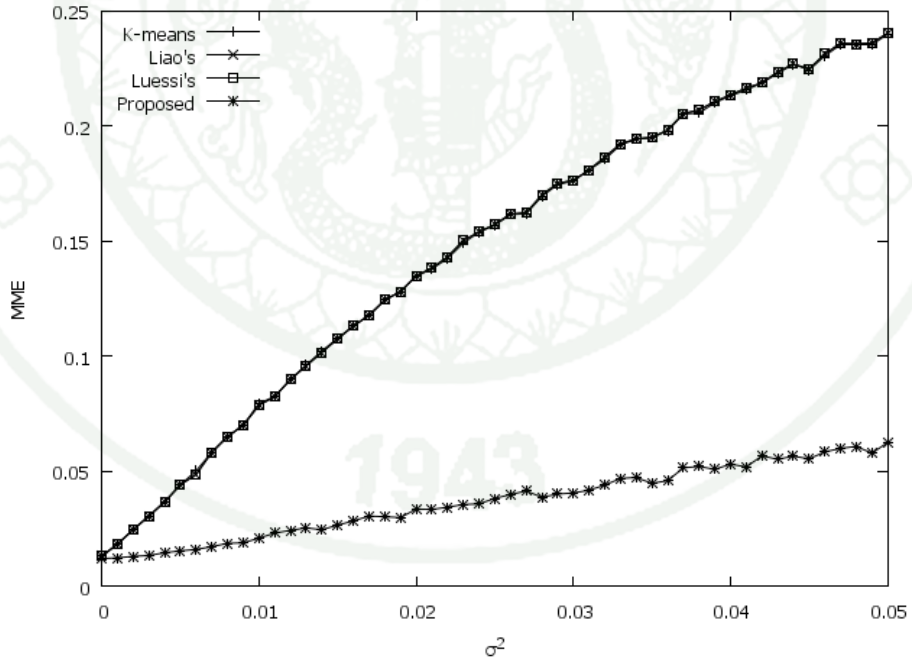
1. Discussion of improving multi-dimensional Otsu's thresholding method

From Table 1, it can be seen that the optimal threshold of the proposed methods is close to the optimal threshold of the other methods, where the second element of optimal threshold of the proposed 2D Otsu's method is the gradient value.

From the evaluation results in the presence of Salt&Pepper noise shown in Fig. 17 and 19, both ME and MHD values of our methods (both 2D and 3D) are lower than those of the other methods except MHD values on the first test images, MHD values of our methods are higher than the 3D K-means method. The thresholding results of the 3D K-means and our methods are shown in Fig. 44. 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 2D and 3D methods give lower number of mistaken pixels in the object region, and higher number of pixels in the background region. MHD of our methods are thus higher than that of the 3D K-means method.

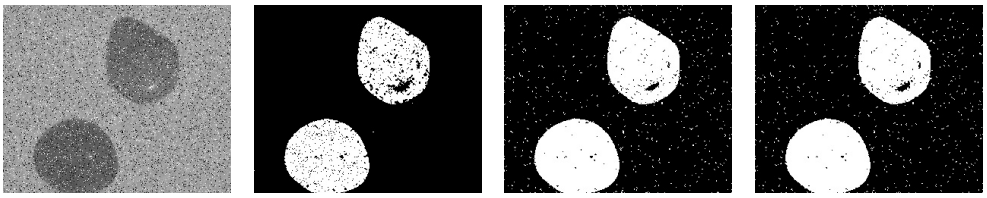


(a) Salt&Pepper noise added at various δ



(b) Gaussian noise added at various σ^2

Figure 43 Comparison of MME for each multilevel thresholding method of the test images



(a) Original (b) 3D K-means (c) Proposed 2D Otsu's (d) Proposed 3D Otsu's

Figure 44 The original image, thresholded image of 3D K-means, and thresholded images of both proposed 2D and 3D methods when δ=0.1

From the evaluation results in the presence of Gaussian noise shown in Fig. 18 and 20, both ME and MHD values of our both 2D and 3D methods on the second test images are lower than those of the other methods; while those values of our 2D method are higher than our 3D method. ME and MHD values of our 3D method on the first test images are very close to those of the 3D K-means method and lower than those of the other methods except Yue's method. ME and MHD values of our 2D method on the first test images are higher than those of the other methods except the 1D Otsu's, 2D Otsu's, 3D Otsu's, and Ningbo's methods.

