

Image Thresholding Using Graph Cuts

Wenbing Tao, Hai Jin, *Senior Member, IEEE*, Yimin Zhang, *Senior Member, IEEE*, Liman Liu, and Desheng Wang

Abstract—A novel thresholding algorithm is presented in this paper to improve image segmentation performance at a low computational cost. The proposed algorithm uses a normalized graph-cut measure as thresholding principle to distinguish an object from the background. The weight matrices used in evaluating the graph cuts are based on the gray levels of the image, rather than the commonly used image pixels. For most images, the number of gray levels is much smaller than the number of pixels. Therefore, the proposed algorithm requires much smaller storage space and lower computational complexity than other image segmentation algorithms based on graph cuts. This fact makes the proposed algorithm attractive in various real-time vision applications such as automatic target recognition. Several examples are presented, assessing the superior performance of the proposed thresholding algorithm compared with the existing ones. Numerical results also show that the normalized-cut measure is a better thresholding principle compared with other graph-cut measures, such as average-cut and average-association ones.

Index Terms—Graph cut, image thresholding, target recognition.

I. INTRODUCTION

IN MANY image processing applications, the gray levels of pixels belonging to an object are substantially different from those belonging to the background. As such, thresholding techniques can be used to extract the objects from their background. Indeed, thresholding is a major operation in many image processing applications such as document processing, image compression, particle counting, cell motion estimation, and object recognition. Thresholding techniques offer an efficient way, in terms of both the processing time and the implementation simplicity, to perform image segmentations.

The basic idea behind a thresholding process is to compare each pixel in an image with a certain threshold value. The de-

termination of an appropriate threshold value, which separates or segments a gray-level image into objects and background regions, is the core of a thresholding problem. An optimum threshold value is a gray level that divides an image into two segments, i.e., an object segment and a background segment, without compromising the object integrity. However, automatic determination of the optimum threshold value is often a difficult task [1]. While a number of approaches for automatic threshold determination have been proposed over the past several decades, applying new ideas and concepts to image thresholding remains an interesting and challenging research area.

Excellent reviews on early thresholding methods can be found in [2] and [3], whereas the latest development in this topic was summarized in [4]. Comparative performance studies of global thresholding techniques were reported by Lee *et al.* [5] and Glasbey [6]. Otsu [7] proposed a method that maximizes the between-class variance. Cheng *et al.* [8] presented a thresholding approach that performs fuzzy partition on a 2-D histogram based on fuzzy relation and maximum fuzzy entropy principle. Tao *et al.* [9] proposed a thresholding method for object segmentation based on fuzzy entropy theory and ant colony optimization algorithm. An image histogram thresholding approach using fuzzy sets was proposed by Tobias and Seara [10]. Belkasim *et al.* [11] presented a phase-based optimal image thresholding approach. Saha and Udupa [12] introduced a thresholding method that accounts for both intensity-based class uncertainty, a histogram-based property, and region homogeneity, an image-morphology-based property.

This paper concerns a graph-cut problem. Recently, there has been significant interest in image segmentation approaches based on graph cuts [13]–[26]. The common theme underlying these approaches is the formation of a weighted graph, where each vertex corresponds to an image pixel or a region. The weight of each edge connecting two pixels or two regions represents the likelihood that they belong to the same segment. A graph is partitioned into components in a way that some cost function of the vertices in the components and/or the boundary between those components is minimized. A graph can also be partitioned into more than two components by recursively bipartitioning the graph until some termination criterion is met. The termination criterion is often based on the same cost function that is used for bipartitioning.

There is a rich body of prior works on graph cuts. However, minimizing a cost function through graph cuts remains a technically challenging problem. Different papers construct different graphs with unique energy functions, and in some cases (e.g., [16] and [27]), the construction is very complicated. The graphs are usually represented using the adjacency list or the matrix representation. These algorithms divide the initial graph into subgraphs that correspond to different image regions.

Manuscript received September 12, 2006; revised May 21, 2007 and November 21, 2007. The work of W. Tao was supported by the National Natural Science Foundation of China under Grant 60603024, and the work of D. Wang was supported in part by the NTU Start-up Grant M58110011. This paper was recommended by Associate Editor J. Wu.

W. Tao and H. Jin are with the Cluster and Grid Computing Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China, and also with the Service Computing Technology and System Laboratory of Ministry of Education, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: wenbingtao@hust.edu.cn; hjin@hust.edu.cn).

Y. Zhang is with the Center for Advanced Communications, Villanova University, Villanova, PA 19085 USA (e-mail: yimin@ieee.org).

L. Liu was with the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China. She is now with O₂Micro Corporation, Wuhan 430074, China (e-mail: mandyliu_80@yahoo.com.cn).

D. Wang is with the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637616 (e-mail: Desheng@ntu.edu.sg).

Digital Object Identifier 10.1109/TSMCA.2008.2001068

There are several methods in this category, which are based on the notion of graph cuts derived from the spectrum of the graph [23]. The spectrum is composed of the eigenvalues and eigenvectors of the matrix representation [13], [14], [28], [29]. Nevertheless, the problem of perceptual grouping is nondeterministic polynomial-time hard, and practically, only approximate solutions can be found [14].

Image segmentation approaches based on graph cuts, in general, require high computational complexity and provide poor real-time performance. Therefore, they may not be practical in many image segmentation problems. In this paper, we use a graph-cut measure as the thresholding principle to distinguish objects from the background. Similar to the existing techniques, the proposed method constructs a weighted graph by treating each pixel as a node and connecting each pair of pixels by an edge. The weight on the edge should reflect the likelihood that the two pixels belong to the same segment. However, unlike the general image partitioning approaches which are developed in [14] and [17] by either solving an eigensystem or using complicated graph reduction procedures, the purpose of this paper is to develop a simple and effective thresholding approach with significantly reduced computational cost. Such cost reduction is achieved by representing a graph using a 256×256 symmetrical weight matrix based on gray levels, rather than the $N \times N$ symmetrical weight matrix based on pixels, where N is the number of pixels in the image. Because N is typically much larger than 256, the size of the weight matrix based on the gray levels is much smaller than $N \times N$. As a result, it becomes feasible to quickly obtain graph cut values for every possible threshold t from this weight matrix.

Based on the type of information used, Sezgin and Sankur [4] classified thresholding algorithms into the following six categories: histogram-shape-based thresholding methods, clustering-based thresholding methods, entropy-based thresholding methods, attribute-similarity-based thresholding methods, spatial thresholding methods, and locally adaptive thresholding methods. To make complete yet effective comparisons, we select one method with relatively good performance from each of the six categories and compare their performance with the proposed thresholding method. The six selected thresholding algorithms are as follows: Ramesh method [30], Kittler method [31], Kapur method [32], Pikaz method [33], Pal method [34], and robust automatic threshold selection (RATS) method [35], [36], [39], where parameter $\lambda = 5$, used in [39].

This paper is organized as follows. In Section II, the new image thresholding method is proposed for effective graph partitioning. In Section III, we evaluate the performance of the proposed thresholding approach using a variety of real images and compare it with other relevant techniques from the literature. In Section IV, the performance dependence of the proposed approach on method's parameters is examined, and the computational cost is compared to other graph-cut-based image segmentation methods. In Section V, the performance of several other graph-cut measures is compared with the normalized-cut measure. Finally, Section VI concludes this paper.

II. THRESHOLDING METHOD BASED ON GRAPH CUTS

A. Background

The set of points in an arbitrary feature space is represented as a weighted undirected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where \mathbf{V} is the set of vertices, \mathbf{E} is the set of edges, and the cardinality of \mathbf{V} is $N = |\mathbf{V}|$. An edge is formed between each pair of nodes, and the weight on the edge, $w(u, v)$, is a function describing the similarity between two nodes u and v . A graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ is partitioned into two disjoint complementary sets \mathbf{A} and $\mathbf{B} = \mathbf{V} - \mathbf{A}$. The degree of dissimilarity between the two sets can be computed as a total weight of the edges connecting the two parts. That closely relates to a mathematical formulation of a cut [13]

$$\text{cut}(\mathbf{A}, \mathbf{B}) = \sum_{u \in \mathbf{A}, v \in \mathbf{B}} w(u, v). \quad (1)$$

The objective is to find the optimal bipartitioning of a graph \mathbf{G} . This optimization problem is well studied, and some algorithms to find the minimum cut are developed in [37] and [38]. Wu and Leahy [13] proposed a clustering method based on the minimum-cut criterion which yields good segmentation results but favors cutting to small sets of isolated nodes in a graph. To prevent such unnatural bias of cutting into small sets of nodes, Shi and Malik [14] proposed a new measure of disassociation between two groups, termed normalized cuts (Ncut), which is expressed as

$$\text{Ncut}(\mathbf{A}, \mathbf{B}) = \frac{\text{cut}(\mathbf{A}, \mathbf{B})}{\text{asso}(\mathbf{A}, \mathbf{V})} + \frac{\text{cut}(\mathbf{A}, \mathbf{B})}{\text{asso}(\mathbf{B}, \mathbf{V})} \quad (2)$$

where $\text{asso}(\mathbf{A}, \mathbf{V}) = \sum_{u \in \mathbf{A}, t \in \mathbf{V}} w(u, t)$ is the total connection from nodes in \mathbf{A} to all nodes in the graph, and $\text{asso}(\mathbf{B}, \mathbf{V})$ is similarly defined. The bipartitioning process that minimizes Ncut is the optimal bipartitioning of \mathbf{G} . In this case, (2) can be transformed into the following standard eigensystem [14]:

$$\mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}}\mathbf{z} = \lambda\mathbf{z} \quad (3)$$

where \mathbf{D} is an $N \times N$ diagonal matrix with diagonal elements $d_i = \sum_j w(i, j)$, \mathbf{W} is a symmetrical matrix with elements $w(i, j)$, and λ and \mathbf{z} are an eigenvalue and the corresponding eigenvector, respectively.

The eigenvector corresponding to the second smallest eigenvalue is the real-valued solution that optimally subpartitions the entire graph [14]. The approximate Lanczos method is often used to compute such an eigenvalue [14]. A heuristic method was adopted for splitting-point search because eigenvector elements take continuous values. The algorithm is recursively applied to every subgraph until the value of Ncut exceeds a certain threshold. The normalized-cut algorithm is computationally expensive when the dimension of the weight matrix is large because the pixel-based weight matrix requires to compute the $N \times N$ weight matrix and solve the eigensystem (3). Even though the approximate eigenvalue method and the algorithm construction optimize the implementations, the computational complexity remains prohibitively high for an image of a moderate or large size. In addition, the performance and stability

of the partitioning procedure depend highly on the selection of the parameters, whereas the optimum parameters are more or less data dependent. The smallest nonzero eigenvalues, for a given eigensystem with sparse matrices, often have a very small magnitude, and therefore, the selection of the splitting point by a heuristic method is influenced by the arithmetic precision. The stability is determined by many factors such as the selection of the splitting point, the precision of eigenvalue computations, and the relative segment position. All these facts may limit the application of normalized cuts in practice.

B. Proposed Approach

In this section, we propose an improved approach which is described by the following steps.

