A NEW IMAGE THRESHOLDING METHOD BASED ON GRAPH CUTS

Wenbing Tao[1], Hai Jin[1], Liman Liu[2]

[1] Service Computing Technology and System Laboratory of Ministry of Education,
Huazhong University of Science and Technology, Wuhan, 430074, China
[2] O_2Micro, Wuhan, 430074, China

ABSTRACT

A novel thresholding algorithm is presented to achieve improved image segmentation performance at low computational cost in this paper. The proposed algorithm uses a normalized graph cut measure as the 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 an image, rather than the commonly used image pixels. Therefore, the proposed algorithm occupies much smaller storage space and requires much lower computational costs and implementation 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 (ATR). A large number of examples are presented to show the superior performance of the proposed thresholding algorithm compared to existing thresholding algorithms.

Index Terms—Image processing, segmentation, object recognition, graph theory

1. INTRODUCTION

Thresholding offers an efficient way, in terms of both the processing time and the implementation simplicity, to perform image segmentations. However, automatic determination of the optimum threshold value is often a difficult task in digital image processing. The latest development in this topic was summarized in [1]. Image segmentation approaches based on graph cuts, in general, have high computation complexity and poor real-time performance. Therefore, they are usually impractical in many image segmentation problems. In this paper, nevertheless, 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. Unlike Shi et al.[2] who developed general image partitioning approaches by solving eigen system, the purpose of this paper is to develop a simple thresholding approach based on graph cuts. In the proposed thresholding approach, a significant reduction in the computation cost is achieved by representing a graph using a 256x256 symmetrical weight matrix based on gray levels, rather than the $N \times N$ symmetrical weight matrix based on the pixels, where N is the number of the pixels in the image. Because the size of the weight matrix based on the gray levels is very small, we can quickly obtain the graph cut values for every possible threshold t from this weight matrix.

2. THRESHOLDING BASED ON GRAPH CUTS

2.1. BACKGROUND

The set of points in an arbitrary feature space is represented as a weighted undirected graph **G**=(**V**,**E**), where **V** is the set of vertices and **E** is the set of edges. An edge is formed between each pair of nodes and the weight on the edge, $w(u,v)$, is a function of the similarity between two nodes u and v. A graph **G** is partitioned into two disjoint complementary sets A and B, where B=**V**-A, which closely relates to a mathematical formulation of a cut [3]:

$$cut(A,B) = \sum w(u,v), (u \in A, v \in B) \ . \qquad (1)$$

Wu and Leahy proposed a clustering method based on the minimum cut criterion [3]. While this method usually produces good segmentation results, the minimum cut criterion favors cutting small sets of isolated nodes in a graph. To avoid this unnatural bias in favor of cutting small sets of nodes, Shi and Malik [2] proposed a new measure of disassociation between two groups, which is referred to as the normalized cuts (*Ncut*), expressed as

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

where $asso(A,V) = \sum w(u,t), (u \in A, t \in V)$ is the total connection from nodes in A to all nodes in the graph, and $asso(B,V)$ can be similarly defined. Using the *Ncut* criterion, the cut no longer has bias in favor of cutting small sets of nodes. In this case, equation (2) can be transformed into the following standard eigen-system [2]

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

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

2.2. PROPOSED APPROACH

Let $V = \{(i,j): i = 0,1,\cdots,n_h - 1; j = 0,1,\cdots,n_w - 1\}$, $L = \{0,1,\cdots,255\}$, 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). V and $f(x,y)$ satisfy:

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

$$V_k = \{(x,y): f(x,y) = k, (x,y) \in V\}, k \in L \tag{5}$$

$$\bigcup_{k=0}^{255} V_k = V, \ V_j \cap V_k = \Phi; k \neq j; k, j \in L \tag{6}$$

Construct a weighted graph **G**=(**V**,**E**) by taking each pixel as a node and connecting each pair of pixels by an edge. 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 following:

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

where F(u) is the gray scale of node u, X(u) is the spatial location of node u, and $\|.\|_2$ denotes vector norm. In addition, d_I and d_X are positive 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.

