

Interactive Grain Image Segmentation using Graph Cut Algorithms

Jarrell Waggoner^a, Youjie Zhou^a, Jeff Simmons^b, Ayman Salem^b,
Marc De Graef^c, and Song Wang^a

^aUniversity of South Carolina, Columbia, SC 29208, USA;

^bMaterials and Manufacturing Directorate, Air Force Research Labs, Dayton, OH 45433, USA;

^c Carnegie Mellon University, Department of Materials Science and Engineering, 5000 Forbes Avenue, Pittsburgh, PA, 15213, USA

ABSTRACT

Segmenting materials images is a laborious and time-consuming process and automatic image segmentation algorithms usually contain imperfections and errors. Interactive segmentation is a growing topic in the areas of image processing and computer vision, which seeks to find a balance between fully automatic methods and fully-manual segmentation processes. By allowing minimal and simplistic interaction from the user in an otherwise automatic algorithm, interactive segmentation is able to simultaneously reduce the time taken to segment an image while achieving better segmentation results. Given the specialized structure of materials images and level of segmentation quality required, we show an interactive segmentation framework for materials images that has two key contributions: 1) a multi-labeling framework that can handle a large number of structures while still quickly and conveniently allowing manual interaction in real-time, and 2) a parameter estimation approach that prevents the user from having to manually specify parameters, increasing the simplicity of the interaction. We show a full formulation of each of these contributions and example results from their application.

Keywords: image segmentation, materials volume segmentation, segmentation propagation, interactive segmentation, graph-cut approaches

1. INTRODUCTION

Interactive segmentation is a rapidly-growing area of computer vision and has seen heightened interest recently.^{1,2} While traditional segmentation seeks to identify objects/structures within an image in a fully-automated fashion, interactive segmentation, similar to active learning,³ accomplishes the goal of image segmentation while incorporating a sparse number of user interactions which are included as additional constraints or guidance in the segmentation model or algorithm. These interactions may take on different forms, and may include drawing a bounding box,⁴ roughly outlining a boundary,⁵ or drawing brush strokes inside and/or outside the object of interest.⁶⁻⁹ A desired property of an interactive segmentation approach is that the user interaction be as convenient (i.e., low cognitive load) and sparse (i.e., few in number) as possible, while simultaneously providing immediate feedback to the user on every interaction.

Many existing methods segment the object of interest using a model learned from user interactions.^{4,7,8} Other approaches incorporate interaction into morphological operations (watershed),² co-segmentation,¹⁰ or incorporate machine-learning to aid in the interactive process.^{1,11} These interactive methods have been applied to a number of domains, including natural images,⁴ medical images,¹² and neuroimages.^{2,13}

Further author information: (Send correspondence to J. Waggoner)

J. Waggoner: E-mail: waggonej@email.sc.edu, Telephone: 847-261-4747

Y. Zhou: E-mail: zhou42@email.sc.edu

J. Simmons: E-mail: jeff.simmons@wpafb.af.mil

A. Salem: E-mail: ayman.salem.ctr@wpafb.af.mil

M. De Graef: E-mail: degrae@cmu.edu

S. Wang: E-mail: songwang@cec.sc.edu, Telephone: 803-777-2487

One domain that has been unaddressed in interactive segmentation literature is materials science image segmentation, where there are no existing techniques focusing solely on segmenting materials images using an interactive approach. Materials science is especially important to the development of new metals and biomaterials, and presents unique challenges in image segmentation. First, materials images often are 3D volumes¹⁴ made up of a sequence of individual 2D image “slices,” as shown by the two sample slices in Figure 1. This large number of slices must all be segmented to fully and properly analyze the 3D structure of the material. Second, depending on the inter-slice distance, the 2D structure in two neighboring slices may show high continuity. Such inter-slice structure continuity must be considered to achieve accurate segmentation. Third, materials volumes consist of numerous substructures (e.g., “grains” in a metallic material, or “cells” in a biomaterial, etc.) with complex relationships (e.g., adjacency/nonadjacency relationships) among them that determine many desirable properties of the material.^{15,16} Existing interactive segmentation techniques often only focus on foreground-background segmentation,^{4,8} and may not scale to the large number of substructures present in materials images. Other methods may handle multiple structures,^{2,13} but do not incorporate any prior knowledge about the unique relationships among substructures in materials images.^{17,18} Finally, the imaging techniques used to obtain a materials image volume may result in significant noise or other ambiguities that increases the difficulty to segment a materials image volume in a fully-automatic fashion.

There are a number of existing, non-interactive approaches to segment materials images.^{19,20} Among the most prominent is the work of Comer et al.^{21,22} on the EM/MPM algorithm, originating from.²³ Other methods that have been specifically used on materials images include graph cut,^{24,25} stabilized inverse diffusion equations,²⁶ Bayesian methods,^{27,28} and the watershed²⁹ method. Most often, materials images are opportunistically segmented by the simplest tools available, such as thresholding,^{30,31} or out-of-the-box methods such as watershed or normalized cut. However, these methods do not incorporate any interaction for manual refinement by a user. Some of these approaches may require significant time to run; requiring the user to examine and correct problems only after the algorithm is complete may not be practical if rapid refinement is desired. Conversely, the general-purpose interactive segmentation techniques discussed previously do not incorporate any specific domain knowledge about materials images, and thus may require additional effort on the part of the user than may otherwise be needed when segmenting a materials image volume.

In this paper, we present an interactive segmentation approach to segment materials science image volumes. We show that an existing propagation-based materials image segmentation approach²⁵ can be extended to allow for convenient interactive segmentation. We illustrate the performance of the proposed approach by using it to segment a materials image volume using smaller number of interactions compared with general-purpose interactive segmentation methods that do not incorporate materials-specific priors. Finally, we develop methods to estimate the parameters of this proposed approach to further reduce the number of user-required interactions in the segmentation process.