- 1) Let $\mathbf{V} = \{(i, j) : i = 0, 1, \dots, n_h - 1; j = 0, 1, \dots, n_w - 1\}$, $L = \{0, 1, \dots, 255\}$, where n_h and n_w are the height and width of the image, respectively. Let $f(x, y)$ be the gray-level value of the image at pixel (x, y) . \mathbf{V} and $f(x, y)$ satisfy

$$f(x, y) \in L \quad \forall (x, y) \in \mathbf{V} \quad (4)$$

$$\mathbf{V}_k = \{(x, y) : f(x, y) = k, (x, y) \in \mathbf{V}\}, \quad k \in L \quad (5)$$

$$\bigcup_{k=0}^{255} \mathbf{V}_k = \mathbf{V} \quad \mathbf{V}_j \cap \mathbf{V}_k = \Phi, \quad k \neq j, \quad k, j \in L. \quad (6)$$

- 2) Construct a weighted graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ by taking each pixel as a node and connecting each pair of pixels by an edge. The weight on an edge should reflect the likelihood that the two pixels belong to the same object.
- 3) Using only the brightness of the pixels and their spatial locations, we can define the weight of the graph edge connecting two nodes u and v as (7), shown at the bottom of the page, where $\mathbf{F}(u)$ and $\mathbf{X}(u)$ are the gray scale and spatial location of node u , respectively, and $\|\cdot\|_2$ denotes the vector norm. In addition, d_I and d_X are positive scaling factors that determine the sensitivity of $w(u, v)$ to the intensity difference and spatial location between two nodes, respectively, and r is a positive integer that specifies the number of neighboring nodes involved in the weight computations. As r increases, more nodes are involved in the computation of the weight, and thus, the computational cost becomes higher.
- 4) For any $t (0 \leq t < 255)$, we can obtain a unique bisection $\mathbf{V} = \{\mathbf{A}, \mathbf{B}\}$ of the corresponding graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where sets \mathbf{A} and \mathbf{B} can be formulated as

$$\mathbf{A} = \bigcup_{k=0}^t \mathbf{V}_k \quad \mathbf{B} = \bigcup_{k=t+1}^{255} \mathbf{V}_k, \quad k \in L. \quad (8)$$

Then, (1) becomes

$$\begin{aligned} \text{cut}(\mathbf{A}, \mathbf{B}) &= \sum_{u \in \mathbf{A}, v \in \mathbf{B}} w(u, v) = \sum_{u \in \mathbf{A}} \left[\sum_{v \in \mathbf{B}} w(u, v) \right] \\ &= \sum_{i=0}^t \sum_{u \in \mathbf{V}_i} \left[\sum_{j=t+1}^{255} \sum_{v \in \mathbf{V}_j} w(u, v) \right] \\ &= \sum_{i=0}^t \sum_{j=t+1}^{255} \left[\sum_{u \in \mathbf{V}_i, v \in \mathbf{V}_j} w(u, v) \right] \\ &= \sum_{i=0}^t \sum_{j=t+1}^{255} \text{cut}(\mathbf{V}_i, \mathbf{V}_j) \end{aligned} \quad (9)$$

where

$$\text{cut}(\mathbf{V}_i, \mathbf{V}_j) = \sum_{u \in \mathbf{V}_i, v \in \mathbf{V}_j} w(u, v) \quad (10)$$

is the total connection between all nodes in \mathbf{V}_i (whose gray level is i) and all nodes in \mathbf{V}_j (whose gray level is j). Similarly

$$\begin{aligned} \text{asso}(\mathbf{A}, \mathbf{A}) &= \sum_{u \in \mathbf{A}, v \in \mathbf{A}} w(u, v) \\ &= \sum_{i=0}^t \sum_{j=i}^t \left[\sum_{u \in \mathbf{V}_i, v \in \mathbf{V}_j} w(u, v) \right] \\ &= \sum_{i=0}^t \sum_{j=i}^t \text{cut}(\mathbf{V}_i, \mathbf{V}_j) \end{aligned} \quad (11)$$

$$\begin{aligned} \text{asso}(\mathbf{B}, \mathbf{B}) &= \sum_{u \in \mathbf{B}, v \in \mathbf{B}} w(u, v) \\ &= \sum_{i=t+1}^{255} \sum_{j=i}^{255} \left[\sum_{u \in \mathbf{V}_i, v \in \mathbf{V}_j} w(u, v) \right] \\ &= \sum_{i=t+1}^{255} \sum_{j=i}^{255} \text{cut}(\mathbf{V}_i, \mathbf{V}_j). \end{aligned} \quad (12)$$

Note that the following relations hold:

$$\text{asso}(\mathbf{A}, \mathbf{V}) = \text{asso}(\mathbf{A}, \mathbf{A}) + \text{cut}(\mathbf{A}, \mathbf{B}) \quad (13)$$

$$\text{asso}(\mathbf{B}, \mathbf{V}) = \text{asso}(\mathbf{B}, \mathbf{B}) + \text{cut}(\mathbf{A}, \mathbf{B}). \quad (14)$$

Therefore, (2) becomes

$$\begin{aligned} \text{Ncut}(\mathbf{A}, \mathbf{B}) &= \frac{\text{cut}(\mathbf{A}, \mathbf{B})}{\text{asso}(\mathbf{A}, \mathbf{A}) + \text{cut}(\mathbf{A}, \mathbf{B})} \\ &\quad + \frac{\text{cut}(\mathbf{A}, \mathbf{B})}{\text{asso}(\mathbf{B}, \mathbf{B}) + \text{cut}(\mathbf{A}, \mathbf{B})}. \end{aligned} \quad (15)$$

$$w(u, v) = \begin{cases} e^{-\left[\frac{\|\mathbf{F}(u) - \mathbf{F}(v)\|_2^2}{d_I} + \frac{\|\mathbf{X}(u) - \mathbf{X}(v)\|_2^2}{d_X} \right]}, & \text{if } \|\mathbf{X}(u) - \mathbf{X}(v)\|_2 < r \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

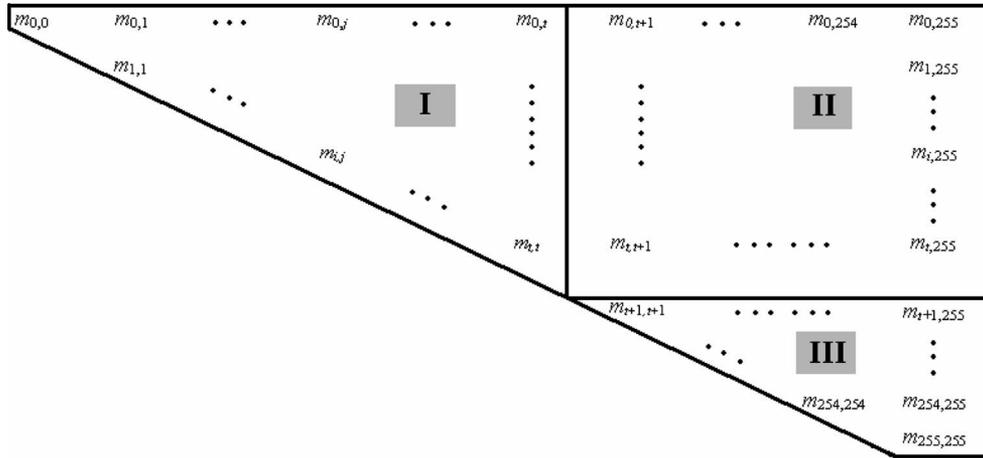


Fig. 1. Graph illustration of symmetrical matrix $\mathbf{M} = [m_{i,j}]_{256 \times 256}$ with $m_{i,j} = m_{j,i}$.

5) Let \mathbf{M} be the 256×256 symmetrical matrix with elements $m_{i,j} = m_{j,i} = \text{cut}(\mathbf{V}_i, \mathbf{V}_j)$. As shown in Fig. 1 (only the elements in the upper triangle are shown due to symmetry), matrix $\mathbf{M} = [m_{i,j}]_{256 \times 256}$ can be uniquely constructed for a given image by computing all the weights on the edges connecting each pair of pixels in the image.

Note that, by partitioning the image using the normalized-cut measure, matrix \mathbf{M} is of size 256×256 , and the information from the $N \times N$ matrix \mathbf{W} is no longer necessary. From matrix \mathbf{M} , we can compute $\text{cut}(\mathbf{A}, \mathbf{B})$, $\text{asso}(\mathbf{A}, \mathbf{A})$, and $\text{asso}(\mathbf{B}, \mathbf{B})$ of the corresponding bisection of the weight graph for each threshold t . As shown in Fig. 1, the elements of matrix \mathbf{M} are divided into three parts. First, the sum of all the elements in part I is the value of $\text{asso}(\mathbf{A}, \mathbf{A})$. Second, the sum of all the elements in part II is the value of $\text{cut}(\mathbf{A}, \mathbf{B})$. Finally, the sum of all the elements in part III gives the value of $\text{asso}(\mathbf{B}, \mathbf{B})$. Therefore, the normalized cuts can now be easily computed from \mathbf{M} for every possible threshold t . Moreover, unlike matrix \mathbf{W} whose dimension depends on the image size, \mathbf{M} is a symmetrical matrix of fixed size 256×256 , irrespective of the image size.

The proposed thresholding method searches the optimal threshold value minimizing the corresponding normalized cuts of the image. The proposed algorithm is shown in Fig. 2, where T is the optimum threshold, $0 \leq t \leq 255$ is a variable threshold, and $Ncut_{\min}$ is the minimum value of the normalized cut.

Note that, when $\text{asso}(\mathbf{A}, \mathbf{V}) = 0$ for some threshold t , the number of nodes in set \mathbf{A} is zero. This case is meaningless because the image has no pixels with brightness less than or equal to the threshold t and cannot be segmented into two parts by this threshold. Because threshold t with the minimum $Ncut$ is the optimal threshold, we can assume that, under the condition $\text{asso}(\mathbf{A}, \mathbf{V}) = 0$, $Ncut$ takes the maximum value to avoid such meaningless situation. Similar treatment can be performed for $\text{asso}(\mathbf{B}, \mathbf{V}) = 0$.

III. PERFORMANCE EVALUATION AND COMPARISON WITH EXISTING METHODS

A set of images is used to evaluate the performance of the proposed algorithm as well as some of the commonly used

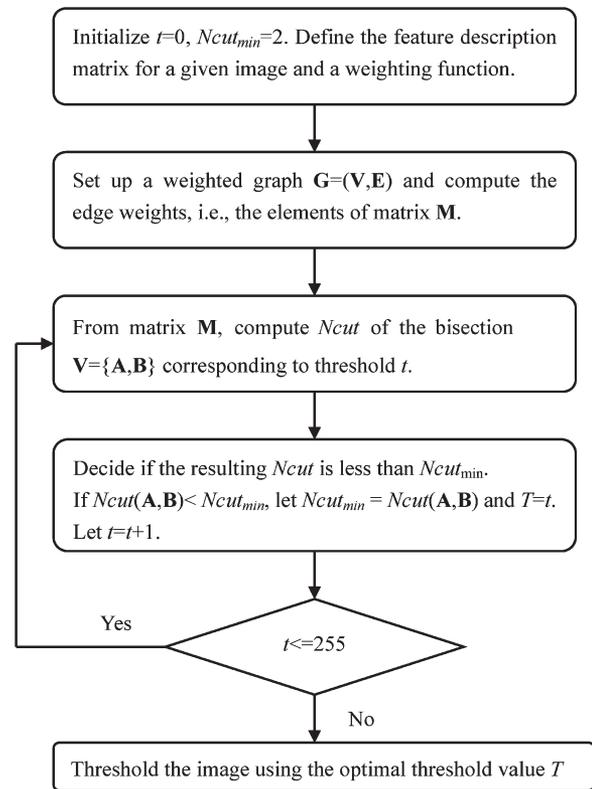


Fig. 2. Implementation flowchart of the proposed algorithm.

algorithms presented in the literature. Performance evaluations and comparisons are carried out by using real images, where the object can be exactly distinguished from the background using some suitable threshold. Particularly, some infrared object images are selected to examine our algorithm because infrared sensors have good night vision performance and are widely applied in automatic target recognition (ATR). In all the examples used in this section, the parameters used in (7) are set to $d_I = 625$, $d_X = 4$, and $r = 2$.