For any t ($0 \leq t < 255$), we can obtain a unique bisection **V**={A,B} of the corresponding graph **G**=(**V**,**E**), where sets A and B can be formulated as

$$A = \bigcup_{k=0}^{t} V_k, \ B = \bigcup_{k=t+1}^{255} V_k, \ k \in L \tag{8}$$

Then equation (1) becomes:

$$\begin{aligned} cut(A,B) &= \sum_{u \in A, v \in B} w(u,v) = \sum_{u \in A} [\sum_{v \in B} w(u,v)] \\ &= \sum_{i=0}^{t} \sum_{u \in V_i} [\sum_{j=t+1}^{255} \sum_{v \in V_j} w(u,v)] = \sum_{i=0}^{t} \sum_{j=t+1}^{255} [\sum_{u \in V_i, v \in V_j} w(u,v)] = \sum_{i=0}^{t} \sum_{j=t+1}^{255} cut(V_i, V_j), \end{aligned} \tag{9}$$

where

$$cut(V_i, V_j) = \sum_{u \in V_i, v \in V_j} w(u,v) \tag{10}$$

is the total connection between all nodes in V_i (whose gray level is i) and all nodes in V_j (whose gray level is j).

Similarly,

$$asso(A,A) = \sum_{u \in A, v \in A} w(u,v) = \sum_{i=0}^{t} \sum_{j=i}^{t} [\sum_{u \in V_i, v \in V_j} w(u,v)] = \sum_{i=0}^{t} \sum_{j=i}^{t} cut(V_i, V_j) \tag{11}$$

$$asso(B,B) = \sum_{u \in B, v \in B} w(u,v) = \sum_{i=t+1}^{255} \sum_{j=i}^{255} [\sum_{u \in V_i, v \in V_j} w(u,v)] = \sum_{i=t+1}^{255} \sum_{j=i}^{255} cut(V_i, V_j) \tag{12}$$

Equation (2) becomes

$$Ncut(A,B) = \frac{cut(A,B)}{asso(A,A) + cut(A,B)} + \frac{cut(A,B)}{asso(B,B) + cut(A,B)} \tag{13}$$

Let **M** be the 256x256 symmetrical matrix with $m_{i,j} = cut(V_i, V_j)$ as its (i, j)-th element and $m_{i,j} = m_{j,i}$, it can be uniquely constructed for a given image by computing all the weights on the edges connecting each pair of pixels in the image. Matrix **M**=$[m_{i,j}]_{256 \times 256}$ is illustrated in Fig.1 (Only the elements in upper triangle are shown due to symmetry).

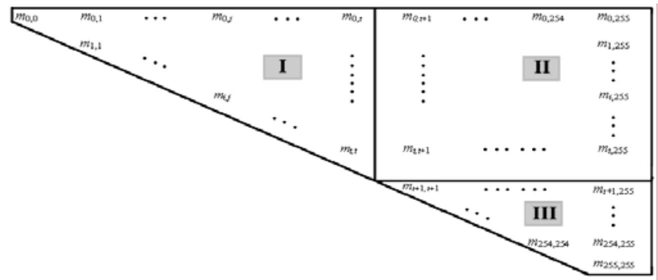

Fig.1 Graph illustration of symmetrical matrix **M**=$[m_{i,j}]_{256\text{x}256}$ with $m_{i,j}$=$m_{j,i}$