The remainder of this paper is organized as follows: in Section 2 we discuss the proposed interactive segmentation approach for materials image volumes. In Section 3, we show how some of the parameters of the proposed method can be automatically estimated. In Section 4, we evaluate the proposed method’s performance against another general-purpose interactive segmentation method. Finally, in Section 5 we provide brief concluding remarks.

2. INTERACTIVE MATERIALS IMAGE SEGMENTATION

In²⁵ we developed a 3D materials science image segmentation method by propagating segmentation S^U of a slice U to a neighboring slice V , resulting in a segmentation S^V . This way, using an initial segmentation on one slice, we can repeatedly propagate this segmentation to the remaining slices in the volume to obtain a complete 3D segmentation. This propagation was done while preserving the topology (i.e., non-adjacency relations among 2D segments), which led to a better performance when compared with methods that did not incorporate topology as a prior. Specifically, let the segmentation

$$S^U = \{S_1^U, S_2^U, \dots, S_n^U\},$$

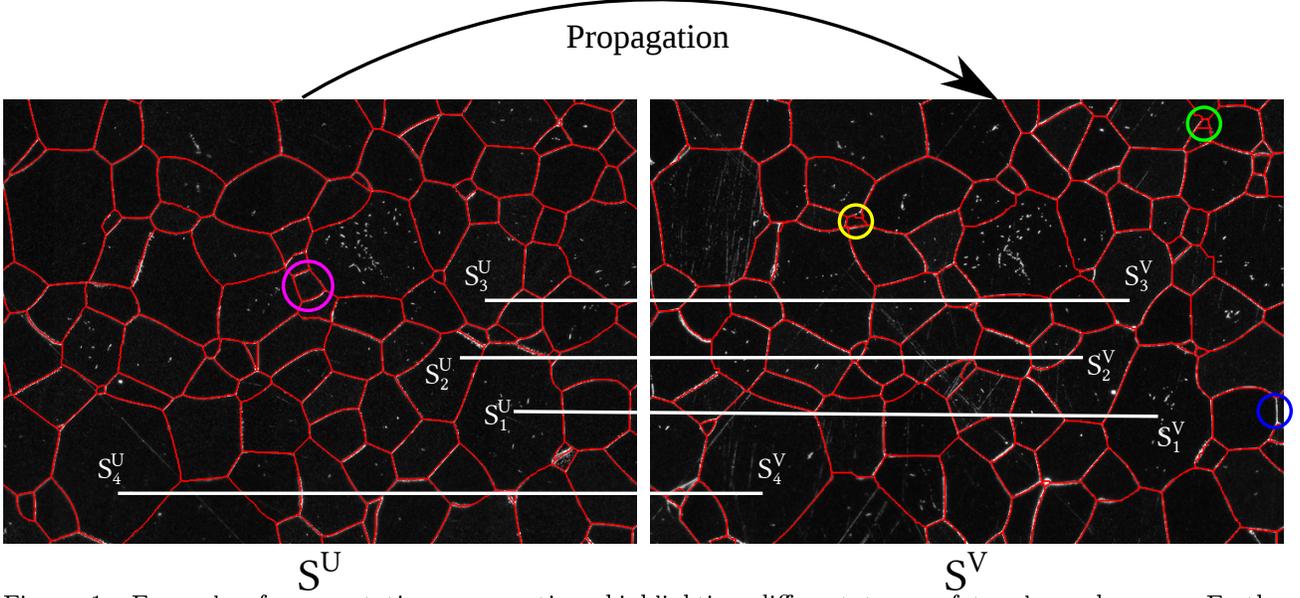


Figure 1: Example of segmentation propagation, highlighting different types of topology changes. Further discussion in the text.

where $S_i^U, i = 1 \dots n$ are disjoint segments in slice U , and this collection of segments makes up a partition of the slice U ,

$$U = \bigcup_{i=1}^n S_i^U.$$

An example is shown in Figure 1 where all the segments (“grain” structures) are separated by red lines. To propagate segmentation S^U to a new slice V to yield the segmentation S^V , we minimize the energy

$$E(S^V) = \sum_{p \in V} \Theta_p(S_i^V) + \sum_{\{p,q\} \in \mathcal{P}_n^V} \Phi_{pq}(S_i^V, S_j^V) \quad (1)$$

where \mathcal{P}_n^V is the set of all 4-connected pixels. The unary term $\Theta_p(S_i^V)$, which represents a cost for a pixel p being assigned to a segment S_i^V in slice V , was set to reflect the structure continuity between U and V ,

$$\Theta_p(S_i^V) = \begin{cases} 0, & \text{distance}(p, S_i^U) < d \\ \infty, & \text{otherwise} \end{cases} \quad (2)$$

where d is a dilation distance that reflects the maximum possible structural change between U and V .²⁵ In addition, the binary term $\Phi_{pq}(S_i^V, S_j^V)$, which represents a cost for a pair of neighboring pixels p, q being assigned to two (possibly the same) segments S_i^V, S_j^V , was constrained to preserve *non-adjacency* segment relationships from U to V ; i.e., any two segments S_i^V, S_j^V are allowed to be adjacent (have pixels that are 4-connected between them) if the corresponding segments in S_i^U, S_j^U are also adjacent,

$$\Phi_{pq}(S_i^V, S_j^V) = \begin{cases} 0, & i = j \\ \infty, & \{S_i^U, S_j^U\} \notin \mathcal{A}^U \\ g(p, q), & \{S_i^U, S_j^U\} \in \mathcal{A}^U \end{cases}, \quad (3)$$

where \mathcal{A}^U contains segment pairs that are adjacent in S^U , and we set $g(p, q)$ to reflect the image boundary information in V .²⁵ An example is shown in Figure 1, where S_1^V and S_2^V are allowed to be adjacent because S_1^U and S_2^U are adjacent in S^U . However, S_1^V and S_4^V are not allowed to be adjacent (have an infinity penalty) because S_1^U and S_4^U are not adjacent in S^U . This topology constraint was found to be particularly important

for materials images, and our proposed method was able to outperform other methods that did not incorporate such a prior.