The main comparison criterion is the absolute error ratio. The absolute error is defined as the absolute difference in the number of object pixels between the optimally thresholded image and the thresholded image obtained by each method. The

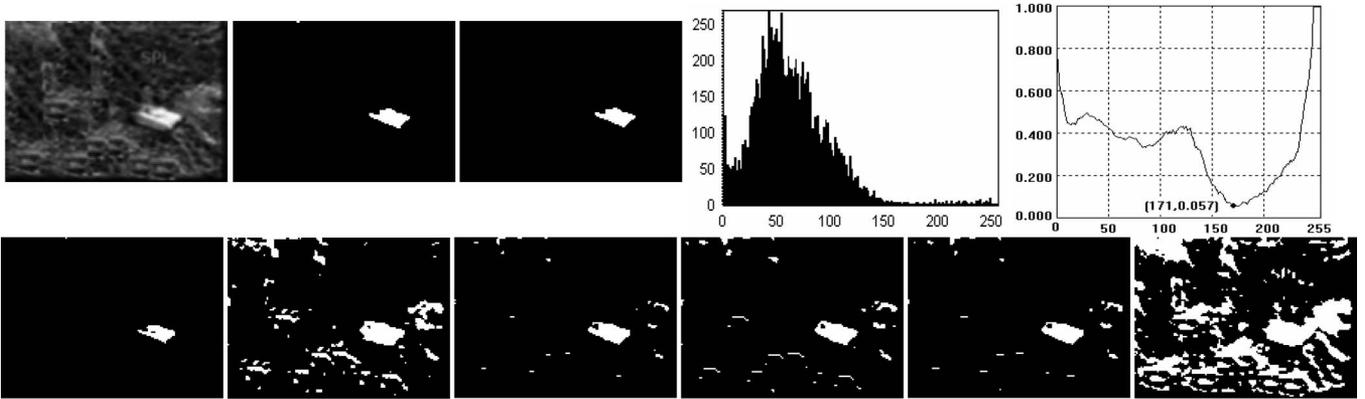


Fig. 3. (Top row, from left to right) Original tank image (148×106), thresholding result by the proposed method ($T = 171$), manually thresholding image ($T = 175$), histogram of the original image, and value of N_{cut} versus threshold t . (Bottom row, from left to right) Results by Pikaz method ($T = 193$), Kittler method ($T = 109$), Kapur method ($T = 140$), RATS method ($T = 128$), Ramesh method ($T = 133$), and Pal method ($T = 78$).

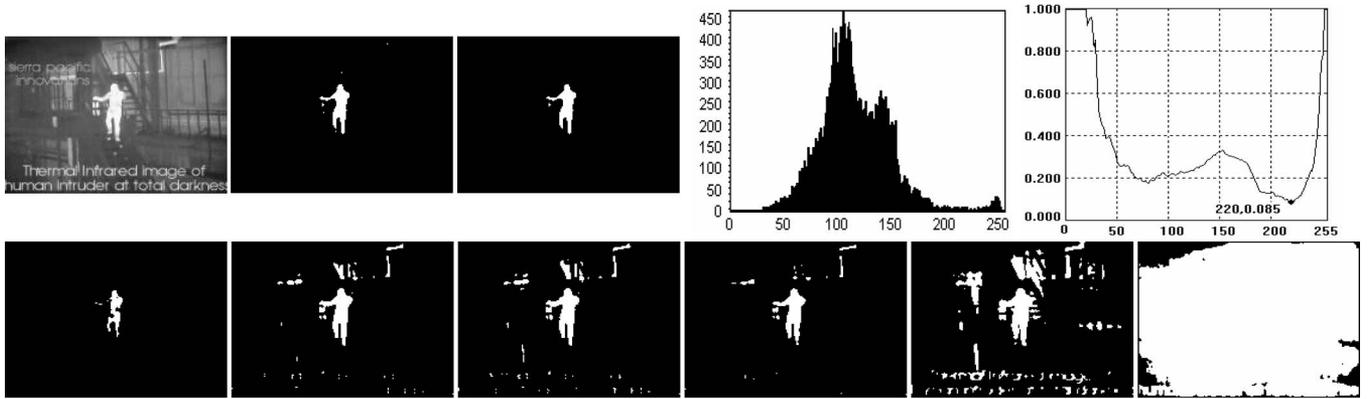


Fig. 4. (Top row, from left to right) Original intruder image (185×141), thresholding result by the proposed method ($T = 220$), manually thresholding image ($T = 221$), histogram of the original image, and value of N_{cut} versus threshold t . (Bottom row, from left to right) Results by Pikaz method ($T = 241$), Kittler method ($T = 180$), Kapur method ($T = 176$), RATS method ($T = 188$), Ramesh method ($T = 154$), and Pal method ($T = 74$).

optimally thresholded image and the corresponding threshold value are manually obtained using visual inspection. The absolute error ratio is determined as the ratio between the absolute error n_{diff} and the total number of pixels N of an image, i.e.,

$$r_{err} = \frac{n_{diff}}{N} \times 100\%. \quad (16)$$

The results of the first set of examples are shown in Figs. 3–6. Each figure shows the original gray-level image, the histogram of the image, the optimally thresholded image, the thresholded images using the proposed as well as other methods used in the comparison, and the values of N_{cut} as a function of threshold t . The images used for comparison include a ground infrared image with one tank object (Fig. 3), a night infrared image with an intruder (Fig. 4), an infrared image with one ship object (Fig. 5), and an air infrared image with two small objects (Fig. 6). For each image, the respective optimum threshold value T is chosen as the one corresponding to the minimum value of N_{cut} . The deviations from the optimum threshold value are shown in Table I for different algorithms. It is evident from Figs. 3–6 and Table I that the proposed algorithm provides better segmentation performance than the other techniques. The values of the absolute error for different algorithms are compared in Table II.

Note that, in all the four examples, the proposed algorithm can effectively extract the infrared object from the background and that the performance is close to the optimum one obtained by manual thresholding. Compared with the other methods, the proposed method has improved adaptability and robustness because it can distinguish not only big objects (e.g., Figs. 3–5) but also small objects of only several pixels (e.g., Fig. 6) from the background.

Fig. 7 shows the test results of some trimmed images of the ship image used in Fig. 5. The purpose of this example is to show that the proposed method is insensitive to the change of the background, given the high homogeneity inside the object. As shown in Fig. 7, we select three different trimmed images that all include the ship object but have different backgrounds from the original ship image. The results obtained using the proposed method show that the distinguished objects are almost identical despite the different backgrounds.

Fig. 8 shows another example. The test images are three coin images with a gray background. The first image includes a black coin and a bright coin, whereas the second image includes only one black coin, and the last one includes only one bright coin. From the result corresponding to each of the last two single-coin images, the proposed method effectively extracted the respective coin from the gray background. Observing the

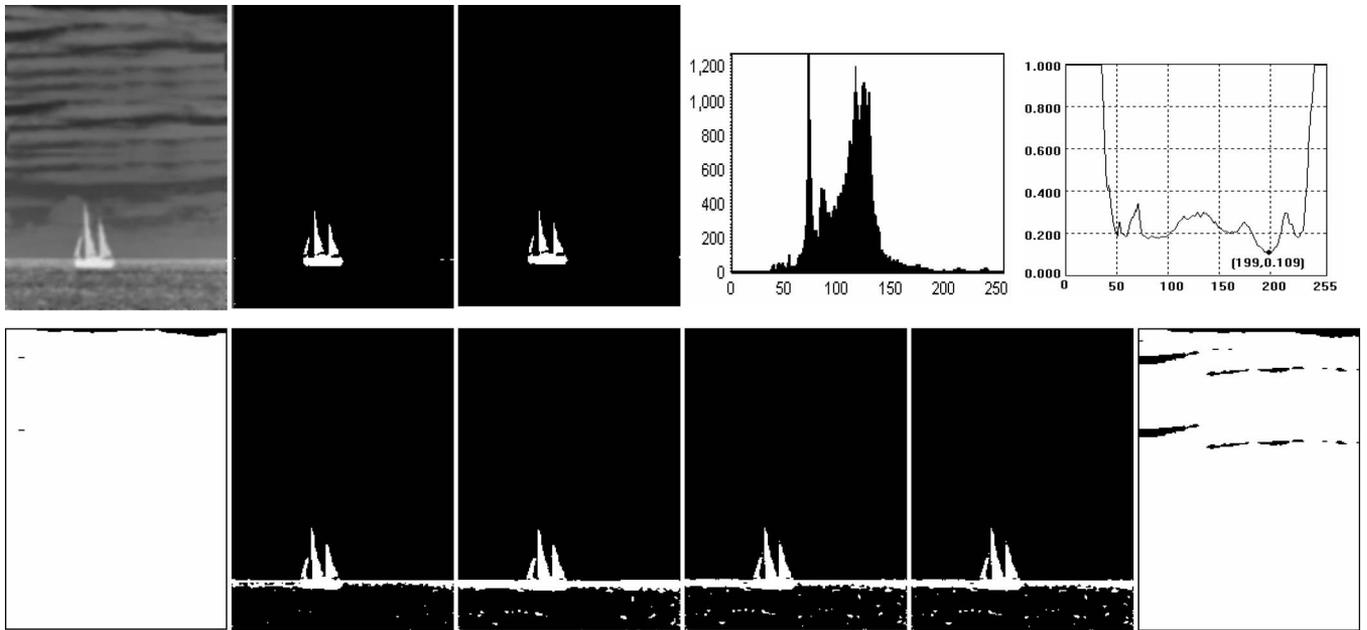


Fig. 5. (Top row, from left to right) Original ship image (182×253), thresholding result by the proposed method ($T = 199$), manually thresholding image ($T = 201$), histogram of the original image, and value of N_{cut} versus threshold t . (Bottom row, from left to right) Result by Pikaz method ($T = 51$), Kittler method ($T = 165$), Kapur method ($T = 148$), RATS method ($T = 156$), Ramesh method ($T = 155$), and Pal method ($T = 66$).

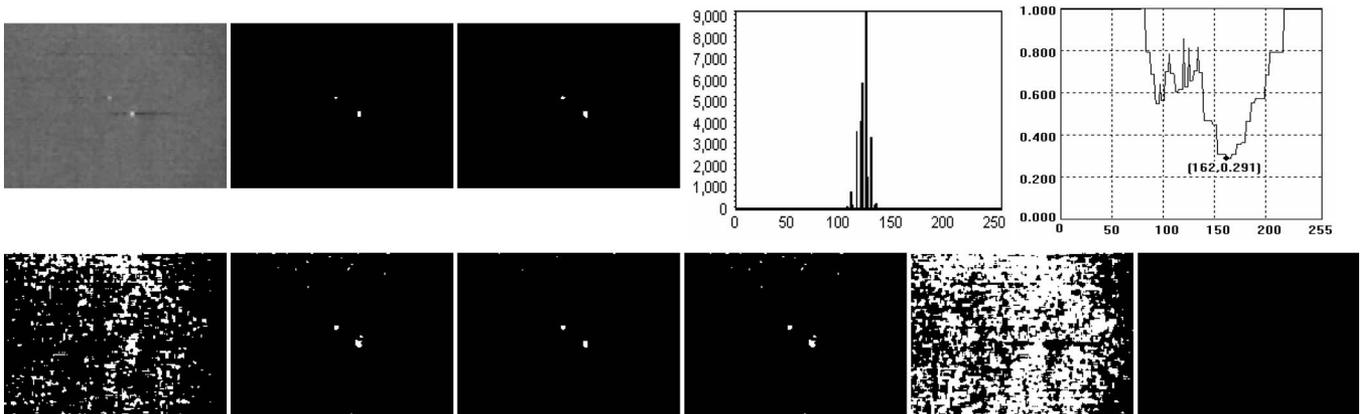


Fig. 6. (Top row, from left to right) Original air object image (200×150), thresholding result by the proposed method ($T = 162$), manually thresholding image ($T = 146$), histogram of the original image, and value of N_{cut} versus threshold t . (Bottom row, from left to right) Result by Pikaz method ($T = 127$), Kittler method ($T = 136$), Kapur method ($T = 139$), RATS method ($T = 138$), Ramesh method ($T = 121$), and Pal method ($T = 249$).

