Multilevel Otsu’s thresholding method with an equivalent 3D Otsu’s 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

Otsu’s method is considered as one of the most popular thresholding method due to its simplicity. In order to increase its capabilities, one approach is to additionally use multidimensional information so that noise in images can be handled; while the other approach is to allow multilevel thresholding instead of bilevel thresholding to extract objects in the case of complex images. These two approaches distinctly offer different advantages, where real world images mostly contain both kinds of image difficulties. In this paper, we propose the use of both multidimensional information and multilevel thresholding together, so that we can efficiently cope with real world images. From the experiments, even though our method requires a bit longer execution time, the method is more resist to noise and also give better or comparable results than the others.

1. Introduction

Otsu’s method [10] is one of the most popular thresholding method. This method is extended to multi-dimensional methods such as two-dimensional (2D) Otsu’s method [7] and three-dimensional (3D) Otsu’s method [5]. Multi-dimensional methods consider not only gray levels of image pixels but also the spatial information of image pixels, which are the mean and the median values of pixels and their neighbourhood. These methods thus give satisfactory results when images contain noise. However, these traditional multi-dimensional methods take a long execution time due to the number of dimensions of the histogram that is increased. Gong et al. [4] proposed a fast recursive 2D Otsu’s method. Ningbo et al. [9] used a histogram projection and look-up tables for searching of the optimal threshold. Yue et al. [14] proposed a decomposition of the 2D Otsu’s method. Chen et al. [2] used a gray level-gradient histogram instead of a gray level-average gray level histogram in the 2D Otsu’s method. Sthitpattanapongsa and Srinark [12] used the gradient value to divide the histogram and used a histogram projection to reduce the search space of the optimal threshold. Sthitpattanapongsa and Srinark [13] showed that the calculation of the 3D Otsu’s method can be approximated by three calculations of the traditional 1D Otsu’s method.

In the other way, Otsu’s method is extended to support multiple levels instead of only two levels. The method selects several thresholds to classify the pixels into several classes, thus it can handle complicated images. Similar to multi-dimensional methods, this multilevel method takes a long execution time due to the number of classes that is increased. Liao et al. [6] modified the between-class variance for more efficient computation. They used a look-up table to store the value of all possible pairs of gray levels to reduce the searching time for the optimal thresholds. Luessi and Eichmann [8] used the shortest path algorithm and the matrix searching algorithm to find the optimal thresholds. Dongju and Jian [3] proved that the objective function of K-means is equivalent to that of Otsu’s and its extended methods for both bi-level and multilevel thresholding. K-means thus can be modified to serve 2D and 3D thresholding as well as multilevel thresholding.

Multi-dimensional and multilevel methods offer distinguished advantages, however, they have never been put together. In this paper, we propose the thresholding method that combines these two methods together in order to get full benefits from both methods.

2. Equivalent 3D Otsu’s method

Given an image \( f(x, y) \) represented by \( L \) gray levels. Let \( g(x, y) \) and \( h(x, y) \) be mean-filtered and median-filtered images, respectively. The method [13] selects three optimal thresholds from three 1D histograms of \( f(x, y), g(x, y), \) and \( h(x, y) \), instead of selecting the
optimal threshold vector from one 3D histogram. For each histogram, the 1D Otsu’s method is used to select an optimal threshold. Let \( s^*, t^*, \) and \( q^* \) be the optimal thresholds of \( f(x, y) \), \( g(x, y) \), and \( h(x, y) \), respectively. For each pixel \( (x, y) \), \( s^*, t^*, \) and \( q^* \) are used to classify \( f(x, y) \), \( g(x, y) \), and \( h(x, y) \), respectively. The most selected class by each threshold is then used as a resulting class of that pixel. The method thus requires less execution time because the searching space for thresholds is reduced from \( L \times L \times L \) (one 3D histogram) to \( L + L + L \) (three 1D histograms); and the method is still robust against noise.

3. Proposed method

There are two approaches to combine the multi-level method with the multi-dimensional method. The first approach is to extend a one-dimensional multilevel method to either two-dimensional or three-dimensional multilevel methods. However, this approach is not efficient because the computation of thresholds cannot take advantage of the recursive structure of the objective function [6]. The second approach is to project either 2D or 3D histograms in a multi-dimensional method to form 1D histograms, and then apply a 1D multilevel method on them. The second approach is obviously more efficient than the first one because using histogram projections can reduce search space in the multi-dimensional method. Moreover, it can easily be modified to a multilevel method without difficulty. We thus chose the second approach for merging of the multilevel method and the multi-dimensional method.