While finding the global minimum to this cost is NP-hard, this cost has been shown to be minimizable to a local optimum using a graph-cut approach.^{32,33} However, one phenomenon that was observed in this previous work was that, during propagation, 2D structure topology between U and V might not always be fully consistent. For example, a new 3D structure with no intersection in slice U might appear in slice V , e.g., the structure in the yellow circle in Figure 1. Similarly, a 3D structure intersected by slice U might disappear in slice V , such as the structure circled in magenta in Figure 1. This breaks the topology constraints given in Eq. (3) in some local regions. This may lead to spurious segments and missing structures, as circled in green and blue respectively, in Figure 1.

The previous method made use of a brute-force automated search to locate such spurious and missing structures in V .²⁵ However, particularly when the inter-slice distance is too large, it is not possible to examine every location for possible spurious or missing structures. In this paper, our goal is to develop effective interactive tools to allow a user to conveniently specify the local areas that contain spurious or missing structures, and incorporate such interactions to refine the segmentation S^V to a corrected \tilde{S}^V on slice V , using the same energy minimization algorithm. more specifically, we propose to allow the user to correct these two types of segmentation errors within this framework by: a) annotating the location of a new segment to handle cases where a new structure appears in slice V , and b) annotation of existing segments that should no longer be present in segmentation S^V .

These interactions are inherently local because the 2D cross section of a 3D structure shows very small size before appearing or disappearing from a neighboring 2D slice. Therefore, correcting S^V to \tilde{S}^V can be achieved by using the same energy minimization in local image areas around the interactive annotations. This is also important because interactive segmentation requires instantaneous user feedback. The previous propagation method segmented entire slices, which was more computationally intensive than is desirable in an interactive system. We will further discuss these two interactions, and how we identify local regions for each, in the following subsections.

2.1 Removal of Spurious Segments

For this interaction, we allow the user to select a spurious segment S_k^V for removal by clicking the mouse on this segment in a visualized segmentation of S^V . Instead of naively removing this segment by arbitrarily merging it into one of its neighbors, we use the same energy minimization discussed above to assign the individual pixels in S_k^V to potentially different neighboring segments. As discussed above, we identify a local region in which we update the segmentation. Specifically, this local region consists of the specified S_k^V and its neighboring segments, e.g., S_1^V, S_2^V, S_3^V surrounding the spurious segment S_k^V in Figure 2 (a), and re-run the energy minimization within this local region after modifying the Θ term to incorporate the interaction, resulting in an updated segmentation in this local region, as shown by the example in Figure 2 (b). For ease of notation, we use similar notation to the adjacency definition in Eq. (3) by using $\{\mathcal{A}^V\}_k$ to refer to the set of segments neighboring the segment S_k^V . This way, the local region for updating the segmentation is

$$\mathcal{L} = \{\mathcal{A}^V\}_k \cup S_k^V. \quad (4)$$

In this local region, we rerun the energy minimization of Eq. (1) by modifying the Θ term. In particular, we do not allow any pixel to be assigned to S_k^V since this segment is to be removed. Instead, the pixels initially in S_k^V can be assigned to any of the segments in $\{\mathcal{A}^V\}_k$ with 0 cost for the Θ term, i.e.,

$$\begin{aligned} \forall p \in S_k^V, \quad \Theta_p(\tilde{S}_i^V) &= \begin{cases} 0, & S_i^V \in \{\mathcal{A}^V\}_k \\ \infty, & \text{otherwise} \end{cases} \\ \forall p \notin S_k^V, \quad \Theta_p(\tilde{S}_i^V) &= \Theta_p(S_i^V) \end{aligned} \quad (5)$$

By updating Θ in this fashion, we do not require the pixels previously in S_k^V merged into a single neighboring segment. Instead, these pixels may be assigned to more than one segment in $\{\mathcal{A}^V\}_k$, as shown in Figure 2 (b).

Note that this interaction is very simple and convenient, as it requires only a single click anywhere inside the spurious segment S_k^V . The full algorithm for removing spurious segments is summarized in Algorithm 1.

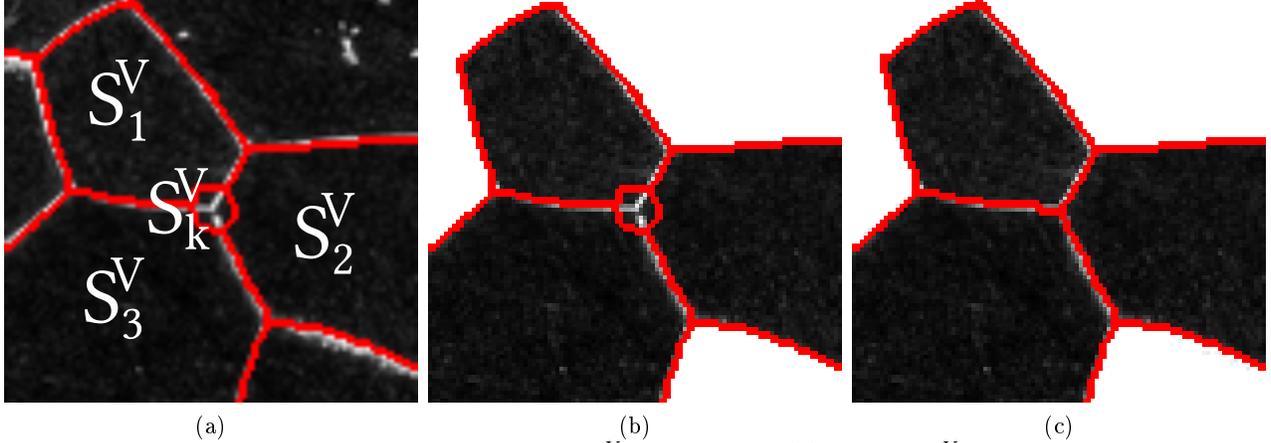


Figure 2: Example selection of a spurious segment S_k^V for removal. **(a)** Chosen S_k^V and surrounding segments. **(b)** Local region extracted around S_k^V . **(c)** The updated segmentation in the extracted local region.