TABLE I
COMPARISON OF THRESHOLD VALUES AND ERROR RATIO (PERCENT) FOR THE IMAGES SHOWN IN FIGS. 3–6

Figure	Ref. Value	Thresholding algorithms													
		Ramesh		Kittler		Kapur		RATS		Pikaz		Pal		Proposed	
		T	Error	T	Error	T	Error	T	Error	T	Error	T	Error	T	Error
Tank	175	133	1.37	109	6.75	140	0.82	128	2.02	193	0.20	78	25.10	171	0.013
Intruder	221	154	7.81	180	1.84	176	2.34	188	1.10	241	0.50	74	90.12	220	0.019
Ship	201	155	0.28	169	1.52	148	3.85	156	2.71	51	97.74	66	95.53	199	0.039
Air obj	146	121	49.48	136	0.18	139	0.04	138	0.18	127	13.42	249	0.097	162	0.037

N_{cut} plots of the three images, we find that the plot of the two-coin image is very close to the superposition of the plots of the two single-coin images, and the two valley points of the plot of the two-coin image exactly match those of the plots of the two single-coin images. This example shows that the proposed

method can simultaneously distinguish multiple objects with different brightness in an image.

The second set of the test images includes the cameraman and Lena images. It differs from the first set of test images in the sense that the two images in the second set do not have

TABLE II
COMPARISON OF ERROR PIXELS FOR THE IMAGES SHOWN IN FIGS. 3–6

Figure	Size of images	Ref. object pixel number	Ramesh	Kittler	Kapur	RATS	Pikaz	Pal	Proposed
Tank	148x106	181	215	1059	128	317	31	3938	2
Intruder	185x141	380	2038	481	610	286	130	23508	5
Ship	182x253	497	1300	700	1773	1249	45006	43987	18
Air obj	200x150	29	14845	55	13	55	4027	29	11

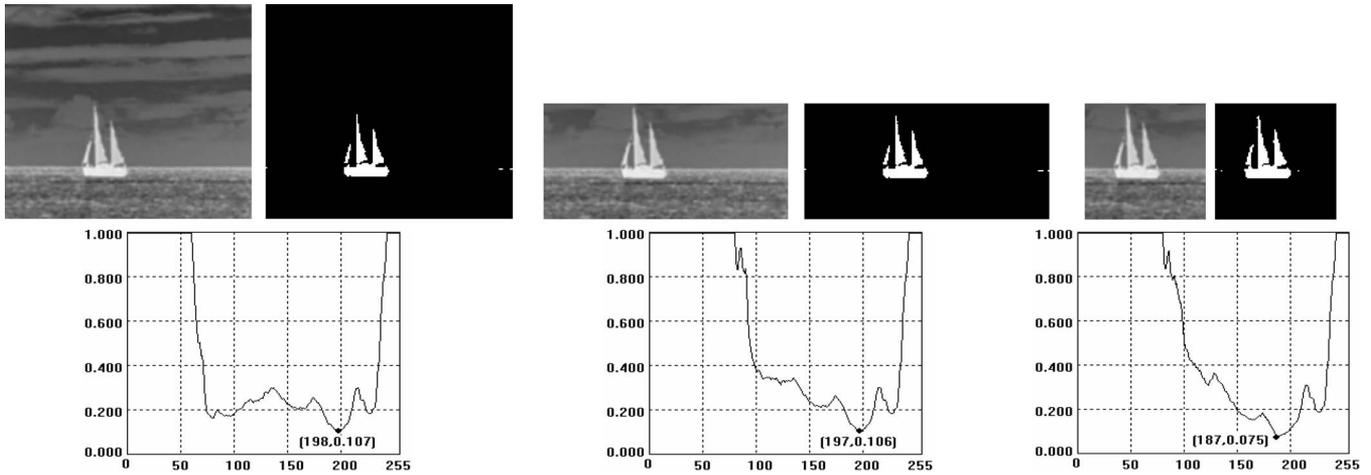


Fig. 7. (Top row, from left to right) The part ship image with size 182×158 and its result by the proposed method. The part ship image with size 182×85 and its result by the proposed method. The part ship image with size 88×85 and its result by the proposed method. (Bottom row, from left to right) Value of N_{cut} for the part ship images with sizes 182×158 , 182×85 , and 88×85 , respectively.

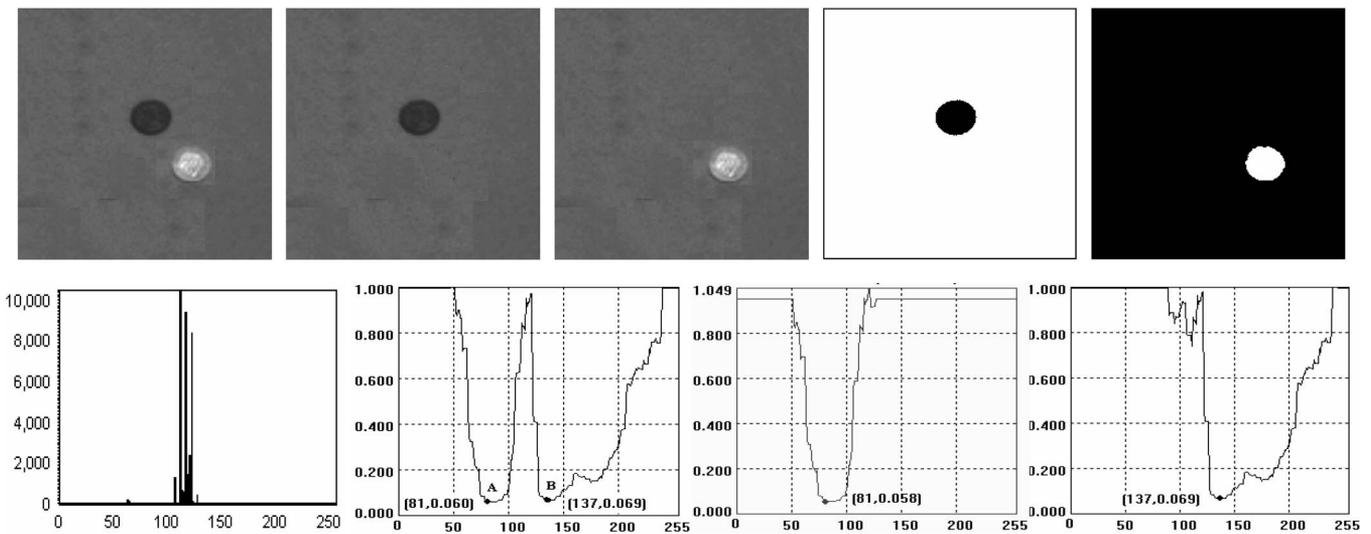


Fig. 8. (Top row, from left to right) The image with a black coin and a bright coin (216×236), the image with only one black coin, the image with only one bright coin, thresholding result of the black coin image by the proposed method ($T = 81$), and thresholding result of the bright coin image by the proposed method ($T = 137$). (Bottom row, from left to right) Histogram of the image with two coins, value of N_{cut} for the image with two coins, value of N_{cut} for the black coin image, and value of N_{cut} for the bright coin image.

distinct objects and backgrounds, and their histograms in Fig. 9 are “clean” bimodal or multimodal. Therefore, we cannot use the absolute error ratio to compare the performance between the proposed method and the other algorithms. The thresholding results of different methods, however, can be evaluated by using visual inspection. Fortunately, the two images are standard

image data which are often used to examine the performance of segmentation algorithms because of their good multimodal characteristics and rich detailed information. From this point of view, the two images are helpful to validate the superiority of the proposed method compared to the other algorithms. Fig. 9 shows the results of the two images processed by the proposed

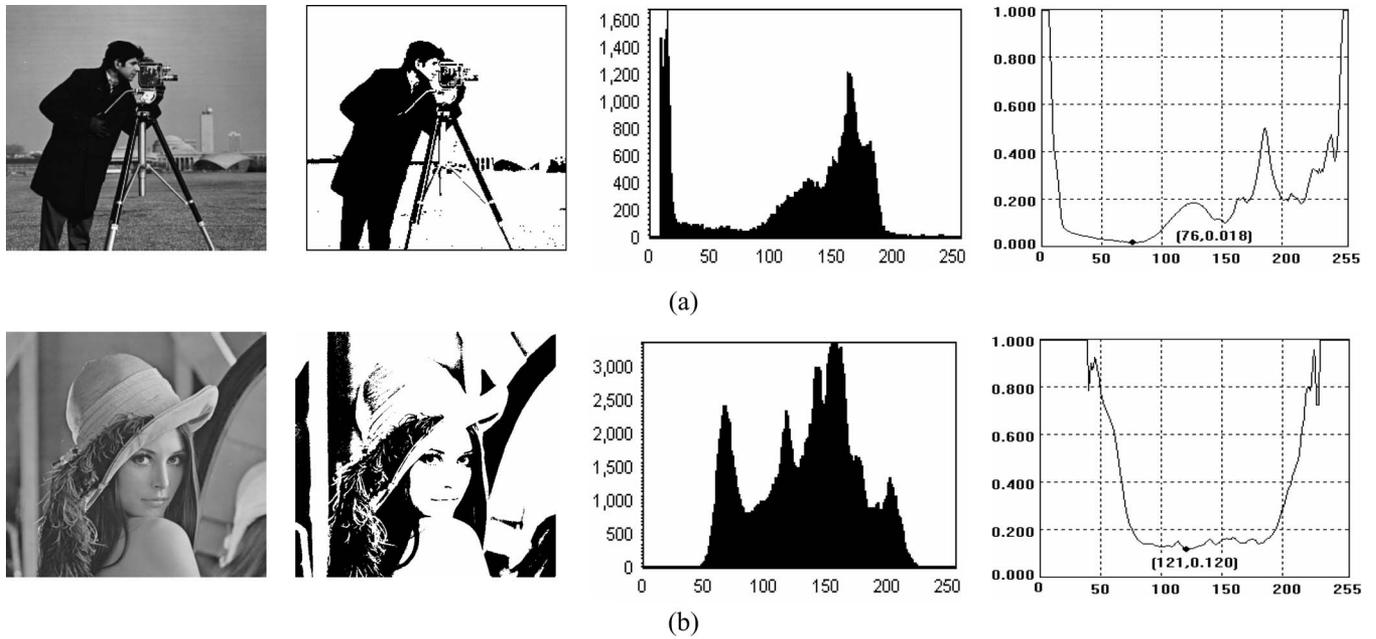


Fig. 9. Four columns in each set represent the original image, the result by the proposed method, the image histogram, and the plot of N_{cut} versus threshold t , respectively. The size of the images is 256×256 . (a) Cameraman. (b) Lena.