Given an image \( f(x, y) \) represented by \( L \) gray levels, the mean and the median of gray values of pixels in the \( k \times k \) neighbourhood regions centered at the coordinate \( (x, y) \) are denoted as \( g(x, y) \) and \( h(x, y) \), respectively. In this paper, we use \( k = 3 \). Let \( M \) be the number of classes. Our method selects the optimal thresholds from three 1D histograms instead of one 3D histogram [13]. Three 1D histograms are the histograms of \( f(x, y) \), \( g(x, y) \), and \( h(x, y) \). Our method uses Luessi’s method [8], which is a multilevel method, instead of using the traditional Otsu’s method [10] to select the optimal thresholds from each histogram.

Luessi’s method applies the shortest path algorithm to find multiple optimal thresholds in a 1D histogram. The trellis structure is used to implement the algorithm. The problem of finding the optimal path in one stage of the trellis structure is equivalent to the problem of finding the row wise maxima in a lower triangular matrix or a search matrix. This search matrix is totally monotone because the class cost of Otsu’s method fulfills the convex quadrangle inequality. This method thus uses the SMAWK algorithm, which is an efficient algorithm for finding the row wise maxima in totally monotone matrices. The time complexity of the method to search for the optimal thresholds is \( O(ML) \).

For each image, there are \( M - 1 \) thresholds to classify all pixels into \( M \) classes. Let \( s_1^*, s_2^*, \ldots, s_{M-1}^* \) be the optimal thresholds of \( f(x, y) \); \( t_1^*, t_2^*, \ldots, t_{M-1}^* \) be the optimal thresholds of \( g(x, y) \); and \( q_1^*, q_2^*, \ldots, q_{M-1}^* \) be the optimal thresholds of \( h(x, y) \). For each pixel \( (x, y) \), \( s_1^*, s_2^*, \ldots, s_{M-1}^* \) is used to classify \( f(x, y) \), \( t_1^*, t_2^*, \ldots, t_{M-1}^* \) is used to classify \( g(x, y) \), and \( q_1^*, q_2^*, \ldots, q_{M-1}^* \) is used to classify \( h(x, y) \). Now, we have three classification results for a pixel \( (x, y) \). However, we cannot use the mostly selected class from all classification results as a resulting class of that pixel because all classification results can be different from each other. Therefore, we use the median class from all classification results as a resulting class of that pixel.

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 noise 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 (\( \delta \)), the probability of swapping a pixel. Gaussian noise is represented by mean (\( \mu \)) and variance (\( \sigma^2 \)). In our experiments, we used only \( \mu = 0 \).

We compared our method with Liao’s method [6], Luessi’s method [8], and K-means [3] because they are multilevel methods which are based on Otsu’s.

In the first experiment, we selected a set of images from Segmentation evaluation database [1] and classify them into three test groups. The first test group is used to test the performance of each method when image pixels are classified into two classes. The second test group is used to test the performance of each method when images are complex but they have only one object class. For the first and second test groups, we used the ground truth that is provided in database and used ME and MHD to measure the classification and shape differences between each resulting image and its corresponding ground truth. ME and MHD are defined in [11]. The average error measurements (\( \text{ME} \) and \( \text{MHD} \)) and the average computational time (\( \bar{T} \)) of each method for the first and second test groups are shown in Table 1 and Table 2, respectively. All segmentation results of the first and second test groups in this experiment are shown on http://give.cpe.ku.ac.th/thresholding/MLT_11.php and
We used the modified ME on the ground truth that are provided in the database. The modified ME is defined as

\[
\text{ME} = 1 - \frac{\sum_{k=0}^{M} |C_{kO} \cap C_{kT}|}{\sum_{k=0}^{M} |C_{kO}|}
\]  

where \(C_{ki}\) denotes the pixels of class \(k\) of an image \(i\), which includes the ground truth \(O\) and thresholded \(T\) images. \(|\cdot|\) is the cardinality of the set. \(M\) is the number of classes. The average error measurements (\(\overline{\text{ME}}\)) and the average computational time (\(\overline{T}\)) of each method for the third test group are shown in Table 3. All segmentation results of the third test group in this experiment is shown on http://give.cpe.ku.ac.th/thresholding/MLT_MM.php.

