# Graph-Cut Methods for Grain Boundary Segmentation

Song Wang, Jarrell Waggoner, and Jeff Simmons

*This paper reviews the recent progress on using graph-cut methods for image segmentation, and discusses their applications to segmenting grain boundaries from materials science images.*

## INTRODUCTION

The size and shape of crystals (i.e., grains) in polycrystalline materials (e.g., metals and metal alloys) are among the strongest determinants of many material properties, such as mechanical strength or fracture resistance. In materials science research, the emerging practice involves constructing three-dimensional (3-D) models of the grain structure and then applying finite element modeling to infer the mechanical properties from that structure.[1,2] Accurately and automatically segmenting the grain boundaries from various material images can substantially facilitate the modeling and analysis of the material microstructure, and shorten the period of design and development of new materials.

Image segmentation is a fundamental problem in computer vision and image processing. In past decades, many image segmentation algorithms and tools have been developed and are used to process images in different domains, such as pictures taken indoors and outdoors, aerial images, medical images, and videos. Particularly, graph-cut methods for image segmentation have attracted tremendous interest in the computer vision community in recent years. Compared to classical image-segmentation methods, such as edge detection, region splitting/merging, and pixel clustering, graph-cut methods are more "global" by considering well-defined, comprehensive segmentation cost functions and seeking their globally optimal solutions using advanced graph theory.

## GRAPH AND MINIMUM CUT

In graph-cut methods, a graph $G = (V, E)$ with vertices $V$ and edges $E$ is first constructed to represent an image. Considering a two-dimensional (2-D) image, we can construct a vertex for each pixel and an edge between two vertices corresponding to two neighboring pixels, as shown in Figure 1. A graph cut divides the graph into two subgraphs $G_1$ and $G_2$ by removing all the edges connecting $G_1$ and $G_2$. Two examples are shown in Figure 1b and d, where the removal of the edges intersected by the dashed curve constitutes a graph cut. A graph cut corresponds to a segmentation boundary (either open or closed) in the image. Multiple-region image segmentation can be obtained by repeated graph cuts on the subgraphs $G_1$ and $G_2$.

Image-intensity information is typically encoded into an edge-weight function $w(u, v)$, where $(u, v) \in E$. For example, we can define $w(u, v)$ as a decreasing function of the intensity difference between vertices (pixels) $u$ and $v$. This way, a graph cut that removes low-weight edges is more preferred for image segmentation. In graph-cut methods, the two central problems are: 1) defining a cost function for each possible graph cut to reflect the aforementioned preference, and 2) developing a graph algorithm to find the optimal graph cut that minimizes this cost function. In Reference 3, the cost function is defined to be the total weight of the removed edges, i.e.

$$c(G_1, G_2) = \sum_{u \in G_1, v \in G_2, (u,v) \in E} w(u,v)$$

This is the well-known minimum cut problem, and its global optima can be efficiently found by the min-cut max-flow algorithm. However, image segmentation using minimum cut has a bias toward producing shorter boundaries.[3]

## GRAPH CUTS WITH NORMALIZED COSTS

Various kinds of normalization have been incorporated in graph-cut cost

### How would you…

**…describe the overall significance of this paper?**

*This paper introduces new methods for accurately and automatically segmenting the grain boundaries from various material images, which can substantially facilitate the modeling and analysis of the material microstructure, and shorten the period of design and development of new materials.*

**…describe this work to a materials science and engineering professional with no experience in your technical specialty?**

*In this paper, we suggest the use of graph-cut methods for material image segmentation. In these methods, an image is modeled by a graph which considers both intensity and spatial relations of the pixels. New approaches are also introduced to enforce the segmentation continuity between neighboring slices in a sequence of 2-D serial sections.*

**…describe this work to a layperson?**

*Of great importance in studying materials is to extract the boundaries of the microstructures that make up a material. This process, called segmentation, is often done by hand, or with various rudimentary software tools on various material images. In this paper, we describe more advanced graph-cut methods recently investigated in the computer vision community for automatic microstructure segmentation.*