Algorithm 1 Interactively specifying segment to remove.

- 1: **function** REMOVESEGMENT(S^V, S_k^V)
 - 2: $\mathcal{L} \leftarrow \{\mathcal{A}^V\}_k \cup S_k^V$
 - 3: $\forall p \in \mathcal{L}$, build graph for energy minimization problem from²⁵
 - 4: $\Theta \leftarrow$ set from Eq. (5)
 - 5: $\tilde{S}^V \leftarrow S^V$ incorporating the updated segmentation in \mathcal{L}
 - 6: **return** updated \tilde{S}^V
 - 7: **end function**
-

2.2 Addition of Missing Segments

Unlike removal, interactively annotating an additional structure cannot be solely formulated as a simple modification of the Θ term in the energy minimization formulation. This is because the multi-labeling problem used to optimize the energy minimization form in Eq. (1) optimizes over a fixed set of segments, and cannot introduce new segments. Thus, for each missing segment, we must explicitly create a new segment at the location interactively specified by the user.

Based on the initial segmentation $S^V = \{S_1^V, S_2^V, \dots, S_n^V\}$, we take as input from the user an annotation specifying the center location c of the new segment \tilde{S}_{n+1}^V . In addition to this, we also accept two parameters from the user: 1) the *seed* radius s specifying a circular region around c such that this circular region is completely contained within the missing structure; 2) a *dilation* radius d , which is similar to the dilation parameter used in,²⁵ such that the circular region with this dilation radius d centered at c completely covers the missing structure to be segmented. We explicitly enforce that $d \geq s$ for any choice of s . We call pixels within the seed radius s of c “seed pixels” and pixels within the dilation radius d of c “dilation pixels.” In this interaction, seed pixels are *guaranteed* to be part of the missing segment that is added, as shown by the green circle in Figure 3(b), and dilation pixels, excluding seed pixels, are *potentially* part of the missing segment that is added, as shown by the blue area in Figure 3(b). This makes the selection of s and d conceptually simple for the user to tune. In Section 3, we discuss how to automate the selection of s and d to further reduce the user’s burden when interactively segmenting a materials volume.

Similar to the removal interaction in Section 2.1, we define a local region around the specified c to update the segmentation of S^V . Specifically, we define this region by taking all segments in S^V that contain one or more seed or dilation pixels. In this local region we modify the Θ term of the energy minimization in Eq. (1) to obtain an updated segmentation. Specifically, we allow all seed and dilation pixels to be reassigned to the new segment \tilde{S}_{n+1}^V by setting

$$\Theta_p(\tilde{S}_{n+1}^V) = \begin{cases} 0, & \|p - c\| \leq d \\ \infty, & \text{otherwise} \end{cases} \quad (6)$$

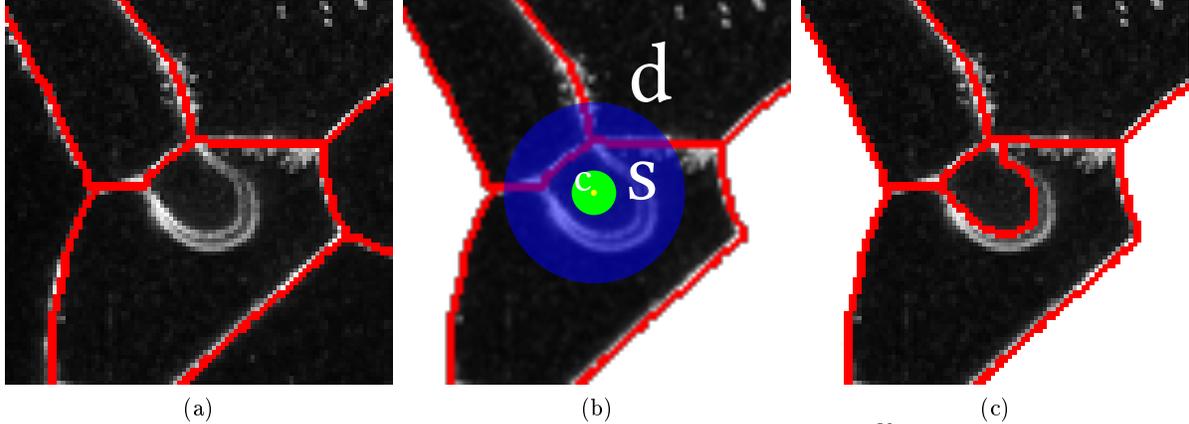


Figure 3: Annotating the addition of a missing segment. (a) Segmentation S^V with a missing segment near the center of the image. (b) Annotation of a center point c , along with a seed radius s and a dilation radius d , and the identified local region for updating the segmentation. (c) The updated segmentation of the local region shown in (b).

Algorithm 2 Interactively specifying segment to add.

- 1: **function** ADDSEGMENT(S^V, c, s, d)
 - 2: $\mathcal{L} \leftarrow$ union of all segments that contain a seed pixel or dilation pixel
 - 3: $\forall p \in \mathcal{L}$, build graph for energy minimization problem from²⁵
 - 4: $\Theta \leftarrow$ set from Eq. (6) and Eq. (7)
 - 5: $\tilde{S}^V \leftarrow S^V$ incorporating the updated segmentation in \mathcal{L}
 - 6: **return** updated \tilde{S}^V
 - 7: **end function**
-

where $\|\cdot\|$ is the euclidean distance between pixels p and c . Furthermore, to insure that the seed pixels are always guaranteed to be part of \tilde{S}_{n+1}^V we set an infinity penalty for seed pixels assigned to any segment other than \tilde{S}_{n+1}^V ,

$$\Theta_p(\tilde{S}_i^V) = \begin{cases} \infty, & \|p - c\| \leq s \text{ and } i \neq n + 1 \\ \Theta_p(S_i^V), & \text{otherwise.} \end{cases} \quad (7)$$

The full algorithm for adding a missing segment is summarized in Algorithm 2.