From Table 1, 2, and 3, it can be seen that the proposed method has lower \(\overline{\text{ME}}\) and \(\overline{\text{MHD}}\) than the other methods. While \(\overline{T}\) is more than that of the other methods except for Liao’s method in Table 2. This is because the proposed method has to create mean-filtered and median-filtered images and use Luessi’s method three times to select the optimal thresholds. However, the proposed method gives the lowest error measurements in both classification and shape evaluations.

In the second experiment, we tested the robustness of each method in the presence of noise. We selected a test image from the images in the third test group, which is in the first experiment. Figure 1(a) and 1(b) show the test image and its ground truth, respectively. We added noise to the test image to generate new 51 images with Salt&Pepper noise using \(\delta\) that are vary from 0 to 0.1, and the other 51 images with Gaussian noise using \(\sigma^2\) that are vary from 0 to 0.01. Figure 1(c) and 1(d) show example noise added images of the test image. We segmented these 102 noise added images. We evaluated the performance of each method based on modified ME. Figure 2(a) and 2(b) show the evaluation results of the test images.

From Figure 2, the modified ME of the proposed method is lower than the other methods for both Salt&Pepper and Gaussian noise tests. Figure 3 and 4 are the thresholding results of each method for Figure 1(c) and 1(d), respectively. It indicates that our method is robust to noise than the other methods.

5. Conclusion

We combine a multi-dimensional Otsu’s method with a multilevel Otsu’s method. We replace bi-level Otsu’s method in [13] with Luessi’s method which is a multilevel Otsu’s method.

We tested our method on real images, which are both simple and complex images, and images with noise added. The results show that our method can analyse complex images and still robust against noise. However, the computational time of our method is increased because of the computational time on mean-

### Table 1. ME, MHD, and T for the first test group

<table>
<thead>
<tr>
<th>Method</th>
<th>ME</th>
<th>MHD</th>
<th>T (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liao’s</td>
<td>0.063902</td>
<td>4.013095</td>
<td>2.84</td>
</tr>
<tr>
<td>Luessi’s</td>
<td>0.063902</td>
<td>4.013095</td>
<td>1.29</td>
</tr>
<tr>
<td>K-means</td>
<td>0.097367</td>
<td>6.753009</td>
<td>1.27</td>
</tr>
<tr>
<td>Proposed</td>
<td>0.058984</td>
<td>3.441292</td>
<td>3.20</td>
</tr>
</tbody>
</table>

### Table 2. ME, MHD, and T for the second test group

<table>
<thead>
<tr>
<th>Method</th>
<th>ME</th>
<th>MHD</th>
<th>T (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liao’s</td>
<td>0.073685</td>
<td>6.936147</td>
<td>8778.28</td>
</tr>
<tr>
<td>Luessi’s</td>
<td>0.073685</td>
<td>6.936147</td>
<td>0.5</td>
</tr>
<tr>
<td>K-means</td>
<td>0.105336</td>
<td>11.048503</td>
<td>0.5</td>
</tr>
<tr>
<td>Proposed</td>
<td>0.068938</td>
<td>6.101892</td>
<td>3.88</td>
</tr>
</tbody>
</table>

### Table 3. \(\overline{\text{ME}}\) and \(\overline{T}\) for the third test group

<table>
<thead>
<tr>
<th>Method</th>
<th>(\overline{\text{ME}})</th>
<th>(\overline{T}) (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liao’s</td>
<td>0.169155</td>
<td>3.86</td>
</tr>
<tr>
<td>Luessi’s</td>
<td>0.169155</td>
<td>1.11</td>
</tr>
<tr>
<td>K-means</td>
<td>0.196328</td>
<td>0.54</td>
</tr>
<tr>
<td>Proposed</td>
<td>0.16149</td>
<td>3.93</td>
</tr>
</tbody>
</table>
filtered and median-filtered images. It can be seen that our method successfully integrates the noise resistance property from the multi-dimensional method and also the complex image handling capability from the multi-level method.

6. Acknowledgment

Funded by Kasetsart University Research and Development Institute (56.52)

References