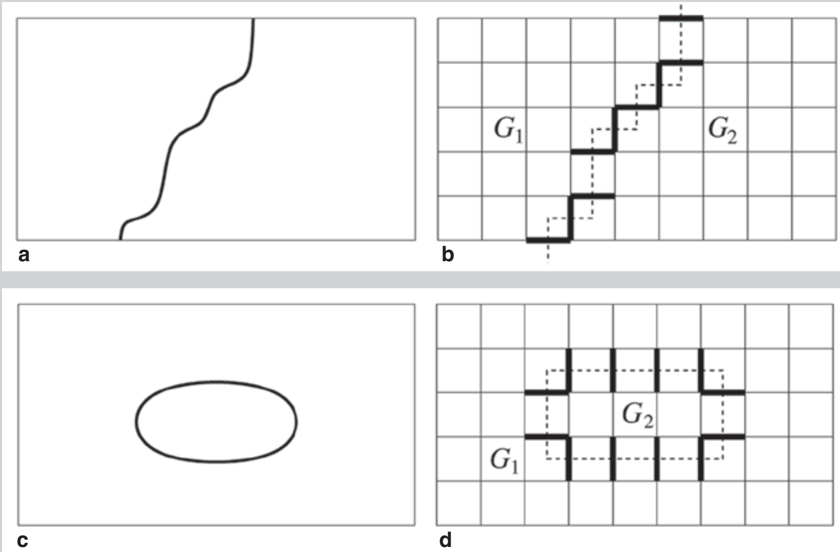



Figure 1. Image segmentation and graph cuts. (a) An image segmented by an open boundary. (b) The graph model for (a) and the graph cut for its segmentation boundary. (c) An image segmented by a closed boundary. (d) The graph model for (c) and the graph cut for its segmentation boundary.

functions to reduce the bias toward producing shorter boundaries. Shi and Malik[4] developed a normalized-cut approach with a cost function

$$\frac{c(G_1,G_2)}{c(G_1,G)}+\frac{c(G_1,G_2)}{c(G_2,G)}$$

which favors two subgraphs with similar total edge weights and, therefore, avoids producing overly short boundaries. Finding the globally optimal normalized cut is a computationally intractable NP-hard problem. In practice, an approximate solution can be obtained using spectral graph theory, by relaxing this discrete optimization problem to the continuous domain.

Wang and Siskind[5] developed a ratio-cut approach with a cost function

$$\frac{c(G_1,G_2)}{\#(G_1,G_2)}$$

where $\#(G_1,G_2)$ is the number of edges connecting $G_1$ and $G_2$. This reflects the average weight along a cut boundary and, therefore, reduces the bias toward short boundaries. The globally optimal ratio cut can be found efficiently in polynomial time. Figure 2 shows sample segmentation results using normalized cut and ratio cut.

Similar to normalized cut is the average-cut approach,[6] where the size of subgraphs $G_1$ and $G_2$ is used for normalization. Finding the optimal average cut is NP-hard and an approximate solution can be obtained using spectral graph theory. Cox, Rao, and Zhong[7] developed a ratio-region approach for image segmentation, which handles only the closed-cut boundary as shown in Figure 1d and uses the area within the closed-cut boundary for normalization. Jermyn and Ishikawa[8] extended the ratio-region approach by considering region information other than the area. The optimal ratio region and its extension in Reference 8 can be found efficiently in polynomial time. Many other graph-cut methods have also been developed for image segmentation.[9–12]

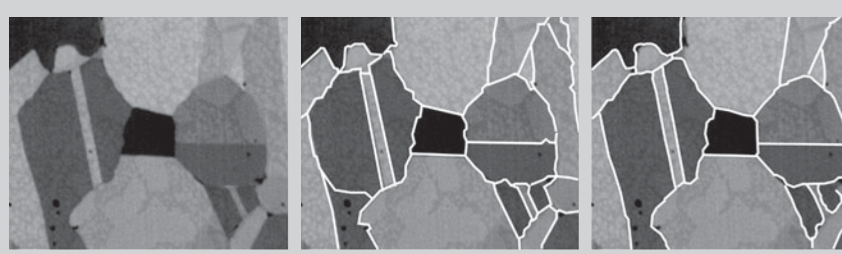
## SEGMENTATION WITH A TEMPLATE

Many materials science images are in the form of a sequence of 2-D serial sections (i.e., slices). The grain structure between neighboring slices usually shows good consistency and continuity. Therefore, we can use the segmentation of one slice as a template to guide the segmentation of its neighboring slices. Graph-cut methods, particularly minimum cut, can be used for this purpose.