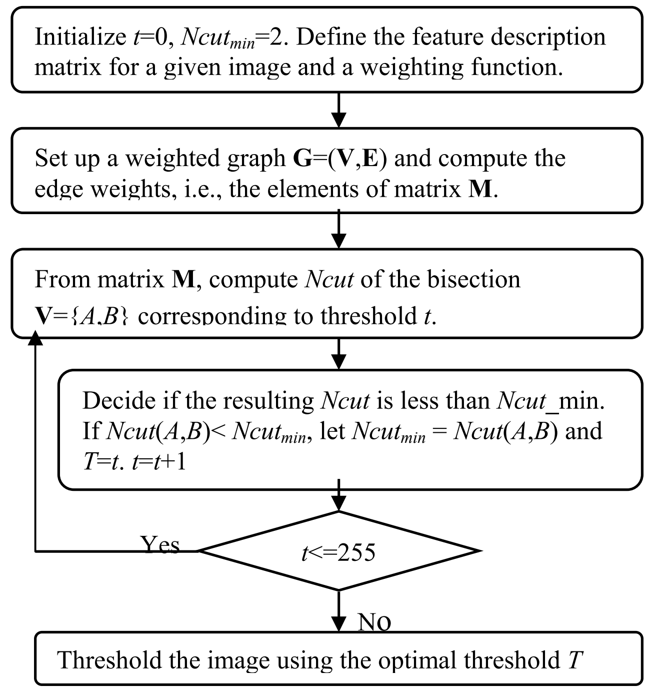

Fig. 2 The implementation flow chart of the proposed

Note that, when we partition the image using the normalized cut measure, matrix **M** of size 256x256 suffices, whereas the information of NxN matrix **W** is not necessary. From matrix **M**, we can compute $cut(A,$B), $asso(A,A)$ and $asso(B,B)$ of the corresponding bisection of the weight graph for each threshold t. As shown in Fig. 1, the elements of matrix **M** are divided into three parts. The sum of all the elements in part I is the value of $asso(A,A)$, whereas the sum of all the elements in part II is the value of $cut(A,B)$, and the sum of all the elements in part III yields the value of $asso(B,B)$. Therefore, the normalized cuts can be easily computed from **M** for every possible threshold t. Moreover, unlike matrix **W**, whose dimension NxN depends on the image size, **M** is a symmetrical matrix of fixed size 256x256 irrespective to the image size.

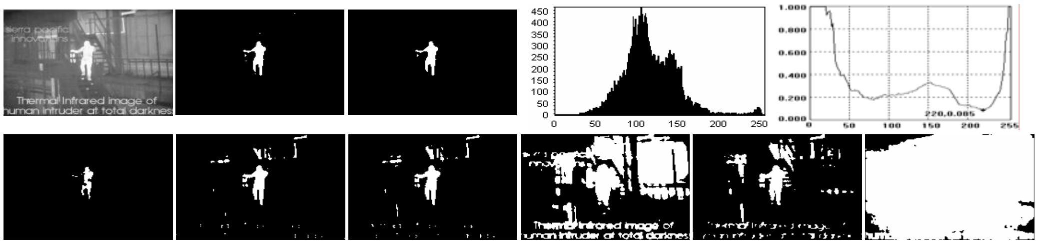

Fig.3: Top row, from left to right: original intruder image (185x141), thresholding result by the proposed method (*T*=220), manually thresholding image (*T*=221), histogram of the original image, value of *Ncut* versus threshold *t*. Bottom row, from left to right: results by Pikaz method (*T*=241), Kittler method (*T*=180), Kapur method (*T*=176), Yanowitz method (*T*=123), Ramesh method (*T*=154), and Pal method (*T*=74).

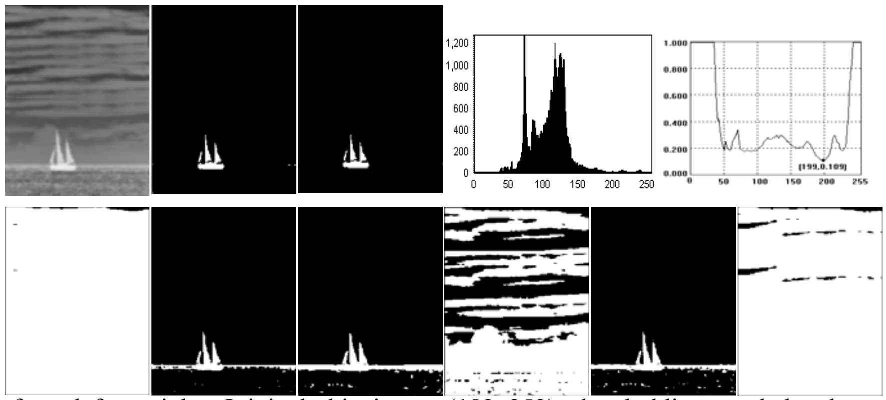

Fig.4: Top row, from left to right: Original ship image (182x253), thresholding result by the proposed method (*T*=199), manually thresholding image(*T*=201), histogram of the original image, value of *Ncut* versus threshold *t*. Bottom row, from left to right: result by Pikaz method (*T*=51), Kittler method(*T*=165), Kapur method (*T*=148), Yanowitz method (*T*=104), Ramesh method (*T*=155), and Pal method (*T*=66).

The proposed thresholding method searches the optimal threshold value that minimizes the corresponding normalized cuts of the image. Let T be the optimum threshold, and $0 \le t \le 255$ be a threshold variable. Let $\mathrm{Ncut}_{\min}$ be the minimum value of the normalized cut. The proposed algorithm is summarized as Fig. 2:

3. PERFORMANCE EVALUATION AND COMPARISON WITH EXISTING METHODS

To make the performance evaluation and comparison meaningful and effective, a set of real images was used to evaluate the performance of the proposed algorithm as well as some of the commonly used algorithms presented in the literature. Each image includes distinct object and the background, and the object can be exactly distinguished from the background by some suitable threshold. Especially, some infrared object images are selected to examine our algorithm for the reason that infrared sensors have good night vision performance and are widely applied in automatic target recognition (ATR). Infrared images reflect the thermal radiations of the targets and the background, and the objects can be distinguished by their gray levels. In all the examples used in this section, the parameters in equation (7) are set to $d_I = 625$, $d_X = 4$, and $r = 2$.