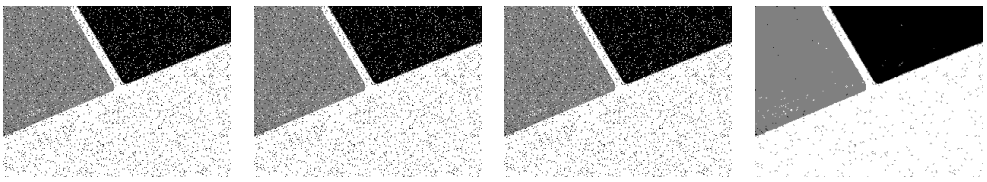
From Table 2, it can be seen that our methods performs faster than the other methods except the 1D Otsu's and Yue's methods. Our 3D method always gives low error measurements in both classification and shape evaluations.

From the results in Table 3, it can be seen that the average computational time of our methods are lower than that of the other methods except the 1D Otsu's and Yue's methods. The average ME value of our 2D method is lower than that of the other methods, however, the average MHD value is higher than that of the other methods, except the 3D K-means and Ningbo's methods. Since we replace the original gray level with the average gray level in our method; the number of pixels that wrongly assigned over the whole image is reduced, but the number of pixels that wrongly assigned on edges is slightly increased. The average ME and MHD values of our 3D method is lower than that of the other methods. It indicates that our 3D method shows the best matching of the object and the background, and also gives the smallest amount of shape distortion. From Fig. 25-26, we can see the limitation of bi-level thresholding method when the gray level of object is nearly the same as the gray level of background.

2. Discussion of multilevel and multi-dimensional Otsu's thresholding method

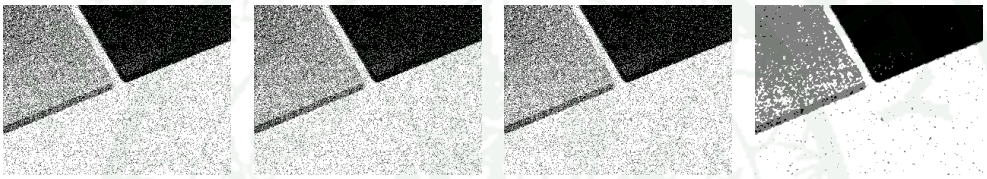
From Table 4, 5, and 6, we can see that our proposed method has lower $\overline{\text{ME}}$ and $\overline{\text{MHD}}$ than other methods. While $\overline{\text{T}}$ is more than that of other methods except Liao's method in Table 5 because our proposed method has to create mean-filtered and median-filtered images and use Eichmann's method three times to select the optimal thresholds. However, it can be seen that our proposed method gives low error measurements in both classification and shape evaluations.

From Fig. 43, MME of our proposed method is lower than the other methods for both Salt&Pepper and Gaussian noise. Fig. 45 and 46 are the thresholding results of each method for Fig. 42(c) and 42(d), respectively. It indicates that our method is robust to noise than the other methods. This is because our method is the combined method of multilevel and multi-dimensional methods, which multi-dimensional method can resist to noise.



(a) Liao's (b) Eichmann's (c) K-means (d) Proposed

Figure 45 The thresholding results of each multilevel thresholding method for Fig. 42(c)



(a) Liao's (b) Eichmann's (c) K-means (d) Proposed

Figure 46 The thresholding results of each multilevel thresholding method for Fig. 42(d)

CONCLUSION AND RECOMMENDATIONS

Conclusion

We proposed the improved multi-dimensional Otsu's methods including 2D and 3D methods that can overcome the shortcoming of 1D, 2D, and 3D Otsu's methods. For the proposed 2D method, it takes advantage from the characteristic of the gradient gray level which expresses the difference between the original gray level and the average gray level of pixels to divide a 2D histogram into meaningful regions. The projections of the 2D histogram are used to reduce the search space for the optimal threshold and the 1D Otsu's method is used to select the optimal threshold. For the proposed 3D method, we show that the threshold selection in 3D Otsu's method can be approximated to three threshold selections of the 1D Otsu's method. Hence, we can use three 1D histograms instead of one 3D histogram in the threshold selection, which is similar to project the 3D histogram into its three main axes. The proposed method computes 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 methods on real images and images with noise added. The results show that our methods are more resist to noise and spend less computational time than the other multi-dimensional methods. Visually, our methods give better or comparable results than the other multi-dimensional methods.