Specifically, image segmentation can be formulated as an assignment of labels to each pixel where the pixels with the same label constitute a segment. In References 13 and 14, the objective of optimal labeling is to find a labeling function, *f*, by minimizing the cost function

$$E(f)=\sum_{p\in P} D_p(f_p) + \sum_{(p,q)\in N} V_{pq}(f_p,f_q)$$

where $P$ is the set of image pixels, $f_p$ is the label of pixel $p \in P$, and $N$ is the set



a b c

Figure 2. Segmentations on a (a) (cropped) Ni-based alloy image using (b) normalized cut and (c) ratio cut.

a b c

Figure 3. Segmentation of a (cropped) Ni-based alloy image by optimal labeling. (a) Segmented grain boundaries in a template slice. (b) Grain boundaries in (a) copied to a new slice. (c) Segmented grain boundaries in this new slice after optimal labeling.

of pixel pairs that are neighbors in the image. The data term $D_p(f_p)$ describes the cost of assigning label $f_p$ to pixel $p$ and the smoothness term $V_{pq}(f_p,f_q)$ describes the cost of assigning labels $f_p$ and $f_q$ to two neighboring pixels $p$ and $q$, respectively. Finding the globally optimal labeling that minimizes this cost function is NP-hard. However, the minimum graph-cut algorithm can be used to obtain a locally optimal labeling efficiently.

In this optimal labeling framework, we can define a specialized data term and smoothness term to enforce the grain structure consistency when propagating an image segmentation from one slice to another. For example, in defining the data term, we can set an infinity cost $D_p(f_p)$ when label $f_p$ is not assigned to any pixel near $p$ in the template. In defining the smoothness term, we can set an infinity cost $V_{pq}(f_p,f_q)$ if the segments with labels $f_p$ and $f_q$ are not adjacent in the template. Figure 3c shows a sample segmentation result using this method. Various kinds of human interaction, such as the selection of seed points for individual grains, can be effectively and conveniently integrated into this framework.[15]

## CONCLUSION

Graph-cut methods have been successfully used in many image-segmentation applications. Compared with other image segmentation methods, graph-cut methods employ global cost functions, attempt to find globally optimal solutions to these cost functions, and have the capability of considering both image information, such as intensity, and structure information, such as a template. These properties make graph-cut methods a very promising solution to the challenging problem of grain boundary segmentation.

### References


1. S. Li and S. Ghosh, *Finite Elements in Analysis and Design,* 43 (2007), pp. 397–410.
2. A.D. Rollett, S. Lee, and R.A. Lebensohn, *Microstructure and Texture in Steels and Other Materials* (New York: Springer, 2009), pp. 255–264.
3. Z. Wu and R. Leahy, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 15 (11) (1993), pp. 1101–1113.
4. J. Shi and J. Malik, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 22 (8) (2000), pp. 888–905.
5. S. Wang and J.M. Siskind, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 25 (6) (2003), pp. 675–690.
6. S. Sarkar and P. Soundararajan, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 22 (5) (2000), pp. 504–525.
7. I. Cox, S.B. Rao, and Y. Zhong, *International Conference on Pattern Recognition* (Hoboken, NJ: IEEE, 1996), pp. 557–564.
8. I.H. Jermyn and H. Ishikawa, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 23 (10) (2001), pp. 1075–1088.
9. O. Veksler, *IEEE Conference on Computer Vision and Pattern Recognition* (Hoboken, NJ: IEEE, 2000), pp. 339–344.
10. Y. Gdalyahu, D. Weinshall, and M. Werman, *IEEE Conference on Computer Vision and Pattern Recognition* (Hoboken, NJ: IEEE, 1999), pp. 596–601.
11. M. Pavan and M. Pelillo, *IEEE Conference on Computer Vision and Pattern Recognition,* 1 (2003), pp. 145–152.
12. P. Felzenszwalb and D. Huttenlocher, *Int. J. Computer Vision,* 59 (2) (2004), pp. 167–181.
13. Y. Boykov, O. Veksler, and R. Zabih, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 23 (11) (2001), pp. 1222–1239.
14. V. Kolmogorov and R. Zabih, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 26 (2) (2004), pp. 147–159.
15. Y.Y. Boykov and M.-P. Jolly, *International Conference on Computer Vision,* 1 (2003), pp. 105–112.



**Song Wang and Jarrell Waggoner are with the Department of Computer Science and Engineering, University of South Carolina, 315 Main Street, Columbia, SC 29208; Jeff Simmons is with the Air Force Research Laboratory, AFRL/RXLMD-Wright Patterson AFB, 2230 10 Street, Wright-Patterson AFB, OH 45433. Prof. Wang can be reached at songwang@cec.sc.edu.**