Sezgin and Sankur [1] classified thresholding algorithms into the following six categories based on the type of information used. The six categories thresholding methods are repectively: 1. Histogram shape-based thresholding methods, 2. Clustering-based thresholding methods, 3. Entropy-based thresholding methods, 4. Thresholding method based on attribute similarity, 5. Spatial thresholding methods, and 6. Locally adaptive thresholding. 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 based on graph cuts. The six selected thresholding algorithms are: Ramesh method [4], Kittler method [5], Kapur method [6], Pikaz method [7], Pal method [8], and Yanowitz method [9].

The results of examples are shown in Figs. 3-4. 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 *Ncut* as a function of threshold *t*. The images used for comparison include a night infrared image with an intruder (Fig. 3), an offing infrared image with one ship object (Fig. 4). For each image, the respective optimum threshold value T is chosen as the one corresponding to the minimum value of Ncut. Note that, in the examples, the proposed algorithm can effectively extract the infrared object from the background and the performance is close to the optimum one obtained by manual thresholding. Compared with the other methods, the proposed method has better performance.

4. EFFECT OF THE PARAMETERS AND COMPUTATIONAL COST

The weight of a graph edge connecting two nodes is affected by a number of parameters that must be appropriately determined. As shown in equation (7), these parameters include d_I, d_X, and r, where d_I controls the effect of gray scale difference of the two nodes to the weight, and d_X controls the effect of spatial position difference (or spatial distance) of two nodes to the weight. A properly selected combination of parameters d_I and d_X can be used to integrate the gray feature and spatial feature of the pixels to effectively segment an image. In addition, parameter r determines the sparse degree of the symmetrical weight matrix **W**. The smaller the value of r is, the sparer the matrix **W** will be. Undoubtedly, a larger value of r reveals a more complete relationship between the nodes in a graph, at the expense of more computational time in calculating the weight matrix **M**. Therefore, in order to perform real-time processing we must choose the value of r as small as possible yet 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. Experientally, the typical value of d_I ranges between 400 and 1000, r ranges between 2 and 8, and d_X ranges between 4 and 30, respectively. Generally, r=2 is enough in most applications.

The proposed method is actually a thresholding method, which is the main reason of the computation efficiency. On the other hand, we develop a 256x256 matrix based on which the graph cut value for every possible threshold is computed instead of an NxN matrix, as also enhances the computation efficiency. The experimental results shows that the computational time is less than 50ms with parameter r=2 for the images of 256x256. All experiments are carried out on a Pentium PC of 1.7GHz CPU.

5. CONCLUSION

In this paper, we have developed a thresholding algorithm based on the normalized cut measure. Unlike the existing image segmentation approaches based on graph cuts which are impractical in many real-time application due to their high computation complexity, the proposed method requires much less computations and, therefore, is suitable for real-time vision applications, such as automatic target recognition (ATR). Significant reduction of the computational cost and memory storage are achieved by constructing the weight matrix based on the gray levels, rather than the 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 have been confirmed by using a series of infrared images and other scenes.

6. ACKNOWLEDGEMENT

The work was supported by the National Natural Science Foundation of China (Grant No. 60603024) and the National Science Foundation for Post-doctoral Scientists of China (Grant No. 2005037198).

7. REFERENCES

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

[2] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 22, no. 8, pp. 888-905, 2000.

[3] Z.Y. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol.15, no.11, pp.1101-1113, 1993.

[4] N. Ramesh, J. H. Yoo, and I. K. Sethi, "Thresholding based on histogram approximation," *IEE Proc. Vision Image Signal Process*. vol. 142, no. 5, pp. 271–279, 1995.

[5] J.Kittler and J.Illingworth, "On threshold selection using clustering criteria," *IEEE Trans. Syst. Man Cybern.* vol. 15, pp. 652–655, 1985.

[6] 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," *Graph. Models Image Process.*, vol.29, pp.273–285, 1985.

[7] A.Pikaz and A.Averbuch, "Digital image thresholding based on topological stable state," *Pattern Recogn.*, vol.29, pp.829–843, 1996.

[8] N. R. Pal and S. K. Pal, ''Entropic thresholding,'' *Signal Process.*, vol.16, pp.97–108, 1989.

[9] S. D. Yanowitz and A. M. Bruckstein, "A new method for image segmentation," *Comput. Graph. Image Process.*, vol.46, pp.82–95, 1989.