3. PARAMETER ESTIMATION

When interactively adding a new segment, as discussed in Section 2.2, the seed radius s and dilation radius d are required to be specified by the user. This results in additional burden on the part of the user. In this section, we develop a parameter estimation approach to automatically select these two parameters so the user need only override them in very rare cases, or not at all.

We do this by leveraging information about the center c the user provided relative to the initial segment in which it resides. Generally a missing segment occurs when 2D cross-section intersects a new 3D structure in V . Given a small inter-slice distance, we expect that these missing segments are often small compared with its neighboring segments in slice V . An example is shown in Figure 4 (a), where a small segment is missing (indicated by the yellow circle) in the segmentation S^V : this missing segment is mistakenly merged into a large neighboring segment S_b^V . Intuitively, placing c near the boundary of S_b^V likely indicates the missing segment is small, as shown by Figure 4 (b). Conversely, placing c closer to the center of S_b^V likely indicates the resulting missing segment is large as shown in Figure 4 (c). We make a simplifying assumption that we do not allow the missing segment to spill over the boundary of S_b^V . For example, the selection of c and s in Figure 4 (b) is able to generate the updated segmentation shown in Figure 4 (d).

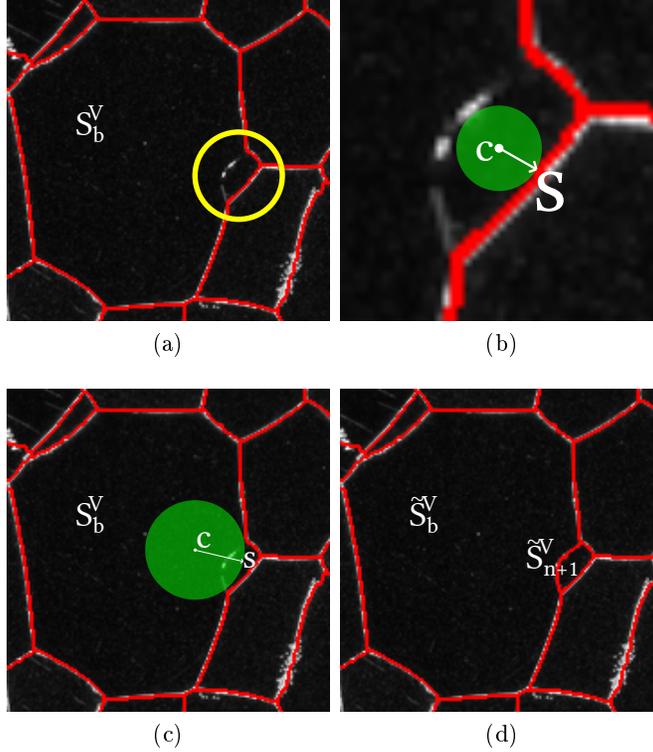


Figure 4: Automatic selection of seed radius s and dilation radius d . **(a)** A missing segment located within a large segment in S_b^V . **(b-c)** Selections of c at varying distances from the boundary of S_b^V , resulting in different estimations of s . **(d)** Updated \tilde{S}^V by adding a missing segment using the c shown (b) and the proposed parameter estimation method to determine s .

To obtain an estimation of s we start by setting $s = 0$, and we then increase s by a small ϵ amount until the circle centered at c with radius s is within ϵ distance of the boundary of the containing segment S_b^V , as shown by the arrow in Figure 5 (b-c). In materials images, the majority of newly-appearing structures when moving from one slice to another are usually near the boundary of an existing structure S_b^V (near a “Y”-junction boundary between structures). This automatic approach is ideally suited for these cases. When the user specifies a c that falls directly on a segment boundary in S^V , we default to requiring user-supplied s in these less-common cases. For estimating d , it is scaled according to the value of s . Specifically, we set $d = 2 \cdot s$. As shown in Section 4, this approach saves both time and effort.

4. EXPERIMENTS

To evaluate the proposed interactive segmentation method, we use it to segment a sequence of 11 (indexed from 0 to 10) microscopic titanium images³⁴ provided by Dave Rowenhorst, NRL. We measure the effort (i.e., number of clicks) used to segment each slice in the dataset, as well as the overall time expended by the user to segment a slice. The previous segmentation propagation approach²⁵ requires a complete segmentation on one slice as an initialization. We count the manual segmentation on this initial slice into the effort and time required. We present the proposed method both with and without using the automatic parameter estimation discussed in Section 3.

For comparison, we use the readily-available “intelligent scissors” interactive segmentation method.⁵ Using the intelligent scissors tool, we independently segment all 11 slices from the same dataset, evaluating both effort (number of clicks) and time. In addition, we produce a hybrid of our previous automatic method²⁵ and the intelligent scissors method, which we call “intelligent scissors + propagation” in Figure 5. This approach uses

the method from²⁵ to propagate a segmentation from an initial slice to the remaining slices, but it uses the intelligent scissors tool⁵ to carry out the interactive component instead of the interaction proposed in this paper.