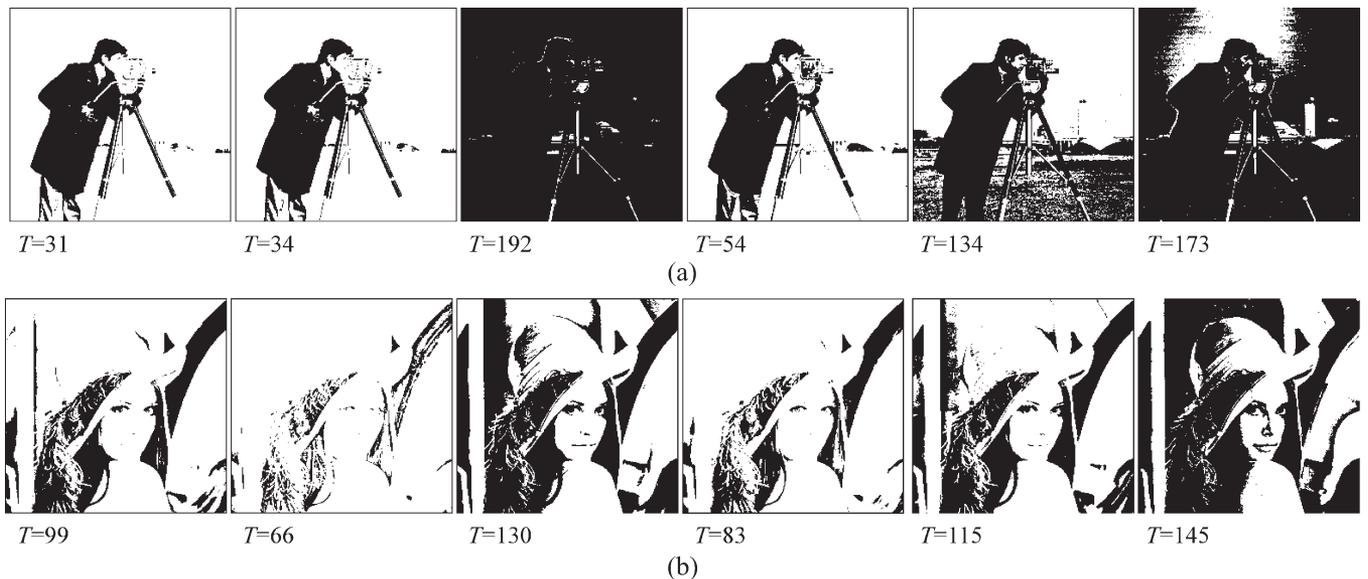


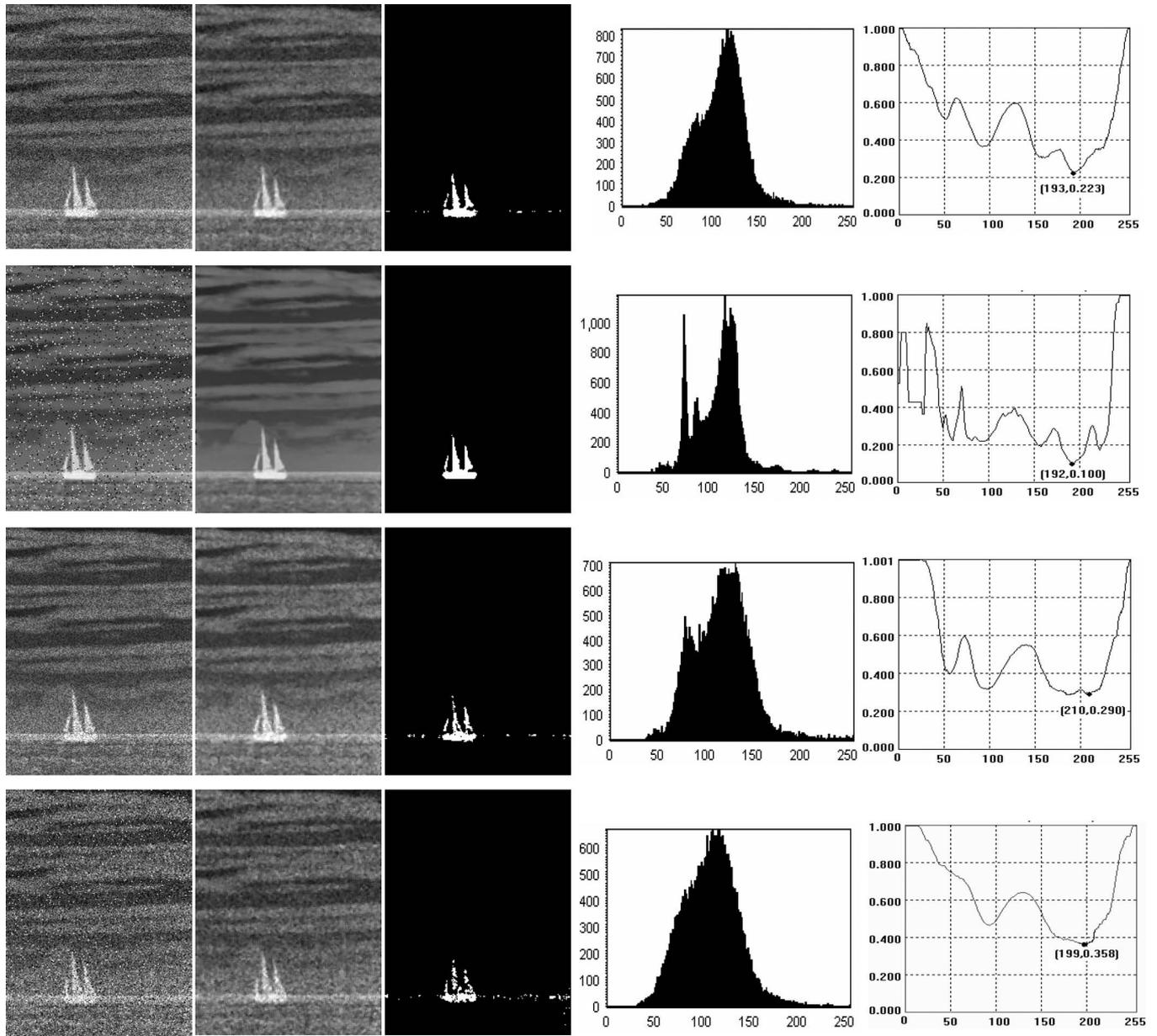
Fig. 10. Six columns in each set represent the results by Pikaz method, Kittler method, Kapur method, RATS method, Ramesh method, and Pal method, respectively. The respective optimum threshold T is indicated in each image. The size of the images is 256×256 . (a) Cameraman. (b) Lena.

method, whereas Fig. 10 shows the results processed by the other six algorithms. Based on visual evaluations, we conclude that the proposed method provides the best thresholding performance among all the methods being compared.

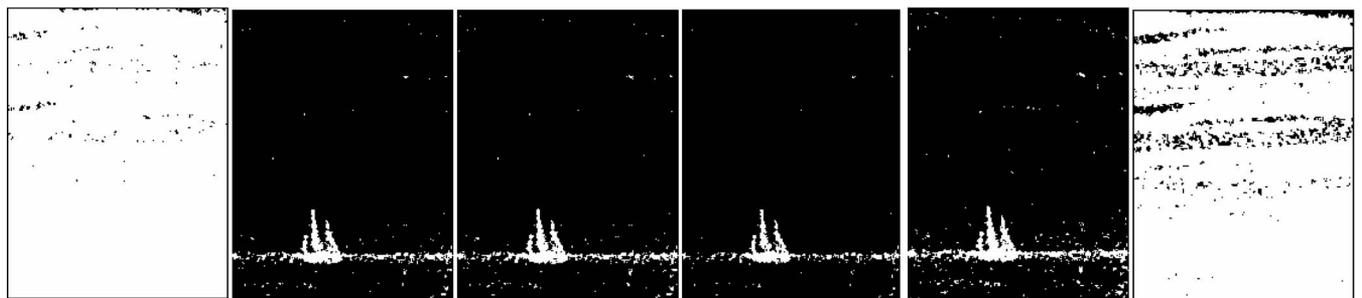
The performance of the proposed method in the presence of noise is also studied. Three types of noise, i.e., Gaussian noise, salt and pepper noise, and speckle noise, are considered. These types of noise are usually encountered in images. We choose the ship image, which has been used earlier in this paper, to evaluate the performance of various algorithms in the presence of degeneration. In this case, local noise information would affect the value of N_{cut} , and therefore, a larger value of r is

used to reduce the effect. The parameters used in (7) are set to $d_I = 625$, $d_X = 8$, and $r = 4$.

Fig. 11 shows the results of the ship image corrupted by the three types of noise. We preprocess the image using a 3×3 median filter before thresholding. The first four rows of Fig. 11 show the results of the ship image in the presence of the three types of noise as well as the mixture of the three types of noise, respectively, using the proposed method. It is evident that the proposed algorithm performs very well in the presence of these types of noise. From the fourth and fifth columns of the first four rows of Fig. 11, it is seen that, while the image histograms are changed due to the presence of noise, the effect



(a)



(b)

Fig. 11. Test results of the noisy ship image. (a) Each row shows the results of the ship image corrupted by a certain type of noise. (From the top) Additive Gaussian noise of zero mean and variance of 0.01, salt and pepper noise with intensity $d = 0.05$, speckle noise with variance $v = 0.04$, and the mixture of the four types of noise. (At each row, from left to right) Noisy image, result image after noise reduction using 3×3 median filter, thresholding result of the filtered image using the proposed method, the filtered image histogram, and the Ncut plot of the filtered image as a function of threshold t . (b) Thresholding images of the filtered image using different methods in the presence of mixture noise. (From left to right) Pikaz method ($T = 47$), Kittler method ($T = 170$), Kapur method ($T = 168$), RATS method ($T = 177$), Ramesh method ($T = 161$), and Pal method ($T = 66$).