We then extend an improved multi-dimensional Otsu's method to a multilevel with multi-dimensional Otsu's method. We replaced the bi-level Otsu's method in the proposed 3D method with Eichmann's method, which is a multilevel Otsu's method. Eichmann's method uses the trellis structure to implement the shortest path algorithm to find the optimal threshold and uses SMAWK algorithm to find the optimal path of nodes in each stage.

We tested our multilevel method on real images, which contain both simple and complex images, and we also tested our method with noise added images. The results show that our proposed multilevel method can analyse complex images and is still robust against noise. However, the computational time of our method is increased because of the additional computation for mean-filtered and median-filtered images. We can see that this method takes advantage of noise resistance from the multi-dimensional method and also takes advantage of a complex image analysis from the multilevel method.

Recommendations

Our methods are inappropriate to threshold images, which have very small objects such as point and thin line, because the objects may be considered as noise and will be removed. In the future work, there are some entropy based methods which are extended to multi-dimensional methods and are extended to multilevel methods as Otsu's. They may be combined to multilevel and multi-dimensional entropy based method because they have the same properties as Otsu's such as the relation of region or volume division as (17)-(18) and (26)-(27), the class cost of the method that fulfills the convex quadrangle inequality, the use of SMAWK algorithm, etc. Moreover, we will extend the combined method to determine the suitable number of classes automatically instead of acquiring the number of classes from the user.

We apply our multilevel method in a real application, *Economical Multiple-Choice Exam Processing and Management System*. Compare to the Otsu's thresholding method, our method gives more accurate results for answer checking. For each answer sheet, we classify pixels into three classes: the first class is the pixels from printed ink, which their gray levels are close to black, the second class is the pixels of the background, which their gray levels are close to white, and the third class is the colored pixels by students, which their gray levels are valued between the gray level of first and second classes. We use the first and third classes of pixels to determine whether a choice is selected or not. In addition to this system, our multilevel method can be used in the other kinds of document analysis systems that the gray levels in images have to separate into several classes or when the background contains noise.

LITERATURE CITED

Alpert, S., M. Galun, R. Basri, and A. Brandt. 2007. Image Segmentation by Probabilistic Bottom-Up Aggregation and Cue Integration, pp. 1–8. In **Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on**. 17-22 June 2007.

Changhua, L., Z. Ping, and C. Yong. 2010. The segmentation algorithm of improvement a two-dimensional Otsu and application research, pp. V1–76–V1–79. In **Software Technology and Engineering (ICSTE), 2010 2nd International Conference on**. 3-5 Oct. 2010.

Chen, W., L. Cao, J. Qian, and S. Huang. 2010a. A 2-phase 2-D thresholding algorithm. **Digital Signal Processing**. 20 (6): 1637–1644.

Chen, Y., D.-r. Chen, Y. Li, and L. Chen. 2010b. Otsu's thresholding method based on gray level-gradient two-dimensional histogram, pp. 282–285. In **Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on**. 6-7 March 2010.

Dongju, L. and Y. Jian. 2009. Otsu Method and K-means, pp. 344–349. In **Hybrid Intelligent Systems, 2009. HIS '09. Ninth International Conference on**. 12-14 Aug. 2009.

Eichmann, M. and M. Lüssi. 2005. **Efficient Multilevel Image Thresholding**. Dipl. Ing. FH Thesis, Northwestern University.

Gong, J., L. Li, and W. Chen. 1998. Fast recursive algorithms for two-dimensional thresholding. **Pattern Recognition**. 31 (3): 295–300.

Hammouche, K., M. Diaf, and P. Siarry. 2010. A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem. **Engineering Applications of Artificial Intelligence**. 23 (5): 676–688.

Hao, G., X. Wenbo, S. Jun, and T. Yulan. 2010. Multilevel Thresholding for Image Segmentation Through an Improved Quantum-Behaved Particle Swarm Algorithm. **Instrumentation and Measurement, IEEE Transactions on**. 59 (4): 934–946.

Horng, M.-H. 2010. A multilevel image thresholding using the honey bee mating optimization. **Applied Mathematics and Computation**. 215 (9): 3302–3310.

Huang, D.-Y. and C.-H. Wang. 2009. Optimal multi-level thresholding using a two-stage Otsu optimization approach. **Pattern Recognition Letters**. 30 (3): 275–284.

Jing, X. J., J. F. Li, and Y. L. Liu. 2003. Image segmentation based on 3-D maximum between-cluster variance. **Tien Tzu Hsueh Pao/Acta Electronica Sinica**. 31 (9): 1281–1285.