The results of this comparative experiment are shown in Figure 5. Note that, in propagated methods (“Proposed,” “Proposed + Parameter Estimation,” and “Intelligent Scissors + Propagation”), the first slice is used as the initial slice U , so it requires significantly more effort and time to segment compared with the remaining slices. From Figure 5, we can see that the method proposed in this paper (“Proposed”) allows much more rapid

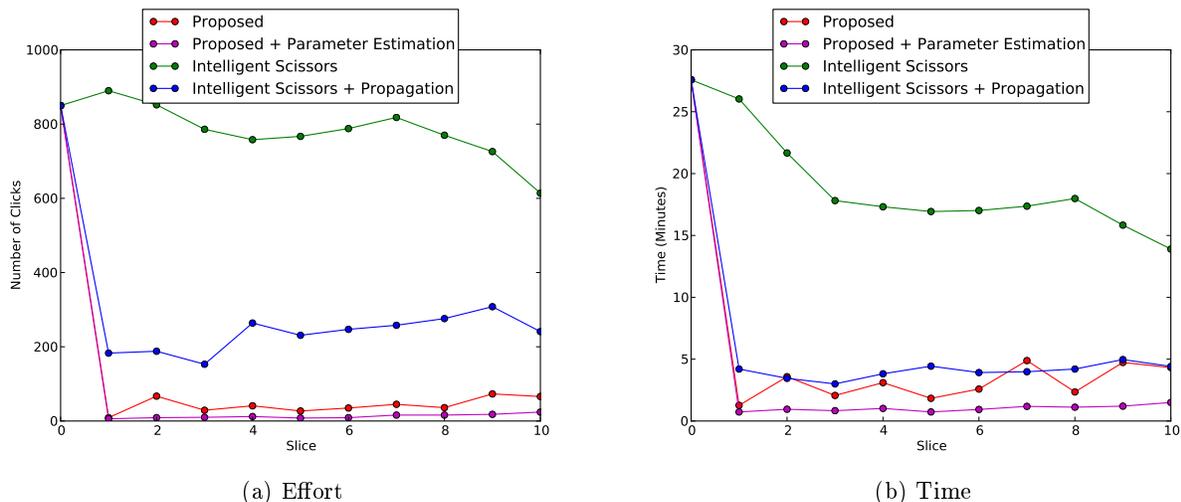


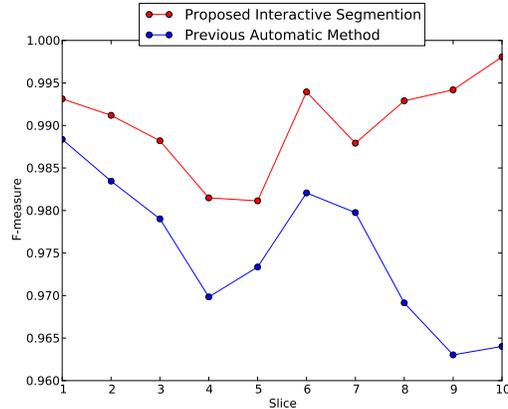
Figure 5: Evaluation of **(a)** the amount of effort (number of clicks) and **(b)** time taken for a user to interactively segment the 11 sample slices. Smaller values are better, for both figures.

segmentation time (< 5 minutes in most cases) and with much less effort (< 100 clicks in most cases) compared with the unpropagated intelligent scissors method. The intelligent scissors method (“Intelligent Scissors”), without the benefit of propagation, requires significantly more time and effort. The hybrid method (“Intelligent Scissors + Propagation”) fares better than the unpropagated intelligent scissors method, but it still requires greater effort than the proposed method. Finally, the proposed parameter estimation method (“Proposed + Parameter Estimation”) can further reduce both the time and effort required by the proposed method.

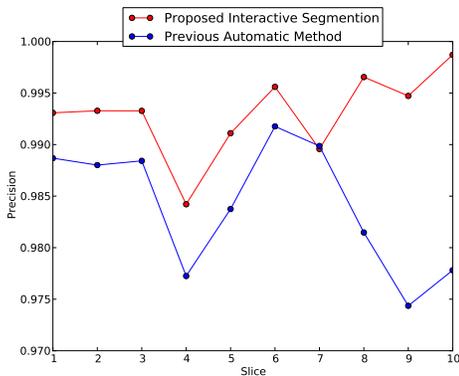
In Figure 6, we show that the proposed interactive method is able to increase the segmentation accuracy of our state-of-the-art materials image segmentation method in.²⁵ As in our previous work,²⁵ we use the precision, recall, and F-measure, which is the harmonic mean of the precision and recall,³⁵ to show the segment boundary coincidence with the manually-constructed ground truth segmentation. For both the proposed and previous automatic methods, we propagate from an initial slice 0 to the remaining 10 slices, and the proposed interaction-enhanced method increases performance for all slices. Finally, qualitative segmentation results using the proposed interactive method are shown in Figure 7 where we show the automatic segmentation results with spurious or missing segments, the human annotation, and the updated segmentation.

5. CONCLUSION

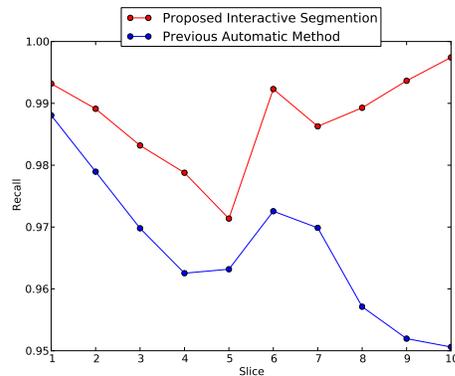
We have presented an interactive segmentation method extended from our automatic segmentation propagation approach. By allowing the user to interactively handle spurious and missing segments when propagating from one slice to another, we show that the time required to segment a materials image volume, as well as the overall effort (number of clicks) needed for interaction, is much less than the comparison “intelligent scissors” method used in popular image processing tools. By updating the segmentation within a local region around the interactive annotation, we are able to obtain a fast, yet robust means to handle segmentation errors when a new structure appears or an existing structure disappears from the 2D cross-section of a particular slice in the volume. We also introduce a simple automatic technique to estimate the seed radius when adding a missing segment. We show that this can further reduce the amount of time and effort needed by the proposed approach.