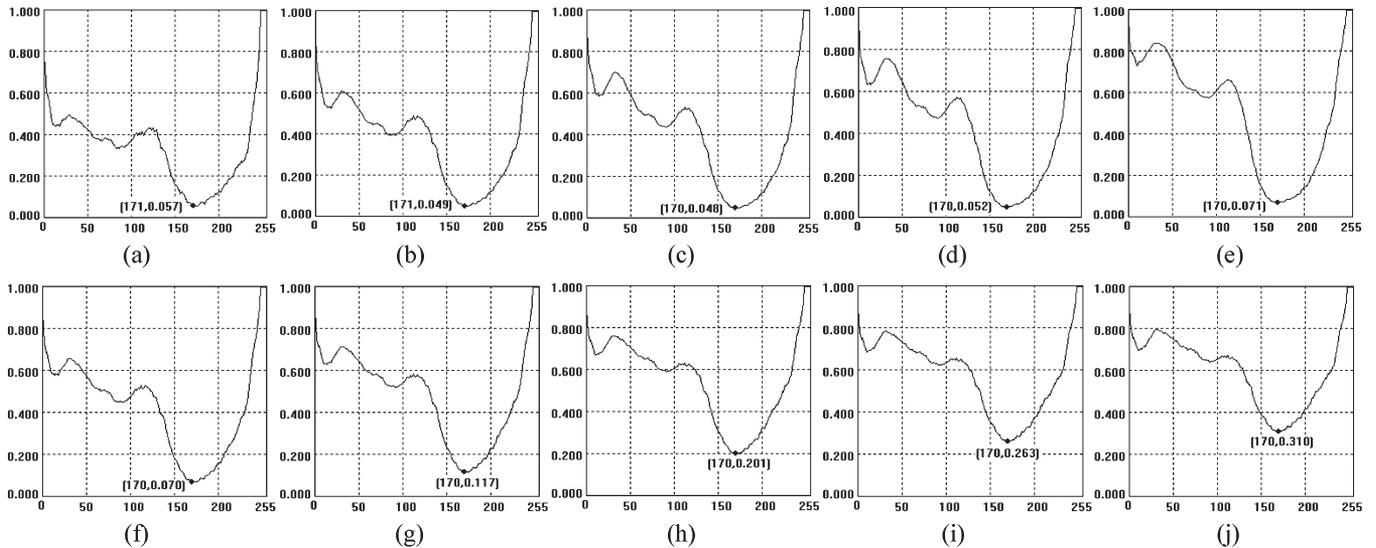


Fig. 12. Plots of $Ncut$ of the tank image as a function of threshold t using different parameter settings. (a) $r = 2$, $d_I = 625$, and $d_X = 4$. (b) $r = 4$, $d_I = 625$, and $d_X = 8$. (c) $r = 8$, $d_I = 625$, and $d_X = 16$. (d) $r = 16$, $d_I = 625$, and $d_X = 32$. (e) $r = 32$, $d_I = 625$, and $d_X = 250$. (f) $r = 4$, $d_I = 1000$, and $d_X = 8$. (g) $r = 4$, $d_I = 2000$, and $d_X = 8$. (h) $r = 4$, $d_I = 5000$, and $d_X = 8$. (i) $r = 4$, $d_I = 10\,000$, and $d_X = 8$. (j) $r = 4$, $d_I = 20\,000$, and $d_X = 8$.

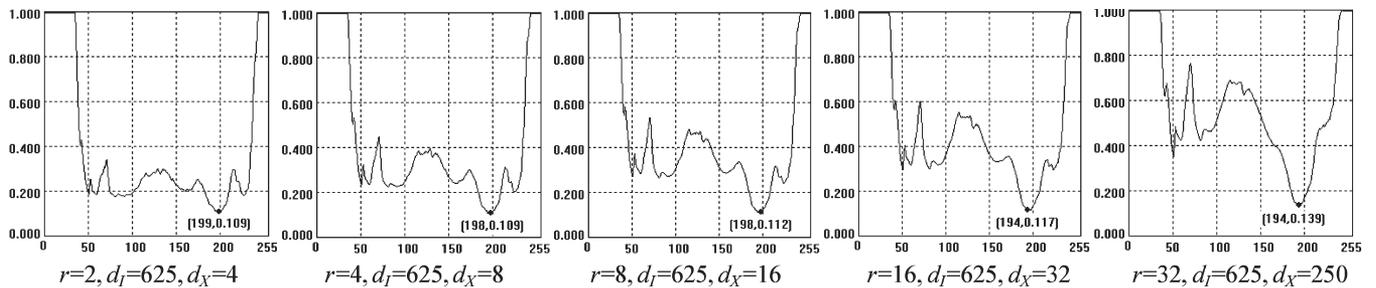


Fig. 13. Plots of $Ncut$ of the ship image with fixed value of d_I and different values of r and d_X .

to the optimal thresholding results is insignificant. The fifth row of Fig. 11 shows the results of the other algorithms with the mixed noise. Obviously, the proposed method outperforms all the other algorithms.

IV. EFFECT OF THE PARAMETERS AND COMPUTATIONAL COST

A. Effect of the Parameters

The weight of a graph edge connecting two nodes is affected by a number of parameters that must be appropriately determined. As shown in (7), these parameters include d_I , d_X , and r , where d_I controls the effect of grayscale difference between the two nodes to the weight, and d_X controls the effect of spatial position difference (or spatial distance) between the two nodes to the weight. A proper pair of parameters d_I and d_X can be selected to integrate the gray and spatial features of the pixels to effectively segment an image. In addition, parameter r determines the sparse degree of the symmetrical weight matrix \mathbf{W} . The smaller the value of r is, the sparser the matrix \mathbf{W} will be. Undoubtedly, a larger value of r more completely reveals the relationship between the nodes in a graph at the expense of higher computational cost in the calculation of weight matrix \mathbf{M} . Therefore, in order to perform real-time

processing, we must choose the value of r as small as possible to provide satisfactory segmentation results. Notice that the value of r should be chosen in conjunction with the value of d_X . Generally, a larger value of r is required when d_X is large.

Fig. 12 shows the value of $Ncut$ for the tank image corresponding to different values of d_X , d_I , and r . From these plots, we find that the effect of d_X and r on the position of the lowest valley point (i.e., the optimum threshold) is minimal. That is, the thresholding result obtained using the proposed method is relatively insensitive to parameters d_X and r . Nevertheless, the plot becomes smoother, and the lowest valley point becomes more outstanding as d_X and r increase, as seen in Fig. 12(a)–(e). When d_X and r are fixed, it is evident from Fig. 12(f)–(j) that increasing the value of d_I results in higher $Ncut$ values.

Figs. 13 and 14 show the values of $Ncut$ with respect to the threshold t for the ship image. Similar to Fig. 12(a)–(e), Fig. 13 also confirms that the thresholding result obtained by the proposed method is insensitive to changes in d_X and r .

In Fig. 14, we take a different look of the results. In this figure, different plots in the same column share the same values of d_X and r , whereas the value of d_I varies. It is seen that, as d_I increases, the optimum value of t shifts from the right valley to the left one. Each plot in the second row

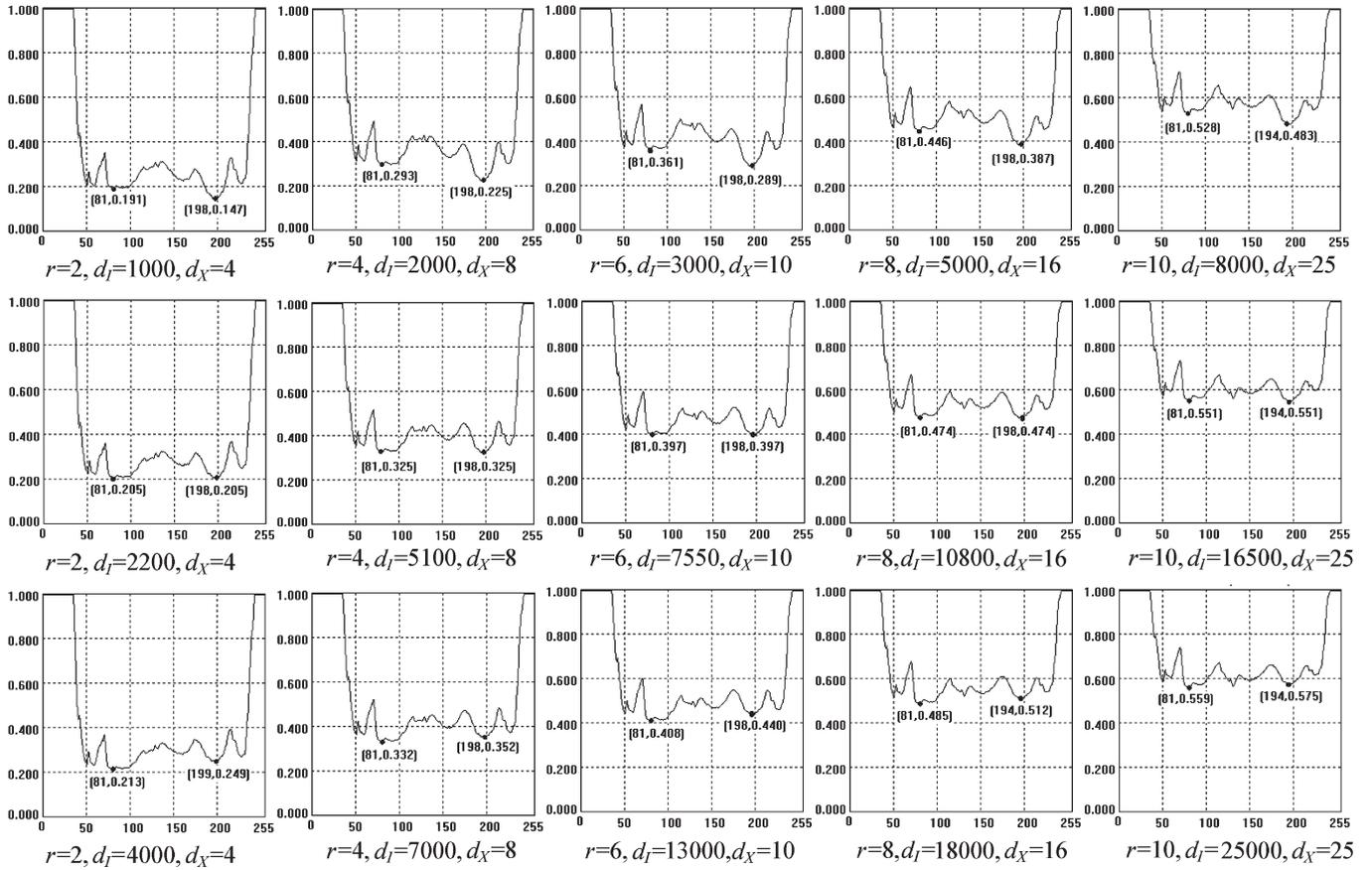


Fig. 14. Plots of $Ncut$ of the ship image as a function of threshold t using different parameter settings.

shows that both valleys have the same minimum $Ncut$ values. That is, for some specific values of d_X and r , the respective value of d_I shown in each plot of this row represents the transition point where the optimum and the second optimum threshold values exchange their positions. We also notice that the transition value of d_I increases as the values of d_X and r increase. In spite of the existence of the transition value in some applications, we maintain that the proposed method is relatively insensitive to parameters r , d_X , and d_I because the transition value of d_I is usually much higher compared with the value of d_X . Empirically, the typical value of d_I ranges between 400 and 1000, whereas r ranges between 2 and 8, and d_X ranges between 4 and 30. Generally, $r = 2$ is enough in most applications.

B. Computational Cost

The main reason why the proposed method is computationally efficient lies in the fact that it is actually a thresholding method. This allows us to reduce the dimensionality of the weight matrix from $N \times N$ to $L \times L$, where L is the gray level of an image and is typically much smaller than the number of image pixels N .

The image thresholding computation using the proposed method consists of two sections. The first section T_1 is to construct the weight matrix \mathbf{M} , and the second section T_2 is to compute the value of $Ncut$ for every possible threshold

$t(0 \leq t \leq 255)$ from weight matrix \mathbf{M} . Obviously, T_2 is determined only by L and is independent of N and r , whereas T_1 is on the order of $O(r^2N)$. For an image of a moderate or large size and a gray level of 256, T_2 is much smaller than T_1 because T_2 primarily involves additions, whereas T_1 involves square and exponent operations. The complexity of a typical solution for graph-cut segmentation algorithms is $O(MN^2)$, where M is the number of arcs. For small values of r (sparse graphs), it is $O(N^3)$. In comparison, the complexity of our solution is $O(N)$. Therefore, the advantage of the proposed method is evident.

The two plots in Fig. 15 show the corresponding execution times versus the size of the Lena image and parameter r , respectively. A Pentium IV personal computer with a 1.7-GHz CPU is used. The results clearly show that the computation cost increases as parameter r and the image size increase.