Liao, P.-s., T.-s. Chen, and P.-c. Chung. 2001. A fast algorithm for multilevel thresholding. **Journal of Information Science and Engineering**. 17: 713–727.

Liu, J., W. Li, and Y. Tian. 1991. Automatic thresholding of gray-level pictures using two-dimension Otsu method, pp. 325–327 vol.1. In **Circuits and Systems, 1991. Conference Proceedings, China., 1991 International Conference on**.

Nabizadeh, S., K. Faez, S. Tavassoli, and A. Rezvanian. 2010. A novel method for multi-level image thresholding using particle swarm Optimization algorithms, pp. V4–271–V4–275. In **Computer Engineering and Technology (ICCET), 2010 2nd International Conference on**. 16-18 April 2010.

Ningbo, Z., W. Gang, Y. Gaobo, and D. Weiming. 2009. A Fast 2D Otsu Thresholding Algorithm Based on Improved Histogram, pp. 1–5. In **Pattern Recognition, 2009. CCPR 2009. Chinese Conference on**.

Otsu, N. 1979. A Threshold Selection Method from Gray-Level Histograms. **Systems, Man and Cybernetics, IEEE Transactions on**. 9 (1): 62–66.

Qidan, Z., J. Liqiu, and B. Rongsheng. 2010. One-dimensional threshold average decomposition for two-dimensional Ostu algorithm, pp. 2783–2788. In **Control and Decision Conference (CCDC), 2010 Chinese**. 26-28 May 2010.

Ramírez-Ortegón, M. A., E. Tapia, R. Rojas, and E. Cuevas. 2010. Transition thresholds and transition operators for binarization and edge detection. **Pattern Recognition**. 43 (10): 3243–3254.

Raut, S., M. Raghuvanshi, R. Dharaskar, and A. Raut. 2009. Image Segmentation – A State-Of-Art Survey for Prediction, pp. 420–424. In **Advanced Computer Control, 2009. ICACC '09. International Conference on**. 22-24 Jan. 2009.

Sahoo, P. K., S. Soltani, and A. K. C. Wong. 1988. A survey of thresholding techniques. **Computer Vision, Graphics, and Image Processing**. 41 (2): 233–260.

Sezgin, M. and B. Sankur. 2004. Survey over image thresholding techniques and quantitative performance evaluation. **Journal of Electronic Imaging**. 13 (1): 146–168.

Shu-Chien, H. 2009. A Multilevel Thresholding Method Based on Cross Entropy and Genetic Algorithms, pp. 385–388. In **Innovative Computing, Information and Control (ICICIC), 2009 Fourth International Conference on**. 7 - 9 December 2009.

Sthitpattanapongsa, P. and T. Srinark. 2011. A two-stage Otsu's thresholding based method on a 2D histogram, pp. 345–348. In **Intelligent Computer Communication and Processing (ICCP), 2011 IEEE International Conference on**. 25-27 Aug. 2011.

Sthitpattanapongsa, P. and T. Srinark. 2012. An Equivalent 3D Otsu's Thresholding Method, Advances in Image and Video Technology. **Lecture Notes in Computer Science**. 7087: 358–369.

Wang, L., H. Duan, and J. Wang. 2008. A fast algorithm for three-Dimensional Otsu's Thresholding method, pp. 136–140. In **IT in Medicine and Education, 2008. ITME 2008. IEEE International Symposium on**. 12-14 Dec. 2008.

Wang, N., X. Li, and X.-h. Chen. 2010. Fast three-dimensional Otsu thresholding with shuffled frog-leaping algorithm. **Pattern Recognition Letters**. 31 (13): 1809–1815.

Yue, F., W. M. Zuo, and K. Q. Wang. 2009. Decomposition based two-dimensional threshold algorithm for gray images. **Zidonghua Xuebao/ Acta Automatica Sinica**. 35 (7): 1022–1027.

CURRICULUM VITAE

NAME : Mr. Puthipong Sthitpattanapongsa

BIRTH DATE : November 24, 1988

BIRTH PLACE : Bangkok, Thailand

EDUCATION :

YEAR	**INSTITUTE**	**DEGREE/DIPLOMA**
2006	Kasetsart Univ.	B.Eng. (Computer)

POSITION/TITLE : -

WORKPLACE : -

SCHOLARSHIP/AWARDS : Teaching Assistant Scholarship from the Department of Computer Engineering, Faculty of Engineering, Kasetsart University