(a) F-measure



(b) Precision



(c) Recall

Figure 6: Performance of the proposed interactive segmentation method compared with our previous automated method²⁵ measured by the boundary coincidence with the ground truth segmentation.

REFERENCES

- [1] Kuang, Z., Schieders, D., Zhou, H., Wong, K.-Y., Yu, Y., and Peng, B., “Learning image-specific parameters for interactive segmentation,” in *[IEEE Conference on Computer Vision and Pattern Recognition]*, 590–597 (2012).
- [2] Straehle, C., Koethe, U., Knott, G., Briggman, K., Denk, W., and Hamprecht, F., “Seeded watershed cut uncertainty estimators for guided interactive segmentation,” in *[IEEE Conference on Computer Vision and Pattern Recognition]*, 765–772 (2012).
- [3] Settles, B., “Active learning literature survey,” Computer Sciences Technical Report 1648, University of Wisconsin–Madison (2009).
- [4] Rother, C., Kolmogorov, V., and Blake, A., “GrabCut: Interactive foreground extraction using iterated graph cuts,” *ACM Transactions on Graphics* **23**(3), 309–314 (2004).
- [5] Mortensen, E. N. and Barrett, W. A., “Intelligent scissors for image composition,” in *[Proceedings of the 22nd annual conference on Computer graphics and interactive techniques]*, 191–198 (1995).
- [6] Santner, J., Pock, T., and Bischof, H., “Interactive multi-label segmentation,” in *[IEEE Asian Conference on Computer Vision]*, 397–410 (2011).
- [7] Unger, M., Pock, T., Trobin, W., Cremers, D., and Bischof, H., “TVSeg—interactive total variation based image segmentation,” in *[British Machine Vision Conference 2008]*, 40.1–40.10 (2008).
- [8] Boykov, Y. and Jolly, M., “Interactive graph cuts for optimal boundary & region segmentation of objects in nd images,” in *[IEEE International Conference on Computer Vision]*, **1**, 105–112, IEEE (2001).

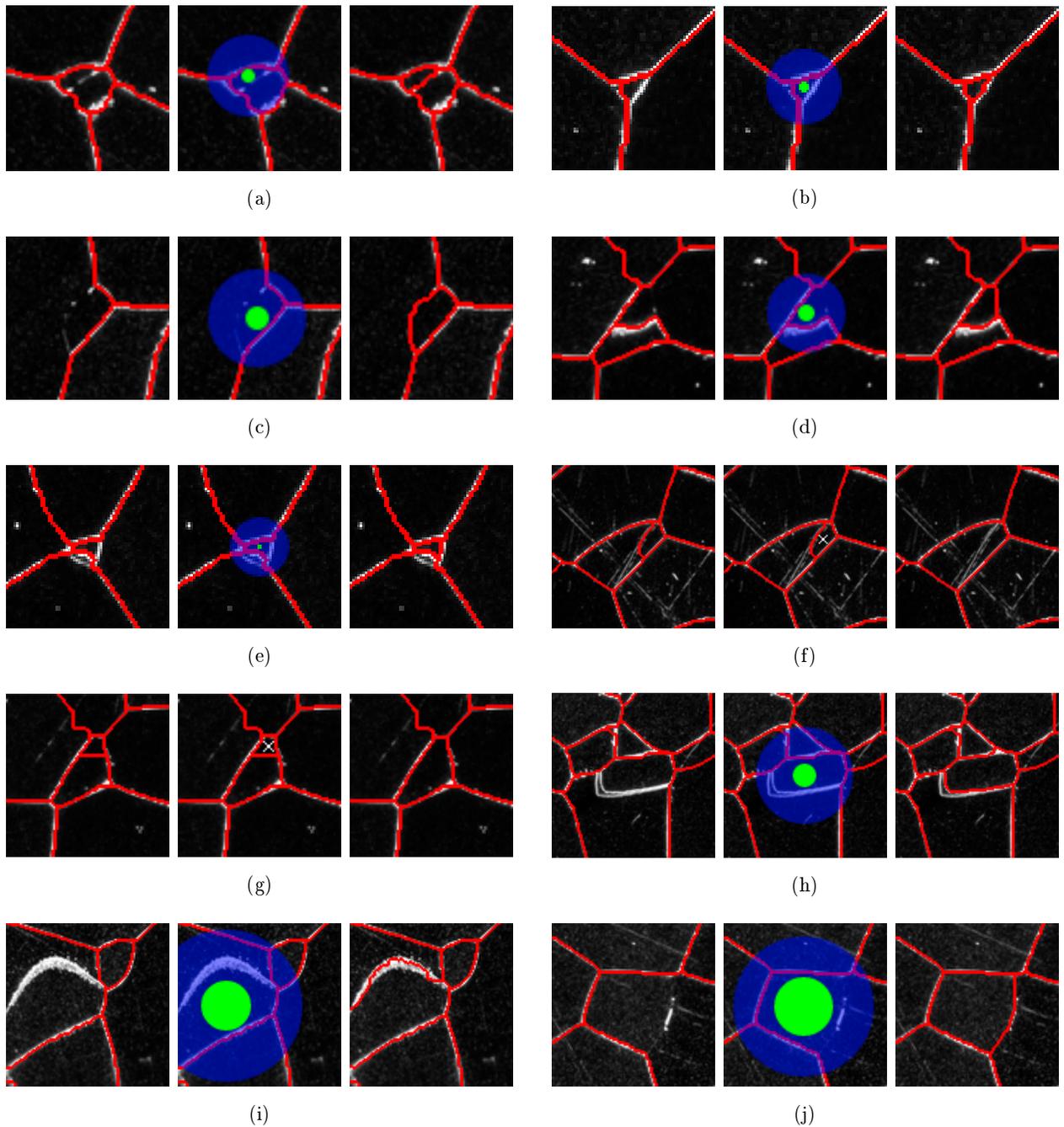


Figure 7: Qualitative results where each subfigure shows the initial automatic segmentation S^V (left); the human annotation (middle) with the seed pixels in green and dilation pixels in blue, and “X”s indicating spurious segments to be removed; and the updated segmentation \tilde{S}^V (right). Note that (f) and (g) illustrate removal annotation and the remaining illustrate addition annotation.