It should be mentioned that the proposed method may not be suitable for floating-point data and becomes less attractive when the number of gray levels is very large. For example, for images of 12-b (i.e., 4096) levels, which are found in medical and astronomical applications, the size of matrix \mathbf{M} becomes 4095×4095 . Therefore, if the data are floating point, then the matrix will become $N \times N$ (N is the number of pixels) again. Another point that should be clarified is that, when the number of image pixels N is smaller than the number of gray levels L , the dimension of matrix \mathbf{M} becomes higher than that of matrix \mathbf{W} , and the proposed scheme is no longer advantageous.

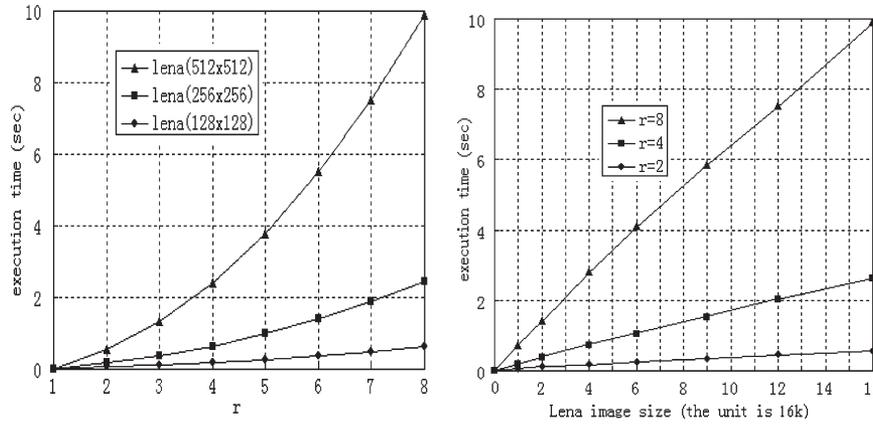


Fig. 15. (Left) Execution time versus parameter r . (Right) Execution time versus image size. The x -axis reflects the number of image pixels (the unit is 16 000).

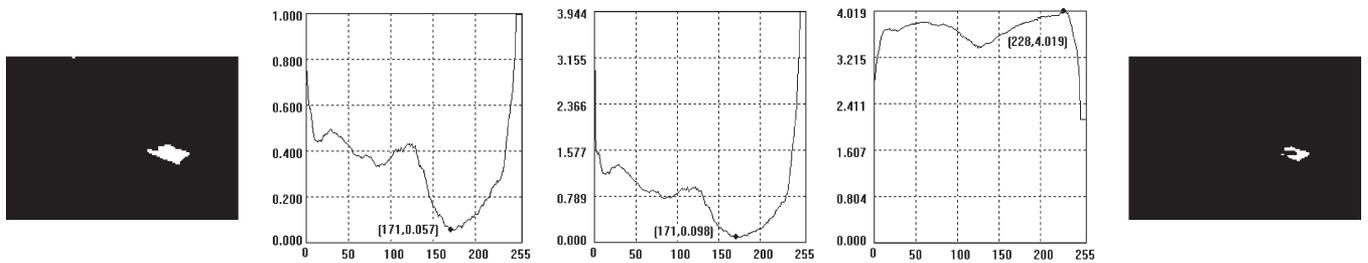


Fig. 16. Test result of the tank image. (From left to right) The result image using the proposed method based on the normalized-cut measure, the value of N_{cut} , the value of A_{cut} , the value of A_{assoc} , and the result image using the proposed method based on the average-association measure.

V. COMPARISON OF DIFFERENT GRAPH-CUT MEASURES FOR IMAGE THRESHOLDING

In some previous works, image segmentation has been formulated as a graph partition problem. For example, Wu and Leahy [13] used the minimum-cut criterion for segmentations. As we mentioned earlier, an undesirable property of this criterion is that it tends to favor cutting off small regions. In addition to the normalized-cut measure that is used in the proposed method, other graph-cut measures, such as average cut (A_{cut}) [15] and average association (A_{assoc}) [14], have also been proposed to achieve unbiased partitions. These graph-cut measures are, respectively, defined as

$$A_{cut} = \frac{cut(A, B)}{|A|} + \frac{cut(B, A)}{|B|} \tag{17}$$

$$A_{assoc} = \frac{assoc(A, A)}{|A|} + \frac{assoc(B, B)}{|B|}. \tag{18}$$

The normalized-cut formulation has certain resemblance to the average cut, the standard spectral graph partitioning algorithm, as well as the average-association formulation. Both the normalized-cut and average-cut algorithms try to find a “balanced partition” of a weighted graph. Note that the average cut does not have a simple relationship to the average association, whereas the normalized association and the normalized cut are related by $N_{assoc} = 2 - N_{cut}$ (see [14]), where $N_{assoc}(A, B) = (cut(A, A)/asso(A, V)) + (cut(B, B)/asso(B, V))$. Therefore, the normalized-cut formulation can be used to better trade off between clustering

and segmentation than the average cut and average association. The normalized cut not only avoids small sets but also assumes some degree of intensity homogeneity inside object and background and some degree of intensity discontinuity between object and background. Even in situations where the background is not homogeneous, the method seems to work well due to the high homogeneity inside the object. Judging from the discrete formulations of the grouping criteria, it can be seen that the average association has a bias, resulting in tighter clusters in the partitioning. Therefore, it runs the risk of becoming too greedy in finding small, but tight, clusters in the data. This bias in grouping has undesirable consequences, as we will illustrate in the examples hereafter. For average cut, on the other hand, one cannot ensure that the two partitions computed have tight intragroup similarity. In fact, the average cut does not avoid this unnatural bias for partitioning out small sets of points and, to a certain extent, is sensitive to individual nodes. Such sensitivity will be illustrated in the examples later. In the experiments of this part, the parameters in (7) are set to $d_I = 625$, $d_X = 4$, and $r = 2$.

Figs. 16 and 17 show the results of two test images: the tank and ship images used previously. The results show that the thresholds obtained using the N_{cut} and A_{cut} measures are almost identical. From the plots of graph cuts, the shape of A_{cut} is similar to that of N_{cut} , and the optimal threshold values for these measures are identical. On the other hand, the optimum threshold value obtained by the A_{assoc} measure is significantly different from the results obtained by the other graph-cut measures. Just like that presented earlier, the average association is apt to find tighter clusters in the graph.

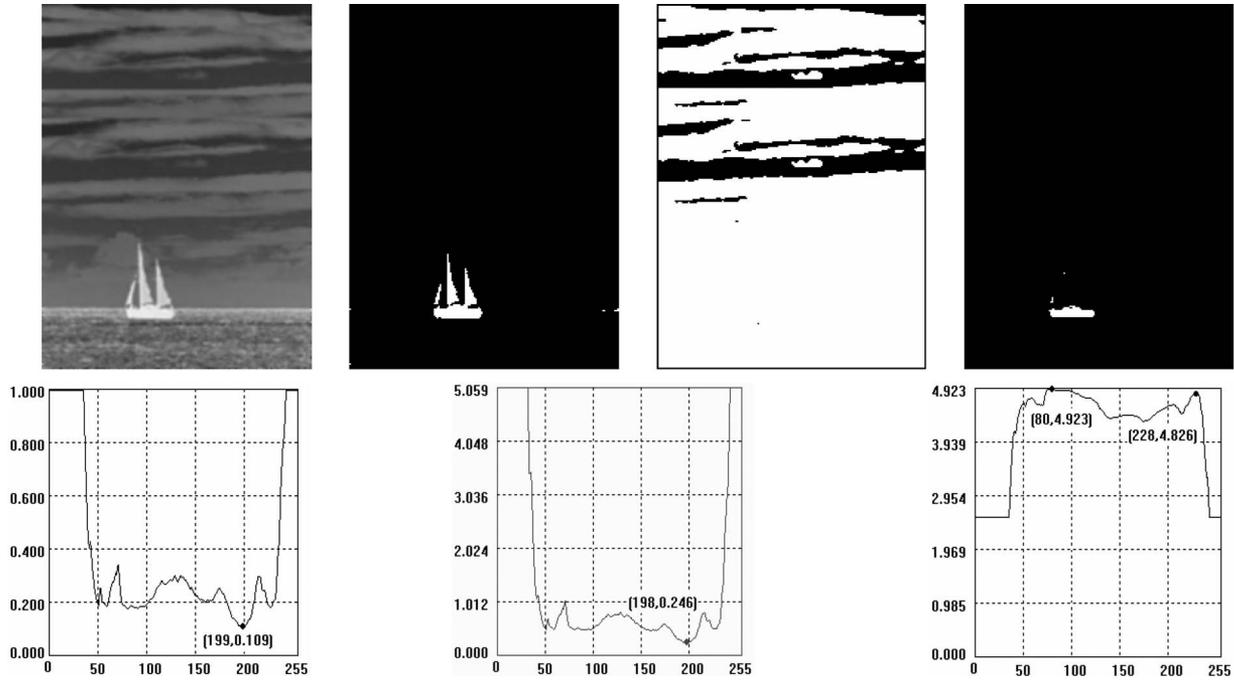


Fig. 17. (Top row, from left to right) Original ship image, the result image using the proposed method based on the normalized-cut measure with $T = 199$, the two result images using the proposed method based on the average-association measure with $T = 80$ and $T = 228$, respectively. (Bottom row, from left to right) Value of Ncut, value of Acut, and value of Aassoc.

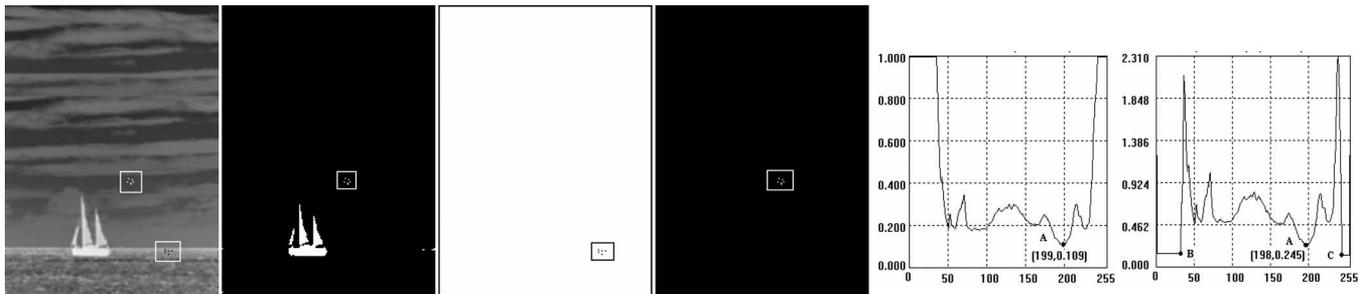


Fig. 18. (From left to right) Synthetic ship image (five bright pixels of gray scale 255 in the upper rectangle and five black pixels of gray scale 0 in the lower rectangle are added to the original ship image); thresholding results of the synthetic ship image using thresholds $T = 199$, $T = 30$, and $T = 245$, respectively; the value of Ncut for the synthetic ship image; the value of Acut for the synthetic ship image [point B is (30, 0.156), and point C is (245, 0.130)].

We demonstrate that a small set of points has much more significant effect to Acut than to Ncut using a synthetic image. In Fig. 18, a synthetic ship image is generated where five black pixels, whose gray value is 0, and six bright pixels, whose gray value is 255, are added to the original ship image. The rectangles in the second image of Fig. 18 highlight these added pixels. The results show that the plots of Ncut of the original and synthetic ship images are almost identical. That is, the effect of the small sets of points to the Ncut measure is trivial. However, the plot of Acut of the synthetic ship image is significantly different from that of the original ship image. Aside from the true valley point A, the addition of the 11 pixels yields two new valley points B and C in the Acut plot. Both point B at (30, 0.156) and point C at (245, 0.130) have lower valley values than that of point A. The corresponding thresholding results using the two threshold points B and C are shown as the middle two images of Fig. 18. The results show that the added small sets of points have great influence

on the performance of the thresholding method based on the Acut measure.

VI. CONCLUSION

In this paper, we have developed a thresholding algorithm based on the normalized-cut measure. Unlike the existing graph-cut-based image segmentation approaches which are impractical in real-time applications due to their high computational complexity, the proposed method requires significantly less computations and, therefore, is suitable for real-time vision applications, such as ATR. Significant reduction of the computational cost and memory storage are achieved by constructing a new weight matrix based on gray levels instead of pixels. In addition, the use of the normalized-cut measure as the thresholding principle enables us to distinguish an object from background without a bias. Because of the compact and fixed size of the weight matrix, we can quickly obtain the

graph-cut value for all the possible thresholding values and determine the optimum threshold values. The effectiveness of the proposed method, as well as its superiority over a number of contemporary thresholding techniques, has been confirmed by using a series of infrared and scenic images. Experimental results on real and synthetic images also showed that the normalized cut is more appropriate to act as the thresholding principle in our graph-cut-based thresholding approach than other graph-cut measures, such as average cut and average association.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments.

REFERENCES

- [1] W. K. Pratt, *Digital Image Processing*. New York: Wiley, 1978.
- [2] P. K. Sahoo, S. Soltani, and A. K. C. Wong, "A survey of thresholding techniques," *Comput. Vis. Graph. Image Process.*, vol. 41, no. 2, pp. 233–260, Feb. 1988.
- [3] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognit.*, vol. 26, no. 9, pp. 1277–1294, Sep. 1993.
- [4] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electron. Imag.*, vol. 13, no. 1, pp. 146–165, Jan. 2004.
- [5] S. U. Lee, S. Y. Chung, and R. H. Park, "A comparative performance study of several global thresholding techniques for segmentation," *Comput. Vis. Graph. Image Process.*, vol. 52, no. 2, pp. 171–190, Nov. 1990.
- [6] C. A. Glasbey, "An analysis of histogram-based thresholding algorithms," *CVGIP, Graph. Models Image Process.*, vol. 55, no. 6, pp. 532–537, Nov. 1993.
- [7] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
- [8] H. D. Cheng, Y. H. Chen, and X. H. Jiang, "Thresholding using two-dimensional histogram and fuzzy entropy principle," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 732–735, Apr. 2000.
- [9] W. Tao, H. Jin, and L. Liu, "Object segmentation using ant colony optimization algorithm and fuzzy entropy," *Pattern Recognit. Lett.*, vol. 28, no. 7, pp. 788–796, May 2007.
- [10] O. J. Tobias and R. Seara, "Image segmentation by histogram thresholding using fuzzy sets," *IEEE Trans. Image Process.*, vol. 11, no. 12, pp. 1457–1465, Dec. 2002.
- [11] S. Belkasim, A. Ghazal, and O. A. Basir, "Phase-based optimal image thresholding," *Digital Signal Process.*, vol. 13, no. 4, pp. 636–655, Oct. 2003.
- [12] P. K. Saha and J. K. Udupa, "Optimum image thresholding via class uncertainty and region homogeneity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 7, pp. 689–706, Jul. 2001.
- [13] Z. Y. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1101–1113, Nov. 1993.
- [14] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
- [15] S. Sarkar and P. Soundararajan, "Supervised learning of large perceptual organization: Graph spectral partitioning and learning automata," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 5, pp. 504–525, May 2000.
- [16] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.
- [17] S. Wang and J. M. Siskind, "Image segmentation with ratio cut," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 6, pp. 675–690, Jun. 2003.
- [18] P. Soundararajan and S. Sarkar, "An in-depth study of graph partitioning measures for perceptual organization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 6, pp. 642–660, Jun. 2003.
- [19] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.
- [20] J. Carballido-Gamio, S. J. Belongie, and S. Majumdar, "Normalized cuts in 3-D for spinal MRI segmentation," *IEEE Trans. Med. Imag.*, vol. 23, no. 1, pp. 36–44, Jan. 2004.
- [21] A. Barbu and S. C. Zhu, "Graph partition by Swendsen–Wang cuts," in *Proc. Int. Conf. Comput. Vis.*, Nice, France, 2003, pp. 320–327.
- [22] W. Tao, H. Jin, and Y. Zhang, "Color image segmentation based on mean shift and normalized cuts," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 5, pp. 1382–1389, Oct. 2007.
- [23] Y. Weiss, "Segmentation using eigenvectors: A unifying view," in *Proc. Int. Conf. Comput. Vis.*, 1999, pp. 957–982.
- [24] E. Sharon, A. Brandt, and R. Basri, "Fast multiscale image segmentation," in *Proc. CVPR*, Hilton Head Island, SC, 2000, vol. 1, pp. 70–77.
- [25] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the Nystrom method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 214–225, Feb. 2004.
- [26] S. X. Yu and J. Shi, "Segmentation given partial grouping constraints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 173–183, Feb. 2004.
- [27] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Proc. Int. Conf. Comput. Vis.*, 2001, pp. 508–515.
- [28] R. Kannan, S. Vempala, and A. Veta, "On clusterings—Good, bad and spectral," in *Proc. Annu. Symp. Found. Comput. Sci.*, 2000, pp. 367–377.
- [29] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*, vol. 14. Cambridge, MA: MIT Press, 2001.
- [30] N. Ramesh, J. H. Yoo, and I. K. Sethi, "Thresholding based on histogram approximation," *Proc. Inst. Elect. Eng.—Vision, Image Signal Process.*, vol. 142, no. 5, pp. 271–279, Oct. 1995.
- [31] J. Kittler and J. Illingworth, "On threshold selection using clustering criteria," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 5, pp. 652–655, Sep/Oct. 1985.
- [32] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram," *Comput. Vis. Graph. Image Process.*, vol. 29, no. 3, pp. 273–285, Mar. 1985.
- [33] A. Pikaz and A. Averbuch, "Digital image thresholding, based on topological stable-state," *Pattern Recognit.*, vol. 29, no. 5, pp. 829–843, May 1996.
- [34] N. R. Pal and S. K. Pal, "Entropic thresholding," *Signal Process.*, vol. 16, no. 2, pp. 97–108, Feb. 1989.
- [35] J. Kittler, J. Illingworth, and J. Foglein, "Threshold selection based on a simple image statistic," *Comput. Vis. Graph. Image Process.*, vol. 30, no. 2, pp. 125–147, May 1985.
- [36] M. H. F. Wilkinson, "Gaussian-weighted moving-window robust automatic threshold selection," in *Proc. CAIP*, Groningen, The Netherlands, 2003, pp. 369–376.
- [37] D. Wagner and F. Wagner, "Between min cut and graph bisection," TU Berlin, Berlin, Germany, Algorithmic Discrete Mathematics Tech. Rep. 307/1991, Sep. 1991.
- [38] A. Pothen and H. Simon, "Partitioning sparse matrices with eigenvectors of graphs," *IBM J. Res. Develop.*, pp. 420–425, 1973.
- [39] M. H. F. Wilkinson, "Optimizing edge detectors for robust automatic threshold selection: Coping with edge curvature and noise," *Graph. Models Image Process.*, vol. 60, no. 5, pp. 385–401, Sep. 1998.



Wenbing Tao received the Ph.D. degree in pattern recognition and intelligent system from Huazhong University of Science and Technology (HUST), Wuhan, China, in 2004.

Since 2005, he has been with the School of Computer Science and Technology, HUST, where he is currently a Research Associate Professor with the Service Computing Technology and System Laboratory and also with the Cluster and Grid Computing Laboratory. His research interests include computer vision, image segmentation, object recognition and

tracking, image search engines, and multimedia retrieval. He has published numerous articles and conference papers in the areas of image processing and object recognition.

Dr. Tao serves as a Reviewer for the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: PART A; *Image and Vision Computing*; *Pattern Recognition Letters*; *Signal Processing*; *Parallel Computing*; *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*; and *International Journal of Computers and Applications*.



Hai Jin (M'99–SM'06) received the Ph.D. degree in computer engineering from Huazhong University of Science and Technology (HUST), Wuhan, China, in 1994.

He was with The University of Hong Kong, Hong Kong, between 1998 and 2000, and was a Visiting Scholar at the University of Southern California, Los Angeles, between 1999 and 2000. He is a Professor of computer science and engineering with HUST, where he is currently the Dean of the School of Computer Science and Technology. He is also with the Cluster and Grid Computing Laboratory and the Service Computing Technology and System Laboratory of Ministry of Education, HUST. He has coauthored 8 books and published over 300 research papers. His research interests include computer architecture, virtualization technology, cluster computing and grid computing, peer-to-peer computing, network storage, and network security.

Dr. Jin was awarded the German Academic Exchange Service Fellowship to visit the Technical University of Chemnitz, Chemnitz, Germany, in 1996. He was awarded the Excellent Youth Award from the National Science Foundation of China in 2001. He is the Steering Committee Chair of the International Conference on Grid and Pervasive Computing and the Asia-Pacific Services Computing Conference. He is a member of the steering committee of the IEEE/ACM International Symposium on Cluster Computing and the Grid, the IFIP International Conference on Network and Parallel Computing, the International Conference on Grid and Cooperative Computing, the International Conference on Autonomic and Trusted Computing, and the International Conference on Ubiquitous Intelligence and Computing. He is the Chief Scientist of ChinaGrid, the largest grid computing project in China. He is a member of the Association for Computing Machinery.



Yimin Zhang (SM'01) received the Ph.D. degree from the University of Tsukuba, Tsukuba, Japan, in 1988.

He was a Member of the Faculty in the Department of Radio Engineering, Southeast University, Nanjing, China, in 1988. He served as the Technical Manager of Communication Laboratory Japan, Kawasaki, Japan, from 1995 to 1997, and was a Visiting Researcher at ATR Adaptive Communications Research Laboratories, Kyoto, Japan, from 1997 to 1998. Since 1998, he has been with Villanova University, Villanova, PA, where he is currently a Research Associate Professor with the Center for Advanced Communications. His research interests include statistical signal and array processing for communications and radar applications, including digital mobile communications, wireless networks, multiple-input–multiple-output systems, cooperative diversity, blind signal processing, time–frequency analysis, source localization and tracking, radio-frequency identification, and image processing.

Dr. Zhang serves as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS and an Associate Editor for the *Journal of the Franklin Institute*.



Liman Liu received the B.S. degree in biomedical engineering and the M.S. degree in electrical information engineering from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2002 and 2005, respectively.

Since 2006, she has been with O₂Micro Corporation, Wuhan, where she is currently an IC Design Engineer. Her research interests include signal processing, mobile communications, image processing, and computer vision.

Ms. Liu has received the Excellent Graduate Award from HUST in 2002 and 2005.



Desheng Wang received the M.S. degree in mathematics from Xiangtan University, Xiangtan, China, in 1997, and the Ph.D. degree in computational mathematics from the Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing, in 2001.

He held a research position at Swansea University, Swansea, U.K., from 2003 to 2005. Currently, he is an Assistant Professor with the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological

University, Singapore. His research interests include partial–differential equation and graph-cut methods for image processing and computer vision, mesh generation, finite-element method, and computational electromagnetics.