- [9] Vezhnevets, V. and Konouchine, V., “Grow-Cut—interactive multi-label N-D image segmentation,” in [*Graphics*], 150–156 (2005).
- [10] Batra, D., Kowdle, A., Parikh, D., Luo, J., and Chen, T., “iCoseg: Interactive co-segmentation with intelligent scribble guidance,” in [*IEEE Conference on Computer Vision and Pattern Recognition*], 3169–3176 (2010).
- [11] Top, A., Hamarneh, G., and Abugharbieh, R., “Active learning for interactive 3D image segmentation,” in [*Proceedings of the 14th international conference on Medical image computing and computer-assisted intervention*], 603–610 (2011).
- [12] Boykov, Y. and Jolly, M.-P., “Interactive organ segmentation using graph cuts,” in [*Medical Image Computing and Computer-Assisted Intervention*], **1935**, 147–175 (2000).
- [13] Straehle, C. N., Köthe, U., Knott, G., and Hamprecht, F. A., “Carving: scalable interactive segmentation of neural volume electron microscopy images,” in [*Medical Image Computing and Computer-Assisted Intervention*], 653–660 (2011).
- [14] Ibrahim, I. A., Mohamed, F. A., and Lavernia, E. J., “Particulate reinforced metal matrix composites—a review,” *Journal of Materials Science* **26**, 1137–1156 (1991).
- [15] Swiler, T. P. and Holm, E. A., “Diffusion in polycrystalline microstructures,” in [*Annual Meeting of the American Ceramic Society*], (1995).
- [16] Rollett, A., Gottstein, G., Shvindlerman, L., and Molodov, D., “Grain boundary mobility: a brief review,” *Z. Metallkunde* **95**, 226–229 (2004).
- [17] Reed, R., [*The Superalloys: Fundamentals and Applications*], Cambridge University Press (2006).
- [18] Tan, J. and Saltzman, W., “Biomaterials with hierarchically defined micro- and nanoscale structure,” *Biomaterials* **25**(17), 3593–3601 (2004).
- [19] Chuang, H., Huffman, L., Comer, M., Simmons, J., and Pollak, I., “An automated segmentation for nickel-based superalloy,” in [*IEEE International Conference on Image Processing*], 2280–2283 (2008).
- [20] Simmons, J. P., Chuang, P., Comer, M., Spowart, J. E., Uchic, M. D., and Graef, D. M., “Application and further development of advanced image processing algorithms for automated analysis of serial section image data,” *Modelling and Simulation in Materials Science and Engineering* **17**(2), 025002 (2009).
- [21] Comer, M. and Delp, E., “Parameter estimation and segmentation of noisy or textured images using the EM algorithm and MPM estimation,” in [*IEEE International Conference on Image Processing*], **2**, 650–654 (1994).
- [22] Comer, M. and Delp, E., “The EM/MPM algorithm for segmentation of textured images: Analysis and further experimental results,” *IEEE Transactions on Image Processing* **9**(10), 1731–1744 (2000).
- [23] Marroquin, J., Mitter, S., and Poggio, T., “Probabilistic solution of ill-posed problems in computational vision,” *Journal of the American Statistical Association*, 76–89 (1987).
- [24] Huffman, L. M., Simmons, J. P., Graef, M. D., and Pollak, I., “Shape priors for map segmentation of alloy micrographs using graph cuts,” in [*IEEE Workshop on Statistical Signal Processing*], 28–30 (2011).
- [25] Waggoner, J., Simmons, J., and Wang, S., “Combining global labeling and local relabeling for metallic image segmentation,” in [*Proceedings of SPIE (Computational Imaging X)*], **8296** (2012).
- [26] Huffman, L., Simmons, J., and Pollak, I., “Segmentation of digital microscopy data for the analysis of defect structures in materials using nonlinear diffusion,” in [*Proceedings of SPIE (Computational Imaging VI)*], (2008).
- [27] Comer, M., Bouman, C., De Graef, M., and Simmons, J., “Bayesian methods for image segmentation,” *JOM Journal of the Minerals, Metals and Materials Society* **63**, 55–57 (2011).
- [28] Simmons, J., Bartha, B., De Graef, M., and Comer, M., “Development of bayesian segmentation techniques for automated segmentation of titanium alloy images,” *Microscopy and Microanalysis* **14**(S2), 602–603 (2008).
- [29] Li, Q., Ni, X., and Liu, G., “Ceramic image processing using the second curvelet transform and watershed algorithm,” in [*IEEE International Conference on Robotics and Biomimetics*], 2037–2042 (2007).
- [30] Gonzalez, R. C. and Woods, R. E., [*Digital Image Processing (3rd Edition)*], Prentice Hall (2008).
- [31] Shapiro, L. G. and Stockman, G. C., [*Computer Vision*], Upper Saddle River, NJ: Prentice Hall (2001).

- [32] Veksler, O., *Efficient graph-based energy minimization methods in computer vision*, PhD thesis, Cornell University, Ithaca, NY, USA (1999).
- [33] Boykov, Y., Veksler, O., and Zabih, R., “Fast approximate energy minimization via graph cuts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(11), 1222–1239 (2001).
- [34] Rowenhorst, D., Lewis, A., and Spanos, G., “Three-dimensional analysis of grain topology and interface curvature in a β -titanium alloy,” *Acta Materialia* **58**, 5511–5519 (2010).
- [35] Martin, D., Fowlkes, C., Tal, D., and Malik, J., “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in [*IEEE International Conference on Computer Vision*], **2**, 416–423 (2001).